Integrated Networking Solutions

ESR series service routers

ESR-10, ESR-12V, ESR-12VF, ESR-14VF, ESR-15V, ESR-20, ESR-21, ESR-30, ESR-100, ESR-200, ESR-1000, ESR-1200, ESR-1500, ESR-1700, ESR-1511, ESR-3100, ESR-3200

User manual

Firmware version 1.18.1

Contents

1 Introduction

1.1 Abstract

Today, large-scale communication network development projects are becoming increasingly common. One of the main tasks in implementation of large multiservice networks is the creation of reliable high-performance transport network that will serve as a backbone in multilayer architecture of next-generation networks.

ESR series service routers could be used in large enterprise networks, SMB networks and operator's networks. Devices provide high performance and bandwidth, and feature protection of transmitted data.

This operation manual describes intended use, specifications, design, installation, first time setup, and firmware update guidelines for the ESR series service router (next, the router or the device).

1.2 Target Audience

This user manual is intended for technical personnel that performs device installation, configuration and monitoring via command line interface (CLI) as well as the system maintenance and firmware update procedures. Qualified technical personnel should be familiar with the operation basics of TCP/IP protocol stacks and Ethernet networks design concepts.

1.3 Symbols

Designation	Description
[ ]	In the command line, optional parameters are shown in square brackets; when entered, they provide additional options.
{ }	In the command line, mandatory parameters are shown in curly braces. Choose one of the following:
"," **"-"**	In the command description, these characters are used to define ranges.
"|"	In the description of the command, this sign means 'or'.
Semibold font	Notes, warnings, or information are shown in bold.
<Semibold italic>	Keyboard keys are shown in bold italic within angle brackets.
`Text box`	Examples and results of the commands are given within the text boxes.

1.4 Notes and warnings

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

Warnings inform users about hazardous conditions which may cause injuries or device damage and may lead to the device malfunctioning or data loss.

The information contains information on the use of the device.

2 Product description

2.1 Purpose

ESR series devices are the high performance multi-purpose network routers. Device combines traditional network features with a complex multi-tier approach to routing security, and ensures robust corporate environment protection.

Device has a built-in firewall that enables protection of your and organization network environment and supports latest data security, encryption, authentication and intrusion prevention features.

Device contains software and hardware means of data processing. Top performance is achieved through optimal distribution of data processing tasks between different subsets of the device.

2.2 Functions

2.2.1 Interface functions

Table 1 lists interface functions of the device.

Table 1 – Device interface functions

Cable connection polarity detection (Auto MDI/MDIX)	Automatic cable type detection – crossover cable or straight-through cable. • **MDI** (Medium Dependent Interface – straight) – cable standard for connection of terminal devices; • **MDIX** (Medium Dependent Interface with Crossover – crossed) – cable standard for connection of hubs and switches.
Back Pressure	The backpressure routing method is utilized in half-duplex connections for management of data streams, coming from the opposite devices, by means of collisions. This method allows to avoid buffer overruns and the loss of data.
Flow control (IEEE 802.3X)	Flow control allows to interconnect the low-speed and the high-speed devices. To avoid buffer overrun, the low-speed device gains the ability to send PAUSE packets, that will force the high-speed device to pause the packet transmission.
(LAG, Link aggregation)	Link aggregation allows to increase the communication link bandwidth and robustness. Router supports static and dynamic link aggregation. For dynamic aggregation, link group management is performed via LACP protocol.

2.2.2 MAC table functions

Table 2 lists MAC address processing functions of the device.

Table 2 – MAC address processing functions

MAC table	MAC address table sets the correspondence between MAC addresses and device interfaces and is used for data packet routing. Routers support table capacity up to 128K of MAC addresses and reserve specific MAC addresses for the system use.
Learning mode	MAC address table may contain either static addresses or addresses learnt during data packet transition through the device. Learning involves registration of packet source MAC addresses with their binding to ports and VLANs. Afterwards, this data is used for incoming packet routing. Registered MAC address lifetime is limited. Administrator may adjust this setting. If destination MAC address specified in the packet that was received by the device is not listed in the table, this packet will be sent further as a broadcast packet within L2 segment of the network.

2.2.3 Second-layer functions of OSI model

Table 3 lists second-layer functions and special aspects (OSI Layer 2).

Table 3 – Second-layer functions description (OSI Layer 2)

VLAN support	VLAN (Virtual Local Area Network) is a solution used for splitting a network into separate segments on L2 level. VLAN utilization allows to increase the operation stability for large networks by splitting them into smaller networks, isolate diversified data traffic by type and solve many other tasks. Routers support various VLAN management methods: • VLAN based on data packet tagging according to IEEE802.1Q • VLAN based on device ports (port-based) • VLAN based on utilization of data classification policies (policy-based)
Spanning Tree Protocol[1]	The main task of Spanning Tree Protocol is to exclude redundant network links and convert network topology into the tree-like structure. Common areas of protocol application involve the prevention of network traffic loops and establishing of redundant communication links.

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

2.2.4 Third-layer functions of OSI model

Table 4 lists third-layer functions (OSI Layer 3).

Table 4 – Third-layer functions description (OSI Layer 3)

Static IP routes	Administrator of the router can add or remove static entries into/from the routing table.
Dynamic routing	With dynamic routing protocols, the device will be able to exchange the routing information with neighbouring routers and automatically create a routing table. Router supports the following protocols: RIPv2, RIPng, OSPFv2, OSPFv3, IS-IS, BGP.
ARP table	**ARP** (Address Resolution Protocol) is a protocol used for resolution of the network and data-link layer addresses. ARP table contains information on the established correspondence. Correspondence is established on the basis of the network device response analysis; device addresses are requested with broadcast packets.
DHCP client	DHCP (Dynamic Host Configuration Protocol) protocol enables automation of the network device management process. DHCP client allows the router to obtain the network address and additional settings from the external DHCP server. As a rule, this method is used for obtaining network settings of a public network operator (WAN).

DHCP server	DHCP server enables automation and centralization of the network device configuration process. DHCP server allocated on a router allows for a complete solution for the local area network support. DHCP server integrated into the router assigns IP addresses to network devices and transfers additional network settings, e.g. server addresses, network gateway addresses and other necessary settings.
DHCP Relay	The DHCP Relay functionality is designed to redirect broadcast DHCP Discover packets from one broadcast domain to unicast DHCP Discover packets in another broadcast domain.
NAT, Network Address Translation	Network address translation is a mechanism that translates IP addresses and port numbers for transit packets. NAT function allows to minimize the quantity of IP address used through translation of multiple internal network IP addresses into a single external public IP address. NAT conceals local area network internal structure and allows to enhance its security. Routers support the following NAT options: • **Source NAT** (SNAT) – the network address and the source port number will be replaced, when packet is transferred forth, and the destination address will be replaced in the response packet; • **Destination NAT** (DNAT) – external access is translated by the router to the user computer in LAN that has an internal address and thus directly inaccessible from outside the network (without NAT).

2.2.5 Traffic tunneling functions

Table 5 – Traffic tunneling functions

Tunneling protocols	Tunneling is a method of packet conversion during their network transfer that involves the replacement, modification and addition of a new packet network header. This method may be used for negotiation of transport protocols when the data is transferred through the transit network as well as for creation of secured connections where tunneled data is being encrypted. Routers support the following types of tunnels: • **GRE** – IP packet/Ethernet frame is encapsulated into another IP packet with GRE (General Routing Encapsulation) header; • **IPv4-IPv4** – tunnel that encapsulates source IP packets into IP packets with alternative network parameters; • **L2TPv3** – tunnel for L2 traffic transmission using IP packets; • **IPsec** – tunnel with the encryption of transmitted data; • **L2TP, PPTP, PPPoE, OpenVPN** – tunnels used for establishing a remote 'client-sever' access.

2.2.6 Management and configuration functions

Table 6 – Basic management and configuration functions

Configuration file download and upload	Device parameters are saved into the configuration file that contains configuration data for the specific device ports as well as for the whole system. The following protocols may be used for file transfers: TFTP, FTP, and SCP.
Command line interface (CLI)	CLI management is performed locally via serial port RS-232, or remotely via Telnet, SSH. Console command line interface (CLI) is the industrial standard. CLI interpreter contains the list of commands and keywords that will help the user and reduce the amount of input data.
Syslog	Syslog protocol is designed for transmission of system event messages and event logging.
Ping, traceroute network utilities	Ping and traceroute utilities allow you to check the availability of network devices and identify data transfer routes in IP networks.
Controlled access management – privilege levels	Routers support system access level management for users. Access levels enable responsibility areas management for device administrators. Access levels are numbered from 1 to 15; Level 15 stands for full access to device management features.
Authentication	Authentication is a user identity check procedure. Routers support the following authentication methods: • **local** – local user database stored on the device is used for authentication; • **group** – user database is located on the authentication server. RADIUS and TACACS protocols are user for server interactions.

SSH Server/ Telnet Server	SSH and Telnet server features allow you to establish connection to the device and perform device management.
Automatic configuration restore	Device features automatic configuration restore system designed to prevent remote access loss after re-configuration. If the configuration change is not confirmed in the specified time, configuration will be rolled back to the last known state.

2.2.7 Network security functions

Table 7 lists network security functions of the device.

Table 7 – Network security functions

Security zones	All router interfaces are distributed by security areas. For each zone pair, you can set the rules that determine the possibility of data transmission between zones, data traffic filtering rules.
Data filtering	For each zone pair, you can specify the rule set that manages the filtering process for data transmitted through the router. Device command interface provides appropriate means for detailed configuration of the traffic classification rules and to apply the resulting solution for traffic transmission.

2.3 Main specifications

Table 8 lists main specifications of the router.

Table 8 – Main specifications

General parameters		
Interfaces	ESR-3200	12 × 1000BASE-X/10GBASE-R/25GBASE-R 1 × Console RS-232 (RJ-45) 1 × OOB port 1 × USB 2.0 1 × microSD card slot
	ESR-3100	8 × Combo Ethernet 10/100/1000BASE-T/1000BASE-X 8 × 10GBASE-R/1000BASE-X (SFP+/SFP) 1 × Console RS-232 (RJ-45) 2 × USB 3.0 1 × SD card slot
	ESR-1700	4 × Combo Ethernet 10/100/1000BASE-T/1000BASE-X 8 × 10GBASE-R/1000BASE-X (SFP+/SFP) 2 × Hard disk installation slot 1 × Console RS-232 (RJ-45) 1 × OOB port 2 × USB 2.0
	ESR-1511	4 × Combo Ethernet 10/100/1000BASE-T/1000BASE-X 4 × Ethernet 10/100/1000BASE-T (RJ-45) 4 × 10GBASE-R/1000BASE-X (SFP+/SFP) 2 × 40GBASE-X (QSFP+) 1 × Console RS-232 (RJ-45) 1 × OOB port 2 × USB 2.0 1 × SD card slot

ESR-1500	4 × Combo Ethernet 10/100/1000BASE-T/1000BASE-X 4 × Ethernet 10/100/1000BASE-T (RJ-45) 4 × 10GBASE-R/1000BASE-X (SFP+/SFP) 1 × Console RS-232 (RJ-45) 1 × OOB port 2 × USB 2.0 1 × SD card slot
ESR-1200	4 × Combo Ethernet 10/100/1000BASE-T/1000BASE-X 12 × Ethernet 10/100/1000BASE-T (RJ-45) 8 × 10GBASE-R/1000BASE-X (SFP+/SFP) 1 × Console RS-232 (RJ-45) 2 × USB 2.0 1 × SD card slot
ESR-1000	24 × Ethernet 10/100/1000BASE-T (RJ-45) 2 × 10GBASE-R/1000BASE-X (SFP+/SFP) 1 × Console RS-232 (RJ-45) 2 × USB 2.0 1 × SD card slot
ESR-200	4 × Combo Ethernet 10/100/1000BASE-T/1000BASE-X 4 × Ethernet 10/100/1000BASE-T (RJ-45) 1 × Console RJ-45 1 × USB 3.0 1 × USB 2.0 1 × SD card slot

ESR-100	4 × Combo Ethernet 10/100/1000BASE-T/1000BASE-X 1 × Console RS-232 (RJ-45) 1 × USB 3.0 1 × USB 2.0 1 × SD card slot
ESR-30	4 × Ethernet 10/100/1000BASE-T (RJ-45) 2 × 10GBASE-R/1000BASE-X (SFP+/SFP) 1 × Console RS-232 (RJ-45) 1 × USB 3.0 1 × USB 2.0 1 × microSD card slot
ESR-21	8 × Ethernet 10/100/1000BASE-T (RJ-45) 4 × 1000BASE-X (SFP) 3 × Serial port RS-232 1 × Console RS-232 (RJ-45) 1 × USB 3.0 1 × USB 2.0 1 × SD card slot
ESR-20	2 × Combo Ethernet 10/100/1000BASE-T/1000BASE-X 2 × Ethernet 10/100/1000BASE-T (RJ-45) 1 × Console RS-232 (RJ-45) 1 × USB 3.0 1 × USB 2.0 1 × SD card slot
ESR-15	4 × Ethernet 10/100/1000BASE-T (RJ-45) 2 × 1000BASE-X (SFP) 1 × Console RS-232 (RJ-45) 2 × USB 2.0

	ESR-14VF	8 × Ethernet 10/100/1000BASE-T (RJ-45) 1 × 1000BASE-X (SFP) 1 × Console RS-232 (RJ-45) 4 × FXS 2 × USB 2.0
	ESR-12VF	8 × Ethernet 10/100/1000BASE-T (RJ-45) 1 × 1000BASE-X (SFP) 1 × Console RS-232 (RJ-45) 3 × FXS 1 × FXO 2 × USB 2.0
	ESR-12V	8 × Ethernet 10/100/1000BASE-T (RJ-45) 1 × Console RS-232 (RJ-45) 3 × FXS 1 × FXO 2 × USB 2.0
	ESR-10	4 × Ethernet 10/100/1000BASE-T (RJ-45) 2 × 1000BASE-X (SFP) 1 × Console RS-232 (RJ-45) 2 × USB 2.0
Types of optical transceivers	ESR-3200	1000BASE-X SFP 10GBASE-R SFP+ 25GBASE-R SFP28
	ESR-1511	1000BASE-X SFP 10GBASE-R SFP+ 40GBASE-X QSFP+

	ESR-1700 ESR-3100 ESR-1500 ESR-1200 ESR-1000 ESR-30	1000BASE-X SFP 10GBASE-R SFP+
	ESR-200 ESR-100 ESR-21 ESR-20 ESR-15 ESR-14VF ESR-12VF ESR-10	1000BASE-X SFP
	ESR-15	1000BASE-R SFP+
Duplex or half-duplex interface modes		• duplex and half-duplex modes for electric ports • duplex mode for optical ports
Maximum bandwidth in L2 mode (hardware switching)	ESR-1700 ESR-1511 ESR-1500 ESR-1200	160 Gbps
	ESR-1000	88 Gbps
Data transfer rate	ESR-3200	• optical interfaces 1/10/25 Gbps
	ESR-1511	• electrical interfaces 10/100/1000Mbps • optical interfaces 1/10/40 Gbps

	ESR-3100 ESR-1700 ESR-1500 ESR-1200 ESR-1000	• electrical interfaces 10/100/1000Mbps • optical interfaces 1/10Gbps
	ESR-200 ESR-100 ESR-21 ESR-20 ESR-15 ESR-14VF ESR-12V(F) ESR-10	• electrical interfaces 10/100/1000Mbps • optical interfaces 1Gbps
Number of VPN tunnels	ESR-3200 ESR-3100 ESR-1700 ESR-1511 ESR-1500 ESR-1200 ESR-1000	500
	ESR-200 ESR-100 ESR-30 ESR-21 ESR-20	250

	ESR-15 ESR-14VF ESR-12V(F) ESR-10	10
Number of static routes	ESR-3200 ESR-3100 ESR-1700 ESR-1511 ESR-1500 ESR-1200 ESR-1000 ESR-200 ESR-100 ESR-30 ESR-21 ESR-20	11k
	ESR-15 ESR-14VF ESR-12V(F) ESR-10	1k
Number of concurrent sessions	ESR-3200 ESR-3100 ESR-1700 ESR-1511 ESR-1500 ESR-1200 ESR-1000	512k

	ESR-200 ESR-100 ESR-30 ESR-21 ESR-20	256k
	ESR-15 ESR-14VF ESR-12V(F) ESR-10	4k
VLAN support		up to 4k active VLANs according to 802.1Q
Number of BGPv4/BGPv6 routes	ESR-3200 ESR-3100 ESR-1700 ESR-1511 ESR-1500 ESR-1200 ESR-1000	5M
	ESR-200 ESR-100 ESR-30 ESR-21 ESR-20	2.5M
	ESR-15 ESR-14VF ESR-12V(F) ESR-10	1M

Number of OSPFv2/OSPFv3/IS-IS routes	ESR-3200 ESR-3100 ESR-1700 ESR-1511 ESR-1500 ESR-1200 ESR-1000	500k
	ESR-200 ESR-100	300k
	ESR-30 ESR-21 ESR-20 ESR-15 ESR-14VF ESR-12V(F) ESR-10	30k
Number of RIP/RIPng routes	ESR-3200 ESR-3100 ESR-1700 ESR-1511 ESR-1500 ESR-1200 ESR-1000 ESR-200 ESR-100 ESR-30 ESR-21 ESR-20	10k

	ESR-15 ESR-14VF ESR-12V(F) ESR-10	1k
MAC address table	ESR-1700 ESR-1511 ESR-1500 ESR-1200	128k entries
	ESR-3200 ESR-1000	16k entries
	ESR-3100 ESR-200 ESR-100 ESR-30 ESR-21 ESR-20 ESR-15 ESR-14VF ESR-12V(F) ESR-10	2k bridge entries
FIB size	ESR-1700	3.0M
	ESR-3200 ESR-3100 ESR-1511 ESR-1500 ESR-1200 ESR-1000	1.7M

	ESR-200 ESR-100 ESR-30 ESR-21 ESR-20	1.4M
	ESR-15 ESR-14VF ESR-12V(F) ESR-10	800k
VRF Lite		32
L3 interfaces	ESR-3200 ESR-3100 ESR-1700 ESR-1500 ESR-1511 ESR-1200 ESR-1000 ESR-200 ESR-100 ESR-30 ESR-21 ESR-20	4000
	ESR-15 ESR-14VF ESR-12V(F) ESR-10	200

Compliance		IEEE 802.3 10BASE-T Ethernet IEEE 802.3u 100BASE-T Fast Ethernet IEEE 802.3ab 1000BASE-T Gigabit Ethernet IEEE 802.3z Fiber Gigabit Ethernet IEEE 802.3ba 40GBASE-SR4, 40GBASE-LR4 ANSI/IEEE 802.3 Speed autodetection IEEE 802.3x Data flow control IEEE 802.3ad LACP link aggregation IEEE 802.1q VLAN virtual local networks IEEE 802.1v IEEE 802.3ac IEEE 802.3ae IEEE 802.1D IEEE 802.1w IEEE 802.1s
Control		
Local control		CLI
Remote control		TELNET, SSH
Physical specifications and ambient conditions		
Power supply	ESR-1700	AC: 176–264 V, 50–60 Hz DC: 36–72 V Power options: • single AC or DC power supply; • two AC or DC power supplies with hot swapping.

	ESR-3200 ESR-3100 ESR-1511 ESR-1500 ESR-1200 ESR-1000	AC: 100–240 V, 50–60 Hz DC: 36–72 V Power options: • single AC or DC power supply; • two AC or DC power supplies with hot swapping.
	ESR-200 ESR-100 ESR-30 ESR-21 ESR-20 ESR-14VF ESR-12V(F)	AC: 100–264 V, 50–60 Hz
	ESR-15	220 V/12 V, 2 A DC power adapter
	ESR-10	220 V/12 V, 1.5 A DC power adapter
Maximum power consumption:	ESR-3200	118 W
	ESR-3100	123 W
	ESR-1700	250 W
	ESR-1511	128 W
	ESR-1500	125 W
	ESR-1200	85 W
	ESR-1000	75 W
	ESR-200	25 W
	ESR-100	20 W
	ESR-30	26 W
	ESR-21	32 W

	ESR-20	25 W
	ESR-15	18 W
	ESR-14VF ESR-12V(F)	22 W
	ESR-10	9 W
Weight	ESR-3200	5 kg
	ESR-3100	4.34 kg
	ESR-1700	12 kg
	ESR-1511 ESR-1500	7 kg
	ESR-1200	5.5 kg
	ESR-1000	3.6 kg
	ESR-200 ESR-100	2.5 kg
	ESR-30	1.8 kg
	ESR-21	3.15 kg
	ESR-20	2 kg
	ESR-15	0.325 kg
	ESR-14VF ESR-12V(F) ESR-10	1 kg
Dimensions (W × H × D)	ESR-3200 ESR-3100	430 × 44 × 330 mm
	ESR-1700	440 × 88 × 490 mm

	ESR-1511 ESR-1500	430 × 44 × 425 mm
	ESR-1200 ESR-1000	430 × 44 × 352 mm
	ESR-200 ESR-100	310 × 44 × 240 mm
	ESR-21	430 × 44 × 225 mm
	ESR-30 ESR-20	267 × 44 × 212 mm
	ESR-15	230 × 32 × 133 mm
	ESR-14VF ESR-12V(F)	267 × 43.6 × 160.5 mm
	ESR-10	185 × 32 × 118 mm
Operating temperature range	ESR-3200 ESR-3100 ESR-1700 ESR-1511 ESR-1500 ESR-1200 ESR-1000 ESR-200 ESR-100 ESR-30 ESR-21 ESR-20	-10 to +45 °C

	ESR-15 ESR-14VF ESR-12V(F) ESR-10	0 to +40 °C
Storage temperature range		-40 to +70 °C
Operation relative humidity (non-condensing)		up to 80 %
Storage relative humidity (non-condensing)		from 10 to 95 %
Lifetime		at least 15 years

2.4 Design

This section describes the design of the device. Depicted front, rear, and side panels of the device, connectors, LED indicators and controls.

The device has a metal-enclosed design for 1U 19" racks; housing size is 1U.

2.4.1 ESR-3200 design

ESR-3200 front panel

The front panel layout is depicted in Figure 1.

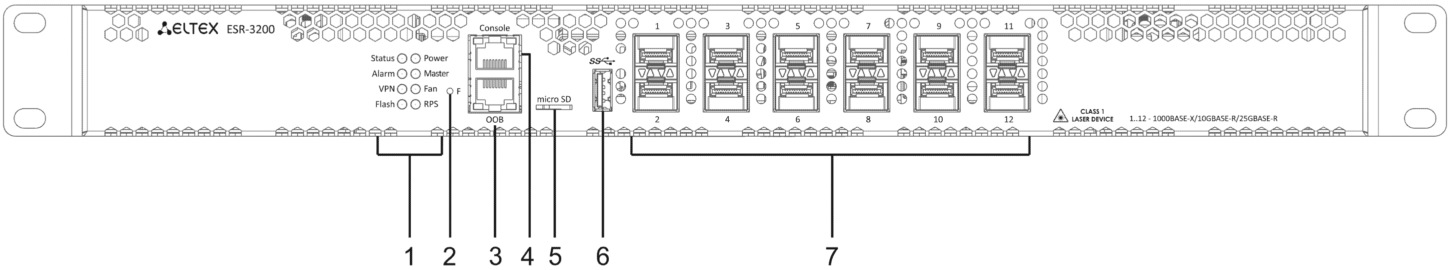


Figure 1 – ESR-3200 front panel

Table 9 lists connectors, LEDs and controls located on the front panel of ESR-3100.

Table 9 – Description of connectors, LEDs and controls located on ESR-3100 front panel

№	Front panel element	Description
1	Status	Current device status LED.
	Alarm	Alarm LED.
	VPN	VPN gateway operation mode LED.
	Flash	Activity of exchange with data storage – SD card or USB Flash.
	Power	Device power LED.
	Master	Failover mode operation LED (is not supported in the current version).
	Fan	Fan operation LED.
	RPS	Redundant power supply LED.
2	F	Functional key that reboots the device and resets it to factory default configuration: • Pressing the key for less than 10 seconds reboots the device; • Pressing the key for more than 10 seconds resets the terminal to factory settings.

№	Front panel element	Description
3	OOB	Ethernet port for router management.
4	Console	Console port RS-232 (RJ-45) for local management of the device.
5	microSD	microSD-card port.
6	USB1	USB 2.0 port for USB devices connection.
7	[1 .. 12]	Slots for installing 25G SFP28/10G SFP+/1G SFP transceivers.

ESR-3200 rear panel

The rear panel of ESR-3200 is depicted in Figure 2.

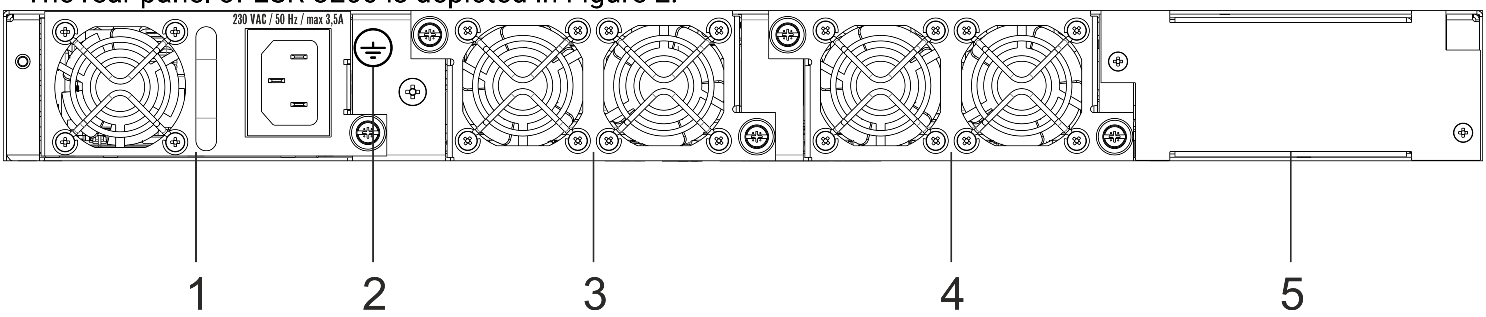


Figure 2 – ESR-3200 rear panel

Table 10 lists rear panel connectors of the router.

Table 10 – Rear panel connectors description

№	Description
1	Main power supply.
2	Earth bonding point of the device.
3	Hot-swappable removable ventilation modules.
4	
5	Place for installation of a redundant power supply.

ESR-3200 side panels

The side panel layout of ESR-3200 is depicted in figures 3 and 4.

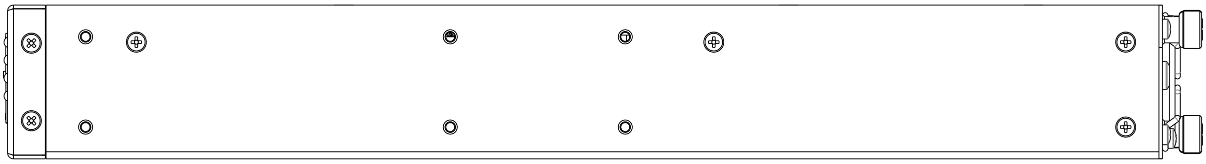


Figure 3 – ESR-3200 right side panel

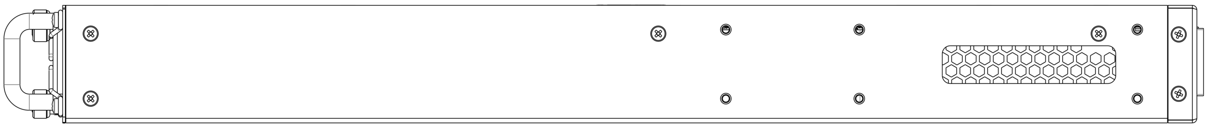


Figure 4 – ESR-3200 left side panel

Side panels of the device have air vents for heat removal. Do not block air vents. This may cause the components to overheat, which may result in device malfunction. For recommendations on device installation, see section Installation and connection.

2.4.2 ESR-3100 design

ESR-3100 front panel

The front panel layout is depicted in 5.

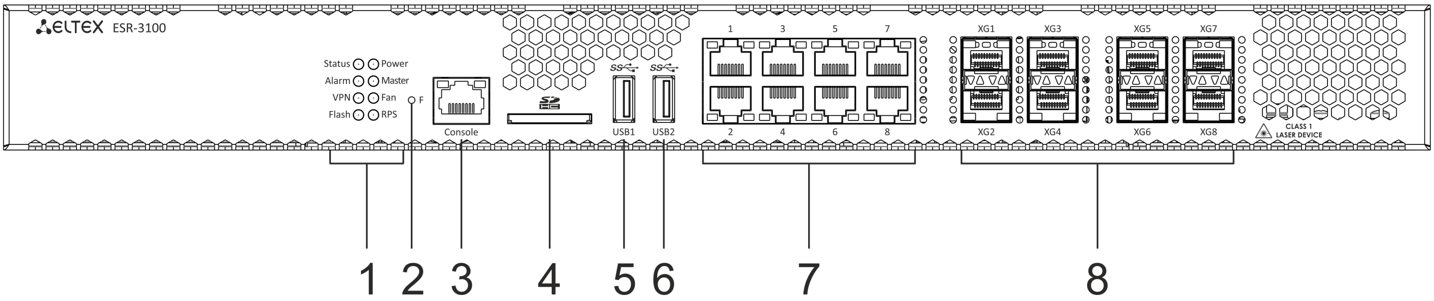


Figure 5 – ESR-3100 front panel

Table 11 lists connectors, LEDs and controls located on the front panel of ESR-3100.

Table 11 – Description of connectors, LEDs and controls located on ESR-3100 front panel

№	Front panel element	Description
1	Status	Current device status LED.
	Alarm	Alarm LED.
	VPN	VPN gateway operation mode LED.
	Flash	Activity of exchange with data storage – SD card or USB Flash.

№	Front panel element	Description
	Power	Device power LED.
	Master	Failover mode operation LED (is not supported in the current version).
	Fan	Fan operation LED.
	RPS	Redundant power supply LED.
2	F	Functional key that reboots the device and resets it to factory default configuration: • Pressing the key for less than 10 seconds reboots the device; • Pressing the key for more than 10 seconds resets the terminal to factory settings.
3	Console	Console port RS-232 (RJ-45) for local management of the device.
4	SD	SD-card connector.
5	USB1	USB 3.0 port for USB device connection.
6	USB2	USB 3.0 port for USB device connection.
7	[1 .. 8]	8 ports of Gigabit Ethernet 10/100/1000BASE-T (RJ-45).
8	XG1 – XG8	Slots for 10G SFP+/1G SFP transceivers.

ESR-3100 rear panel

The rear panel of ESR-3100 is depicted in the figure below.

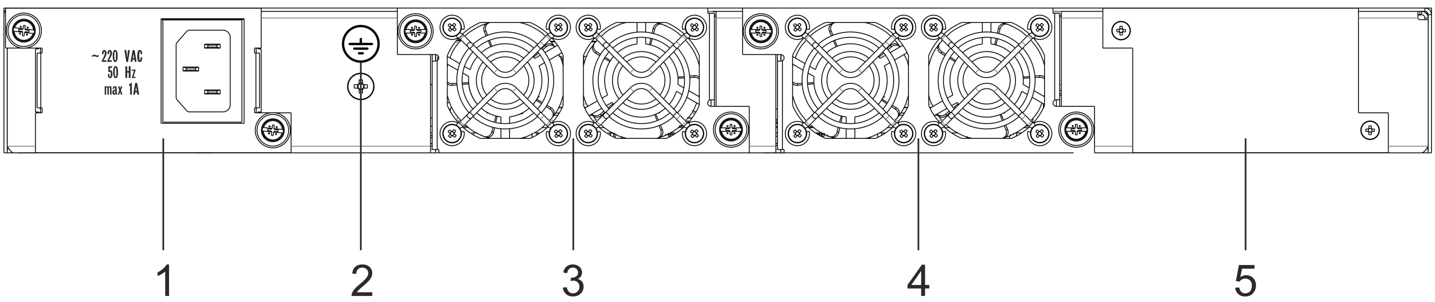


Figure 6 – ESR-3100 rear panel

Table 12 lists rear panel connectors of the router.

Table 12 – Rear panel connectors description

№	Description
1	Main power supply.
2	Earth bonding point of the device.
3	Hot-swappable removable ventilation modules.
4	
5	Place for installation of a redundant power supply.

ESR-3100 side panels

The side panel layout of ESR-3100 is depicted in figures 7 and 8.

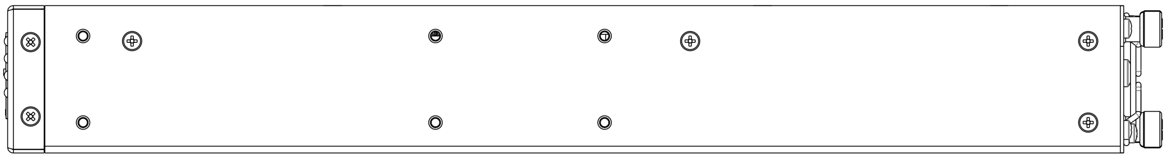


Figure 7 – ESR-3100 right side panel

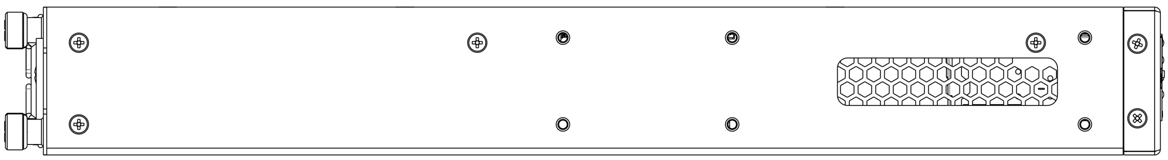


Figure 8 – ESR-3100 left side panel

Side panels of the device have air vents for heat removal. Do not block air vents. This may cause the components to overheat, which may result in device malfunction. For recommendations on device installation, see section Installation and connection.

2.4.3 ESR-1700 design

ESR-1700 front panel

The front panel layout is depicted in figure 9.

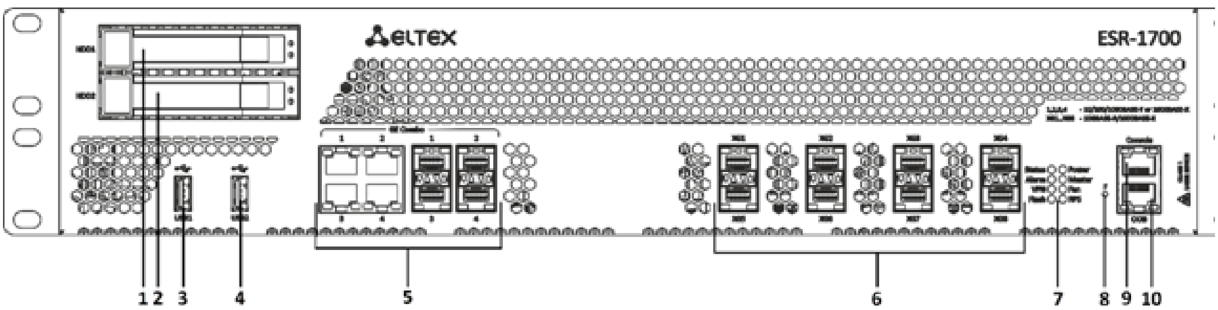


Figure 9 – ESR-1700 front panel

Table 13 lists connectors, LEDs and controls located on the front panel of ESR-1700.

Table 13 – Description of ESR-1700 connectors, LEDs and front panel controls

№	Front panel element	Description
1	HDD1	Connector for HDD installation.
2	HDD2	Connector for HDD installation.
3	USB1	Port for USB device connection.
4	USB2	Port for USB device connection.
5	Combo Ports [1 .. 4]	4 ports of Gigabit Ethernet 10/100/1000BASE-X (SFP).
6	XG1 – XG8	Slots for 10G SFP+/1G SFP transceivers.
7	Status	Current device status LED.
	Alarm	Alarm LED.
	VPN	VPN gateway operation mode LED (is not supported in the current version).
	Flash	Activity of exchange with data storage – SD card or USB Flash.
	Power	Device power LED.
	Master	Failover mode operation LED (is not supported in the current version).
	Fan	Fan operation LED.
	RPS	Redundant power supply LED.
8	F	Functional key that reboots the device and resets it to factory default configuration: • Pressing the key for less than 10 seconds reboots the device; • Pressing the key for more than 10 seconds resets the terminal to factory settings.
9	Console	Console port RS-232 for local management of the device.
10	OOB	Ethernet port for router management.

ESR-1700 rear panel

The rear panel of ESR-1700 is shown in the picture below.

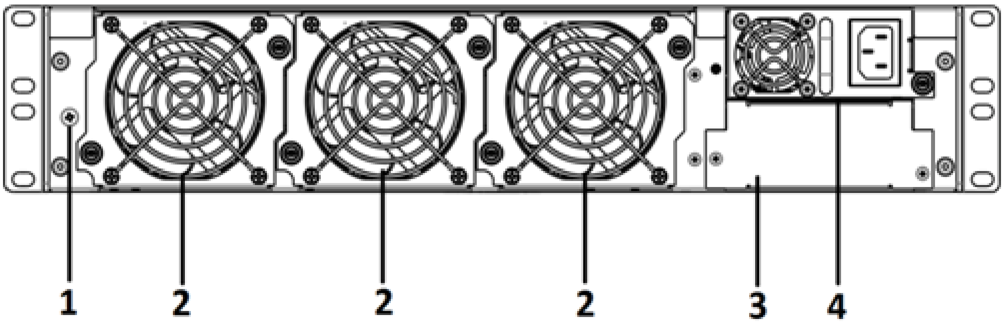


Figure 10 – ESR-1700 rear panel

Table 14 lists rear panel connectors of the router.

Table 14 – Rear panel connectors description

№	Description
1	Earth bonding point of the device.
2	Hot-swappable removable ventilation modules.
3	Main power supply.
4	Place for installation of a redundant power supply.

ESR-1700 side panels

The side panel layout of ESR-1700 is depicted in figures 3 and 4.

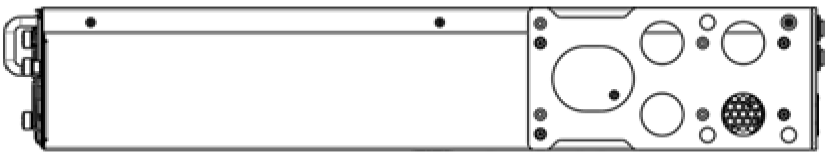


Figure 11 – ESR-1700 right side panel

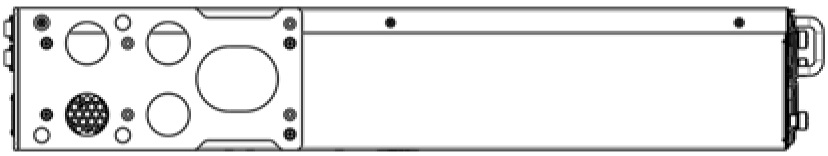


Figure 12 – ESR-1700 left side panel

Side panels of the device have air vents for heat removal. Do not block air vents. This may cause the components to overheat, which may result in device malfunction. For recommendations on device installation, see section Installation and connection.

2.4.4 ESR-1511, ESR-1510 design

ESR-1511 front panel

The front panel layout is depicted in figure 9.

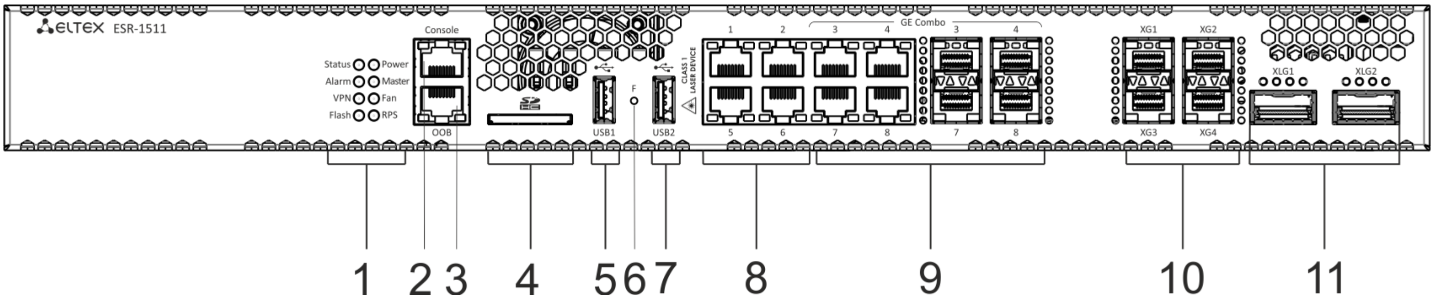


Figure 13 – ESR-1511 front panel

Table 15 lists connectors, LEDs and controls located on the front panel of ESR-1511.

Table 15 – Description of connectors, LEDs and controls located on ESR-1511 front panel

№	Front panel element	Description
1	Status	Current device status LED.
	Alarm	Alarm LED.
	VPN	VPN gateway operation mode LED (is not supported in the current version).
	Flash	Activity of exchange with data storage – SD card or USB Flash.
	Power	Device power LED.
	Master	Failover mode operation LED (is not supported in the current version).
	Fan	Fan operation LED.
	RPS	Redundant power supply LED.
2	Console	Console port RS-232 for local management of the device.
3	OOB	Ethernet port for router management.
4	SD	SD-card connector.
5	USB1	Port for USB device connection.

№	Front panel element	Description
6	F	Functional key that reboots the device and resets it to factory default configuration: • Pressing the key for less than 10 seconds reboots the device; • Pressing the key for more than 10 seconds resets the terminal to factory settings.
7	USB2	Port for USB device connection.
8	Ethernet	4 ports of Ethernet 10/100/1000BASE-T.
9	Combo Ports [1 .. 4]	4 ports of Gigabit Ethernet 10/100/1000BASE-X (SFP).
10	XG1 – XG4	Slots for 10G SFP+/1G SFP transceivers.
11	XLG1 – XLG2	Slots for 40G QSFP+ transceivers.

ESR-1500 front panel

The front panel layout is depicted in figure 14.

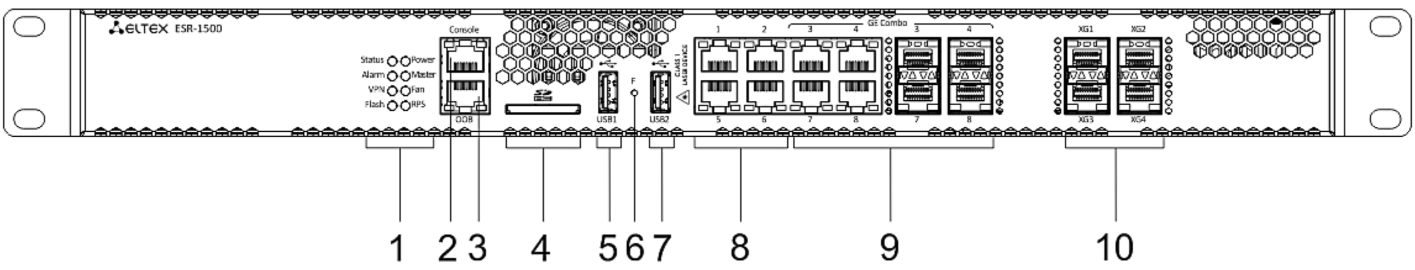


Figure 14 – ESR-1500 front panel

Table 16 lists connectors, LEDs and controls located on the front panel of ESR-1500.

Table 16 – Description of ESR-1500 connectors, LEDs and front panel controls

№	Front panel element	Description
1	Status	Current device status LED.
	Alarm	Alarm LED.
	VPN	VPN gateway operation mode LED (is not supported in the current version).
	Flash	Activity of exchange with data storage – SD card or USB Flash.

№	Front panel element	Description
	Power	Device power LED.
	Master	Failover mode operation LED (is not supported in the current version).
	Fan	Fan operation LED.
	RPS	Redundant power supply LED.
2	Console	Console port RS-232 for local management of the device.
3	OOB	Ethernet port for router management.
4	SD	SD-card connector.
5	USB1	Port for USB device connection.
6	F	Functional key that reboots the device and resets it to factory default configuration: • Pressing the key for less than 10 seconds reboots the device; • Pressing the key for more than 10 seconds resets the terminal to factory settings.
7	USB2	Port for USB device connection.
8	Ethernet	4 ports of Ethernet 10/100/1000BASE-T.
9	Combo Ports [1 .. 4]	4 ports of Gigabit Ethernet 10/100/1000BASE-X (SFP).
10	XG1 – XG4	Slots for 10G SFP+/1G SFP transceivers.

ESR-1511, ESR-1500 rear panel

The rear panel layout of ESR-1511 and ESR-1500 routers is depicted in figure 15.

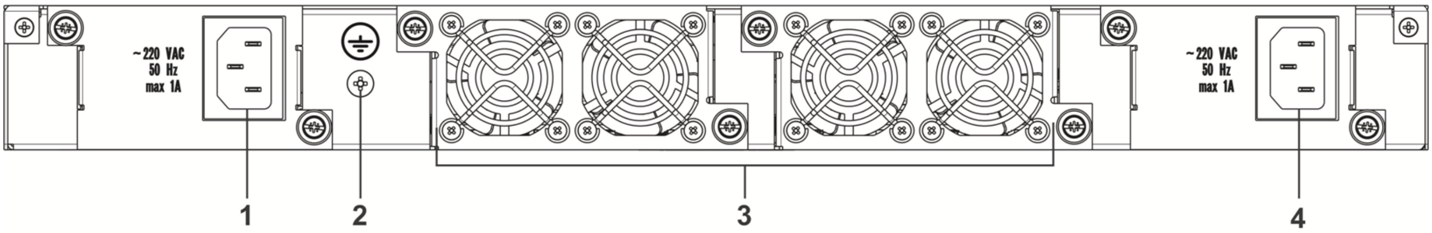


Figure 15 – ESR-1511, ESR-1500 rear panel

Table 17 lists rear panel connectors of the router.

Table 17 – Rear panel connectors description

№	Description
1	Main power supply.
2	Earth bonding point of the device.
3	Hot-swappable removable ventilation modules.
4	Place for installation of a redundant power supply.

ESR-1511, ESR-1500 side panels

The side panel layout of ESR-1511, ESR-1500 is depicted in Figures 16 and 17.

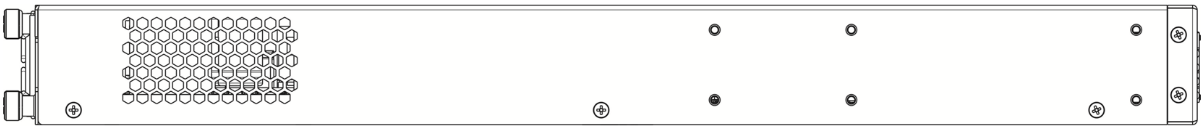


Figure 16 – ESR-1511, ESR-1500 right side panel

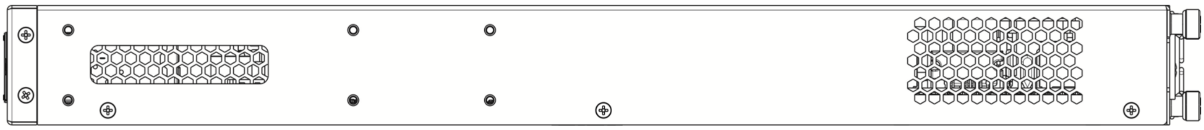


Figure 17 – ESR-1511, ESR-1500 left side panel

Side panels of the device have air vents for heat removal. Do not block air vents. This may cause the components to overheat, which may result in device malfunction. For recommendations on device installation, see section Installation and connection.

2.4.5 ESR-1200, ESR-1000 design

ESR-1200 front panel

The front panel layout is depicted in 18.

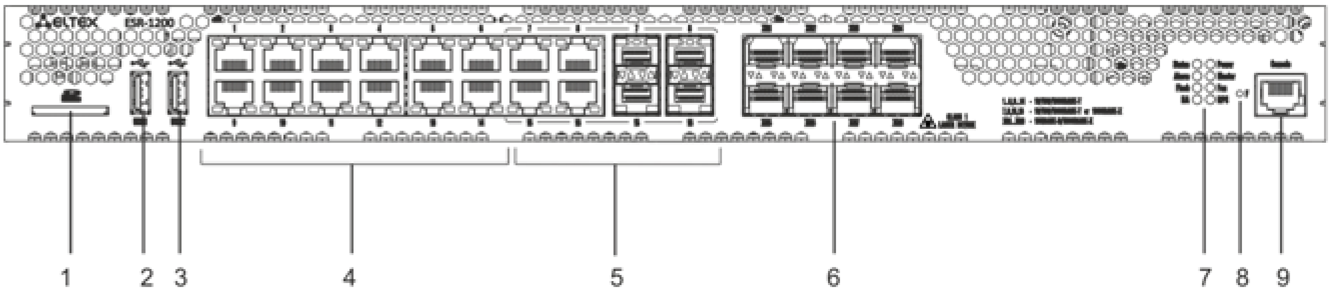


Figure 18 – ESR-1200 front panel

Table 18 lists connectors, LEDs and controls located on the front panel of ESR-1200.

Table 18 – Description of connectors, LEDs and controls located on the front panel of ESR-1200

№	Front panel element	Description
1	SD	SD-card connector.
2	USB1	Port for USB device connection.
3	USB2	Port for USB device connection.
4	[1 .. 12]	12 ports of Gigabit Ethernet 10/100/1000BASE-T (RJ-45).
5	Combo Ports	4 ports of Gigabit Ethernet 10/100/1000BASE-X (SFP).
6	XG1 – XG8	Slots for installation of 10G SFP+/1G SFP transceivers.
7	Status	Current device status LED.
	Alarm	Alarm LED.
	HA	HA operation mode LED.
	Flash	Activity indicator of exchange with data storages (SD-card or USB Flash).
	Power	Device power LED.

№	Front panel element	Description
	Master	Indicator of failover modes operation.
	Fan	Fan operation LED.
	RPS	Redundant power supply LED.
8	F	Functional key that reboots the device and resets it to factory default configuration: • Pressing the key for less than 10 seconds reboots the device; • Pressing the key for more than 10 seconds resets the terminal to factory settings.
9	Console	Console port RS-232 for local management of the device.

ESR-1000 front panel

The front panel layout is depicted in 19.

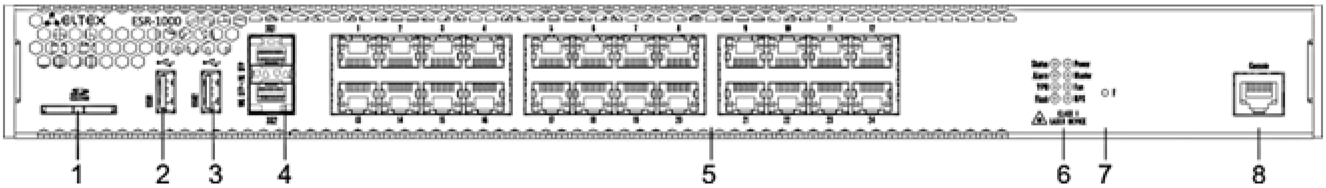


Figure 19 – ESR-1000 front panel

Table 19 lists sizes, LEDs and controls located on ESR-1000 front panel.

Table 19 – Description of connectors, LEDs and controls located on ESR-1000 front panel

№	Front panel element	Description
1	SD	SD-card connector.
2	USB1	Port for USB device connection.
3	USB2	Port for USB device connection.
4	XG1, XG2	Slots for 10G SFP+/1G SFP transceivers.
5	[1 .. 24]	24 ports of Gigabit Ethernet 10/100/1000BASE-T (RJ-45).
6	Status	Current device status LED.

№	Front panel element	Description
	Alarm	Alarm LED.
	VPN	Active VPN sessions indicator.
	Flash	Activity indicator of exchange with data storages (SD-card or USB Flash).
	Power	Device power LED.
	Master	Indicator of failover modes operation.
	Fan	Fan operation LED.
	RPS	Redundant power supply LED.
7	F	Functional key that reboots the device and resets it to factory default configuration: • Pressing the key for less than 10 seconds reboots the device; • Pressing the key for more than 10 seconds resets the terminal to factory settings.
8	Console	Console port RS-232 for local management of the device.

ESR-1200, 1000 rear panel

The rear panel of ESR-1000 is depicted in the figure below.

> ⚠ **The figure shows the router delivery package with a single AC power supply.**

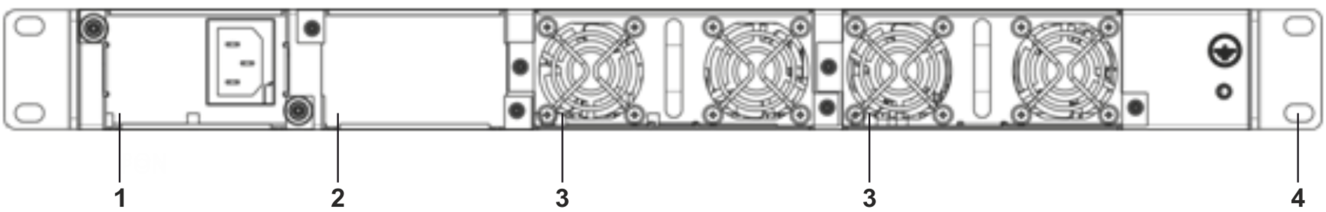


Figure 20 – ESR-1000 rear panel

Table 20 lists rear panel connectors of the router.

Table 20 – Rear panel connectors description

№	Description
1	Main power supply.
2	Place for installation of a redundant power supply.
3	Hot-swappable removable ventilation modules.
4	Earth bonding point of the device.

ESR-1200, ESR-1000 side panels

The side panel layout of ESR-1200, ESR-1000 is depicted in Figures 21 and 22.

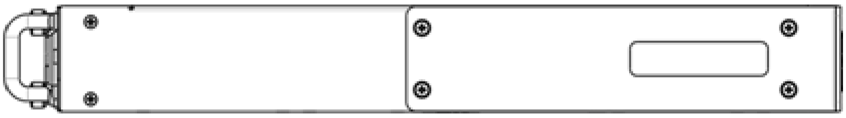


Figure 21 – ESR-1200, 1000 right side panel

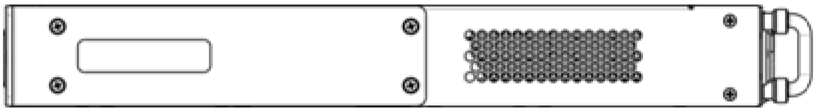


Figure 22 – ESR-1200, 1000 left side panel

Side panels of the device have air vents for heat removal. Do not block air vents. This may cause the components to overheat, which may result in device malfunction. For recommendations on device installation, see section Installation and connection.

2.4.6 ESR-200, ESR-100 design

ESR-100, ESR-200 front panel

The front panel layout of ESR-200 is depicted in figure 23.

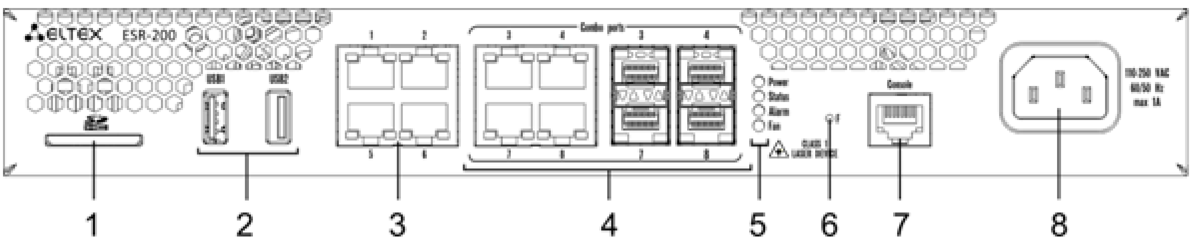


Figure 23 – ESR-200 front panel

The front panel layout of ESR-100 is depicted in figure 24.

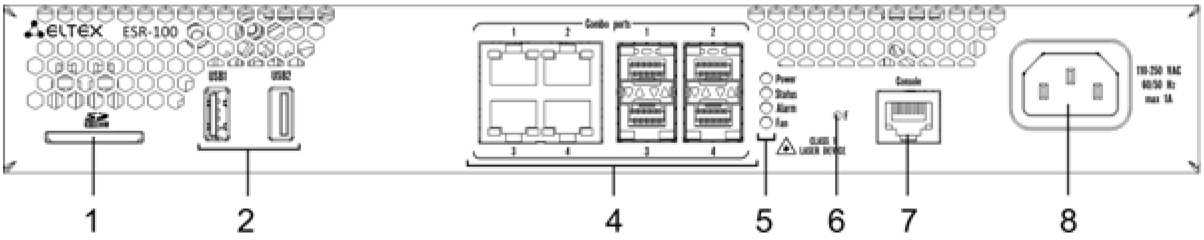


Figure 24 – ESR-100 front panel

Table 21 lists connectors, LEDs and controls located on the front panel of ESR-100 and ESR-200 routers.

Table 21 – Description of connectors, LEDs and controls located on ESR-200, ESR-100 front panel

№	**Front panel element**	**Description**
1	SD	SD-card connector.
2	USB1, USB2	2 × USB-enabled devices connection port.
3	[1 .. 4]	4 ports of Gigabit Ethernet 10/100/1000BASE-T (RJ-45).
4	Combo Ports	4 ports of Gigabit Ethernet 10/100/1000BASE-X (SFP).
5	Power	Device power LED.
	Status	Current device status LED.
	Alarm	Alarm LED.
	Fan	Fan operation LED.
6	F	Functional key that reboots the device and resets it to factory default configuration: • Pressing the key for less than 10 seconds reboots the device; • Pressing the key for more than 10 seconds resets the terminal to factory settings.
7	Console	Console port RS-232 for local management of the device.
8	110-250 VAC 60/50 Hz max 1A	Power supply.

ESR-200, ESR-100 rear panel

The rear panel layout of ESR-200 and ESR-100 routers is depicted in figure 25.

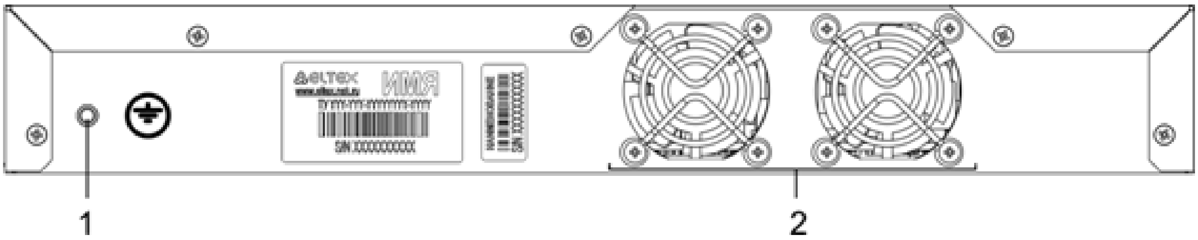


Figure 25 – ESR-200, ESR-100 rear panel

Table 22 lists rear panel connectors of the router.

Table 22 – Rear panel connectors description

№	Description
1	Earth bonding point of the device.
2	Ventilation module.

ESR-100, ESR-200 side panels

The side panel layout of ESR-200, ESR-100 is depicted in Figures 26 and 27.

Figure 26 – ESR-100 and ESR-200 right side panel

Figure 27 – ESR-100 and ESR-200 left side panel

2.4.7 ESR-21 design

The device has a metal-enclosed design for 1U 19" racks.

ESR-21 front panel

The front panel layout of ESR-21 is depicted in figure 28.

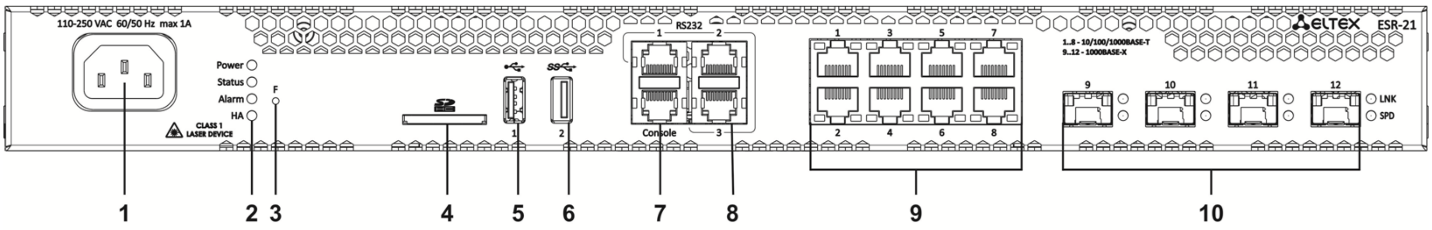


Figure 28 – ESR-21 front panel

Table 23 lists sizes, LEDs and controls located on ESR-21 front panel.

Table 23 – Description of connectors, LEDs and controls located on ESR-21 front panel

№	Front panel element	Description
1	220V AC	Power supply
2	Power	Device power LED
	Status	Device status LED
	Alarm	Device alarm presence and level LED
	HA	HA operation mode LED (is not supported in the current version)
3	F	Functional key that reboots the device and resets it to factory default configuration: - pressing the key for less than 10 seconds reboots the device. – pressing the key for more than 10 seconds resets the device to factory default configuration.
4	SD	SD-card connector
5	USB1	USB 2.0 connector for connecting external USB devices
6	USB2	USB 3.0 connector for connecting external USB devices
7	Console	Console port for local management of the device
8	RS-232	3 serial ports

№	Front panel element	Description
9	[1 .. 8]	8 ports of Gigabit Ethernet 10/100/1000BASE-T (RJ-45)
10	Optical Port	4 ports of Gigabit Ethernet 10/100/1000BASE-X (SFP)

ESR-21 rear panel

The rear panel layout of ESR-21 is depicted in figure 29.

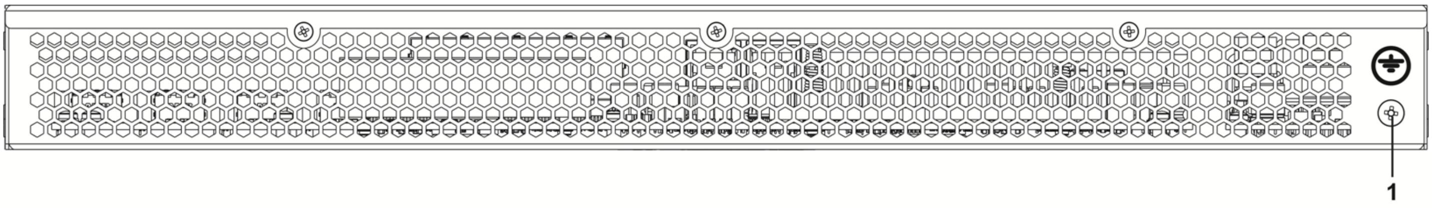


Figure 29 – ESR-21 rear panel

Table 24 lists rear panel connectors of the router.

Table 24 – Rear panel connectors description

№	Description
1	Earth bonding point of the device.

ESR-21 side panels

The side panel layout of ESR-21 is depicted in figures 30 and 31.

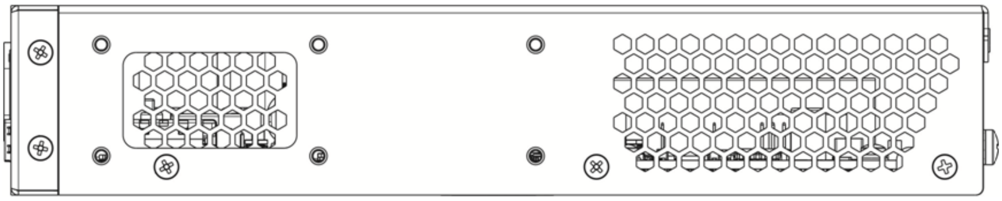


Figure 30 – ESR-21 left side panel

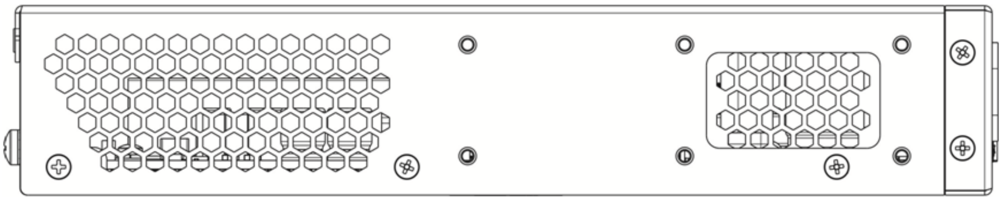


Figure 31 – ESR-21 right side panel

Side panels of the device have air vents for heat removal. Do not block air vents. This may cause the components to overheat, which may result in device malfunction. For recommendations on device installation, see section Installation and connection.

2.4.8 ESR-30, ESR-20 design

The device has a metal-enclosed design for 1U 19" racks.

ESR-30 front panel

The front panel layout is depicted in figure 32.

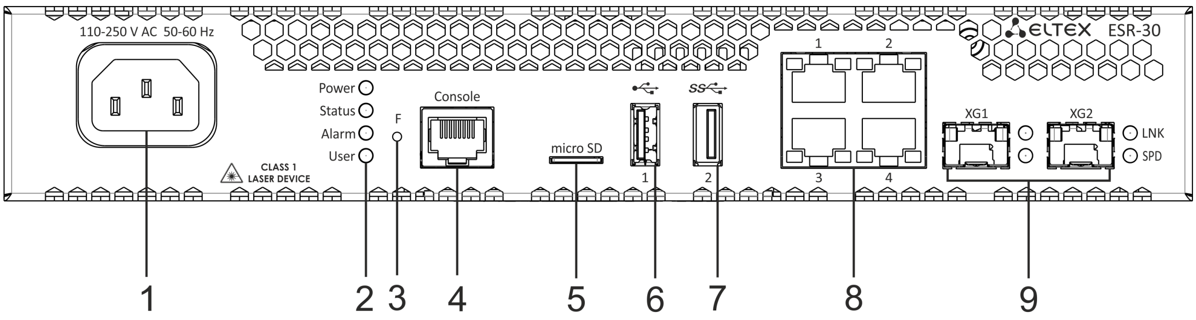


Figure 32 – ESR-20 front panel

Table 25 lists connectors, LEDs and controls located on the front panel of ESR-30.

Table 25 – Description of connectors, LEDs and controls located on ESR-30 front panel

№	Front panel element	Description
1	110-250 VAC	Power supply.
2	Power	Device power LED.
	Status	Current device status LED.
	Alarm	Alarm LED.
	User	LED for custom scenarios that can be assigned in device configuration mode.
3	F	Functional key that reboots the device and resets it to factory default configuration: - pressing the key for less than 10 seconds reboots the device. – pressing the key for more than 10 seconds resets the device to factory default configuration.
4	Console	Console port RS-232 (RJ-45) for local management of the device.
5	microSD	microSD-card connector.
6	USB1	USB 2.0 connector for connecting external USB devices.
7	USB2	USB 3.0 connector for connecting external USB devices.

№	Front panel element	Description
8	[1 .. 4]	4 ports of 10/100/1000BASE-T.
9	1, 2	2 ports of10GBASE-R (SPF+)/1000BASE-X.

ESR-20 front panel

The front panel layout is depicted in figure 33.

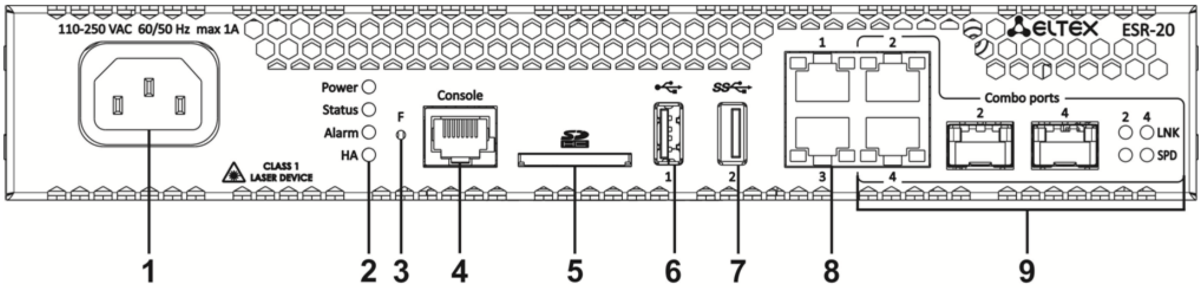


Figure 33 – ESR-20 front panel

Table 26 lists connectors, LEDs and controls located on the front panel of ESR-20.

Table 26 – Description of connectors, LEDs and controls located on ESR-20 front panel

№	Front panel element	Description
1	110-250 VAC	Power supply.
2	Power	Device power LED.
	Status	Current device status LED.
	Alarm	Alarm LED.
	HA	HA operation mode LED (is not supported in the current version).
3	F	Functional key that reboots the device and resets it to factory default configuration: - pressing the key for less than 10 seconds reboots the device. – pressing the key for more than 10 seconds resets the device to factory default configuration.
4	Console	Console port RS-232 (RJ-45) for local management of the device.
5	SD	SD-card connector.
6	USB1	USB 2.0 connector for connecting external USB devices.
7	USB2	USB 3.0 connector for connecting external USB devices.

№	Front panel element	Description
8	1, 2	2 ports of Gigabit Ethernet 10/100/1000BASE-T (RJ-45).
9	[1 .. 4]	2 Combo ports of Ethernet 10/100/1000BASE-X/10/100/1000BASE-T.

ESR-20, ESR-30 rear panel

The rear panel layout of ESR-20 and ESR-30 is depicted in figure 34.

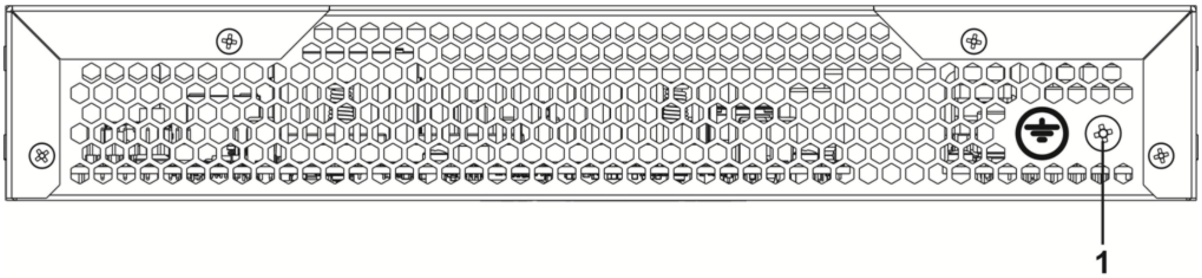


Figure 34 – ESR-20, ESR-30 rear panel

Table 27 lists rear panel connectors of the routers.

Table 27 – Rear panel connectors description

№	Description
1	Earth bonding point of the device.

ESR-20, ESR-30 side panels

The side panel layout of ESR-20 and ESR-30 is depicted in figures 35 and 36.

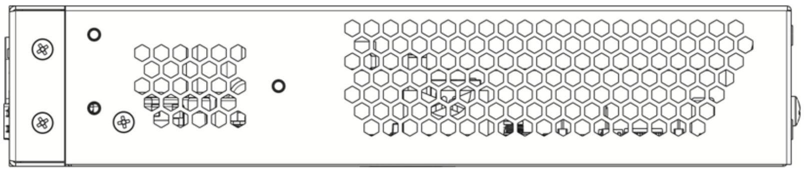


Figure 35 – ESR-20, ESR-30 left side panel

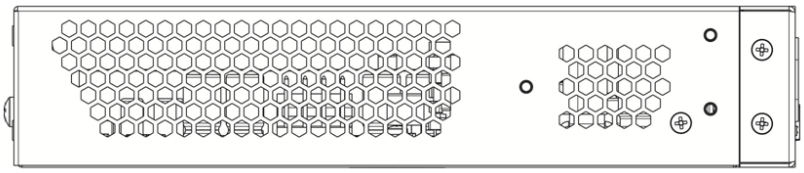


Figure 36 – ESR-20, ESR-30 right side panel

Side panels of the device have air vents for heat removal. Do not block air vents. This may cause the components to overheat, which may result in device malfunction. For recommendations on device installation, see section Installation and connection.

2.4.9 ESR-15 design

The device has a metal-enclosed design for 1U 19" racks.

ESR-15 front panel

The front panel layout is depicted in figure 37.

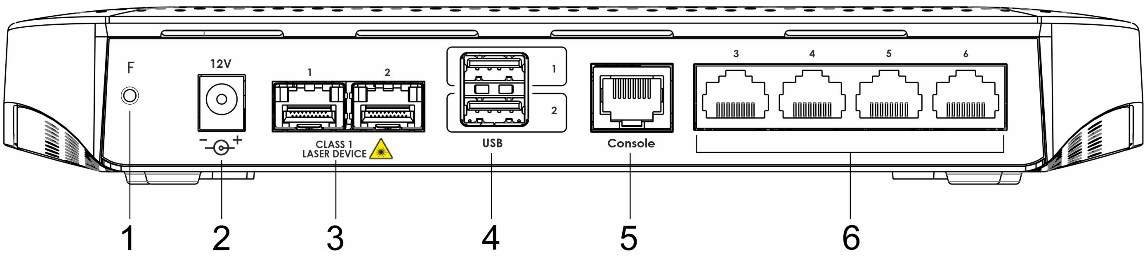


Figure 37 – ESR-15 front panel

Table 28 lists connectors, LEDs and controls located on the front panel of ESR-15 router.

Table 28 – Description of connectors, LEDs and controls located on ESR-15 front panel

№	Front panel element	Description
1	F	Functional key that reboots the device and resets it to factory default configuration: • pressing the key for less than 10 seconds reboots the device; • pressing the key for more than 10 seconds resets the device to factory default configuration.
2	12V DC	Power supply.
3	Optical ports	2 ports of Gigabit Ethernet – 1000BASE-X (SFP).
4	USB1, USB2	2 USB connectors for connecting external USB devices.
3	Console	Console port RS-232 (RJ-45) for local management of the device.
9	[1 .. 4]	4 ports of Gigabit Ethernet 10/100/1000BASE-T (RJ-45).

ESR-15 top panel

The top panel layout of ESR-10 is depicted in figure 38.

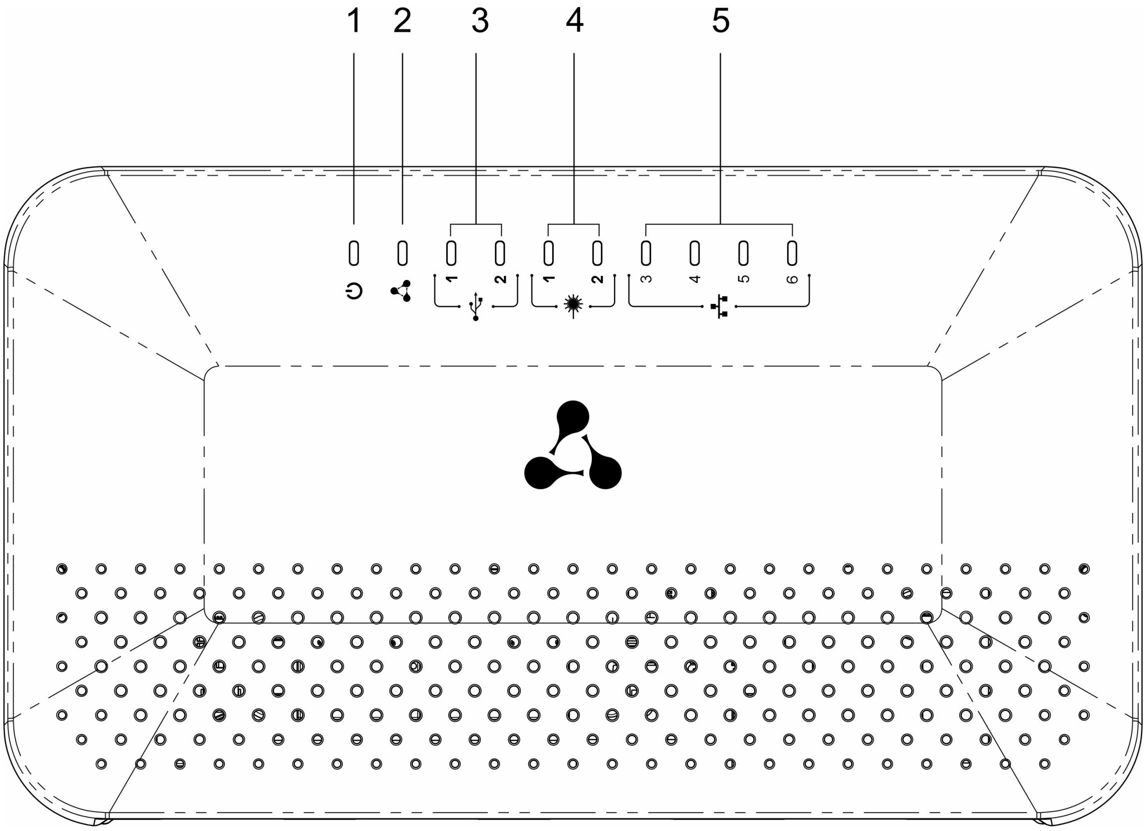


Table 29 lists LEDs located on ESR-15 top panel.

Table 29 – Description of front panel LEDs

№	Top panel element	Description
1	Power	Device power and operation status LED
2	-	The LED is not used
3	USB1, USB2	External USB devices LED
4	[1 .. 2]	Ethernet ports LED
5	[3 .. 6]	Optical interfaces LED

2.4.10 ESR-14VF, ESR-12VF design

The device has a metal-enclosed design for 1U 19" racks.

ESR-14VF, ESR-12VF front panel

The front panel layout is depicted in figure 39.

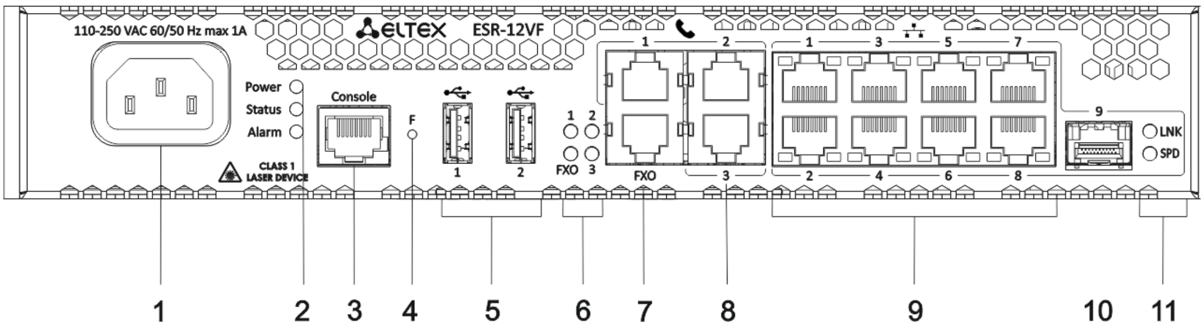


Figure 39 – ESR-14VF, ESR-12VF front panel

Table 30 lists connectors, LEDs and controls located on the front panel of ESR-14VF and ESR-12VF routers.

Table 30 – Description of connectors, LEDs and controls located on ESR-14VF, ESR-12VF front panel

№	Front panel element	Description
1	220V AC	Power supply.
2	Power	Device power LED.
3	Console	Console port RS-232 (RJ-45) for local management of the device.
4	F	Functional key that reboots the device and resets it to factory default configuration: - pressing the key for less than 10 seconds reboots the device. – pressing the key for more than 10 seconds resets the device to factory default configuration.
5	USB1, USB2	2 USB connectors for connecting external USB devices.
6	FXO	PSTN external subscriber line LED.
	1,2,3	Internal subscriber terminals LED.
7	FXO	1 FXO connector for connection PSTN external subscriber line (only for ESR-12VF).
8	FXS 1, FXS 2, FXS 3	3 connectors for internal subscriber terminals (for ESR-12VF).

№	Front panel element	Description
	FXS 1, FXS 2, FXS 3	4 connectors for internal subscriber terminals (for ESR-14VF).
9	[1 .. 8]	8 ports of Gigabit Ethernet 10/100/1000BASE-T (RJ-45).
10	Optical Port	1 port of Gigabit Ethernet-100/1000BASE-X (SFP).
11	1,2	Optical interfaces LED.

ESR-14VF, ESR-12VF rear panel

The rear panel layout of ESR-12VF, ESR-14-VF is depicted in figure 40.

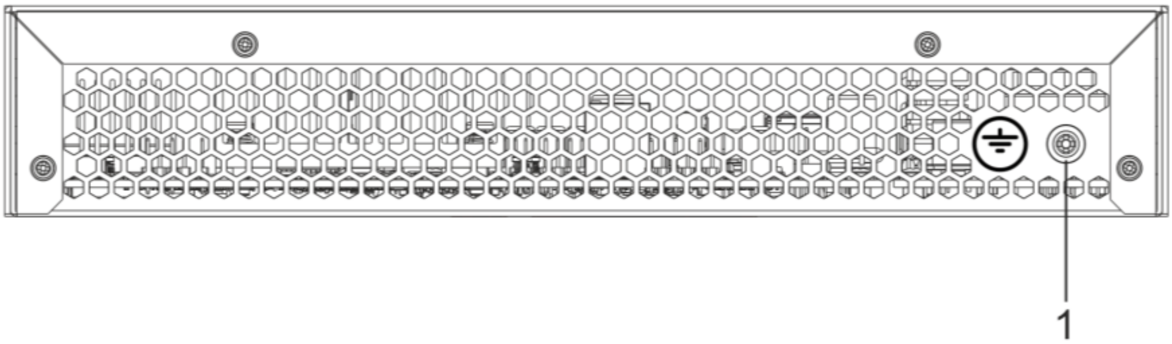


Figure 40 – ESR-12VF, ESR-14VF rear panel

Table 31 lists rear panel connectors of the router.

Table 31 – Rear panel connectors description

№	Description
1	Earth bonding point of the device.

ESR-12VF, ESR-14VF side panels

The side panel layout of ESR-12VF, ESR-14VF is depicted in Figures 41 and 42.

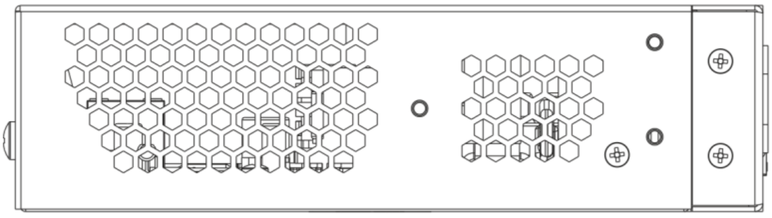


Figure 41 – ESR-12VF, ESR-14VF left side panel

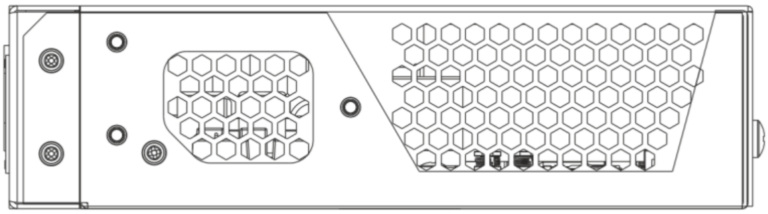


Figure 42 – ESR-12VF, ESR-14VF right side panel

Side panels of the device have air vents for heat removal. Do not block air vents. This may cause the components to overheat, which may result in device malfunction. For recommendations on device installation, see section Installation and connection.

2.4.11 ESR-12V design

The device has a metal-enclosed design for 1U 19" racks.

ESR-12V front panel

The front panel layout is depicted in figure 43.

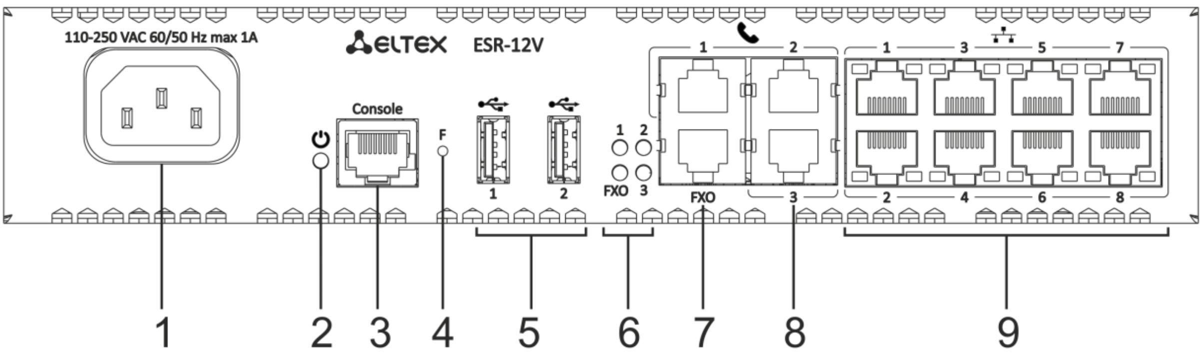


Figure 43 – ESR-12V front panel

Table 32 lists connectors, LEDs and controls located on the front panel of ESR-12VF router.

Table 32 – Description of connectors, LEDs and controls located on ESR-12V front panel

№	Front panel element	Description
1	220V AC	Power supply.
2	Power	Device power LED.
3	Console	Console port RS-232 (RJ-45) for local management of the device.
4	F	Functional key that reboots the device and resets it to factory default configuration: - pressing the key for less than 10 seconds reboots the device. – pressing the key for more than 10 seconds resets the device to factory default configuration.
5	USB1, USB2	2 USB connectors for connecting external USB devices.
6	FXO	PSTN external subscriber line LED.
	1,2,3	Internal subscriber terminals LED.
7	FXO	1 FXO connector for connection PSTN external subscriber line.
8	FXS 1, FXS 2, FXS 3	3 connectors for internal subscriber terminals.
9	[1 .. 8]	8 ports of Gigabit Ethernet 10/100/1000BASE-T (RJ-45).

ESR-12V rear panel

The rear panel layout of ESR-12V is depicted in 44.

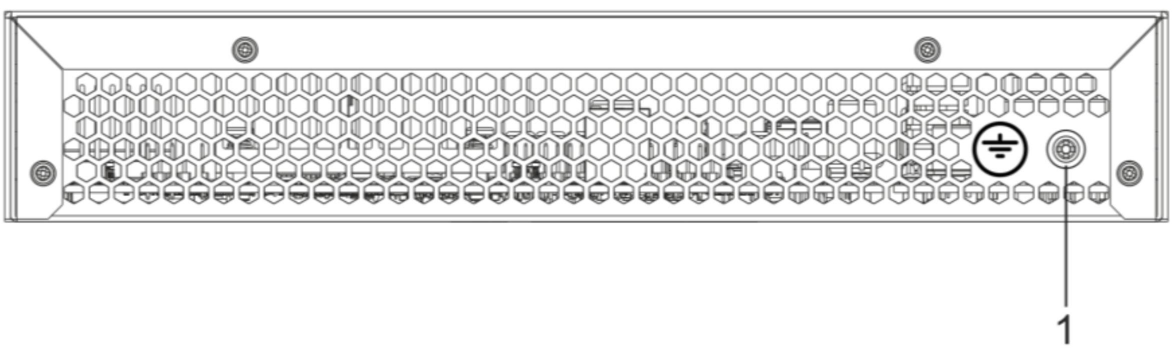


Figure 44 – ESR-12V rear panel

Table 33 lists rear panel connectors of the router.

Table 33 – Rear panel connectors description

№	Description
1	Earth bonding point of the device.

ESR-12V side panels

The side panel layout of ESR-12V is depicted in figures 45 and 46.

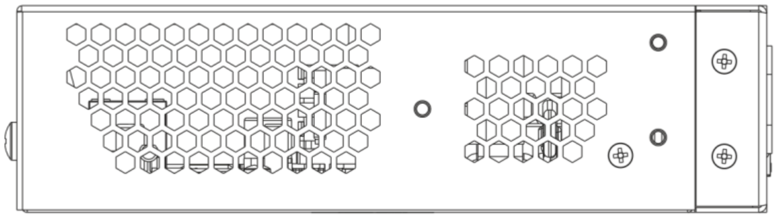


Figure 45 – ESR-12V left side panel

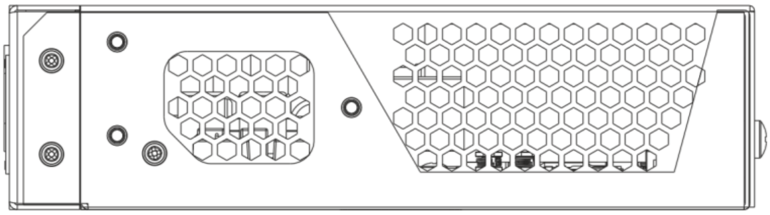


Figure 46 – ESR-12V right side panel

Side panels of the device have air vents for heat removal. Do not block air vents. This may cause the components to overheat, which may result in device malfunction. For recommendations on device installation, see section Installation and connection.

2.4.12 ESR-10 design

ESR-10 rear panel

The rear panel layout of the device is depicted in figure 47.

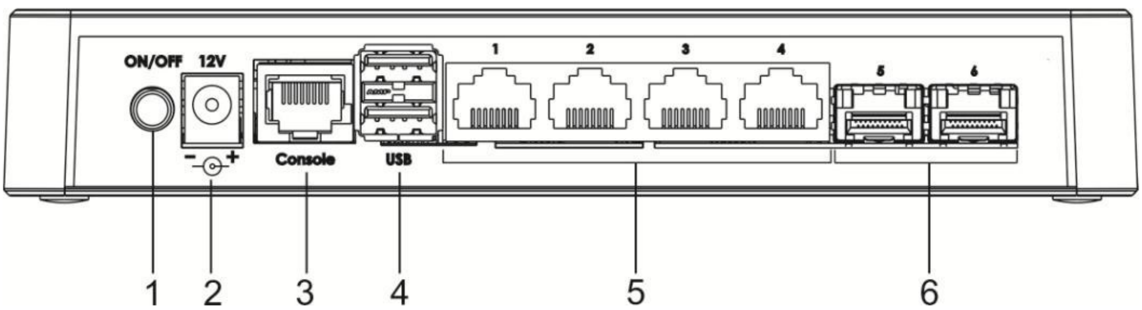


Figure 47 – ESR-10 rear panel

Table 34 lists connectors, LEDs and controls located on the rear panel of ESR-10.

Table 34 – Description of connectors, LEDs and controls located on ESR-10 rear panel

№	Front panel element	Description
1	ON/OFF	Power on/off button.
2	12V DC	Connector for power adapter connection.
3	Console	Console port RS-232 (RJ-45) for local management of the device.
4	USB1, USB2	2 USB connectors for connecting external USB devices.
5	[1.. 4]	4 ports of Gigabit Ethernet – 10/100/1000BASE-T (RJ-45).
6	Optical Ports	2 ports of Gigabit Ethernet-100/1000BASE-X (SFP).

ESR-10 side panels

The side panel layout of ESR-10 is depicted in figure 48.

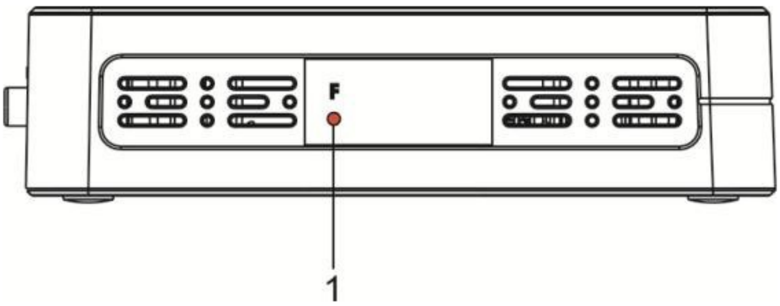


Figure 48 – ESR-10 side panel

Table 35 lists right panel controls of the router.

Table 35 – Right panel connectors description

№	Side panel element	Description
1	F	Functional key that reboots the device and resets it to factory default configuration: • pressing the key for less than 10 seconds reboots the device. • pressing the key for more than 10 seconds resets the device to factory default configuration.

ESR-10 top panel

The top panel layout of ESR-10 is depicted in figure 49.

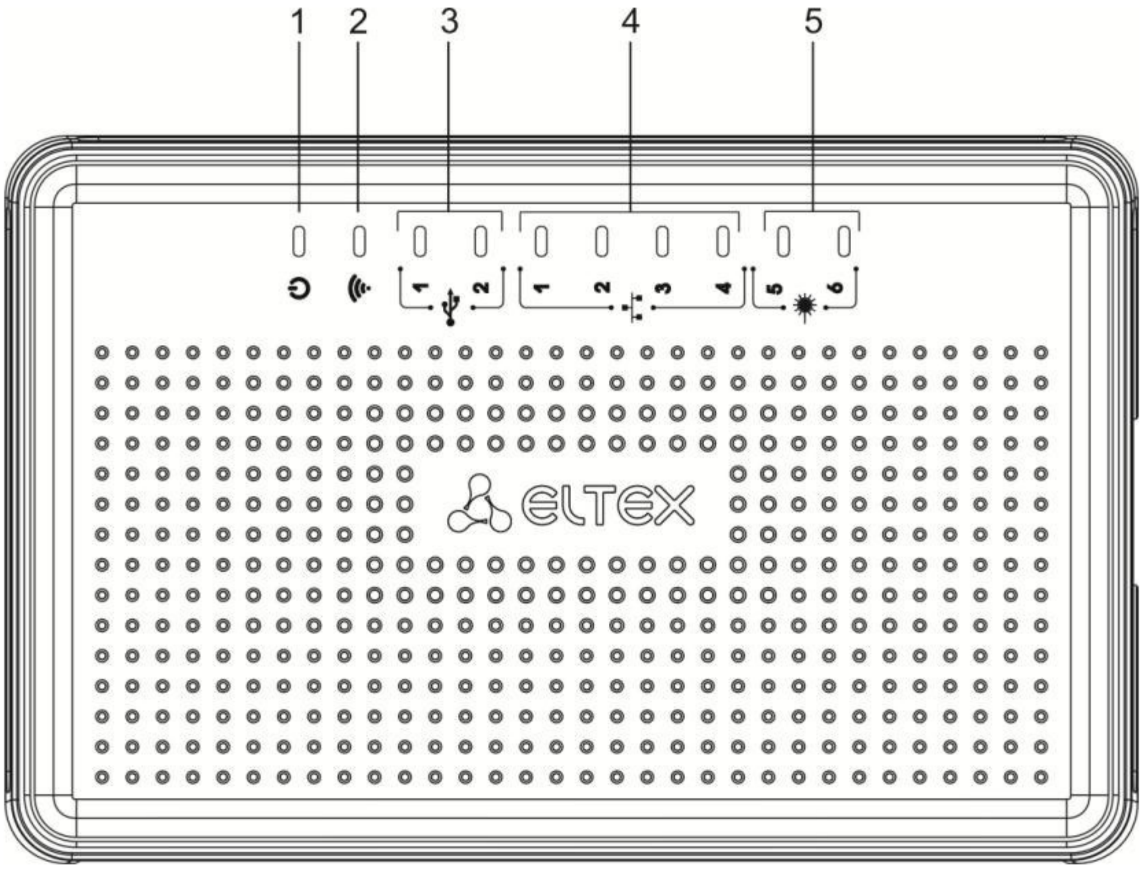


Figure 49 – ESR-10 top panel

Table 36 lists LEDs located on ESR-10 top panel.

Table 36 – Description of front panel LEDs

№	Top panel element	Description
1	Power	Device power and operation status LED.
2	-	The LED is not used.
3	USB1, USB2	External USB devices LED.
4	[1 .. 4]	Ethernet ports LED.
5	[5 .. 6]	Optical interfaces LED.

2.4.13 Light Indication

ESR-1700, ESR-1200, ESR-1000 light indication

Gigabit Ethernet copper interface statuses are represented by two LEDs – green *LINK/ACT* LED and amber *SPEED* LED. Location of the copper interface LEDs is depicted in figure 50. SFP interface status is represented by two LEDs – RX/ACT and TX/ACT – depicted in figure 51. For light indication meaning, see Tables 37 and 38 respectively.

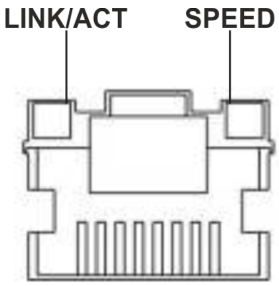


Figure 50 – Location of RJ-45 connector indicators

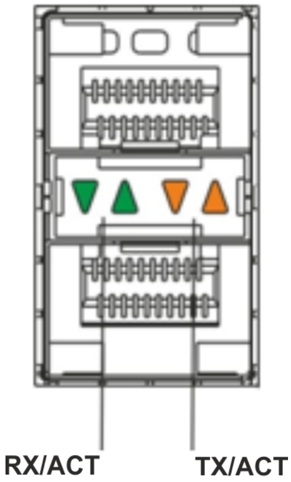


Figure 51 – Location of optical interface indicators

Table 37 – Light indication of copper interface status

SPEED indicator is lit	LINK/ACT indicator is lit	Ethernet interface state
Off	Off	The port is disabled or connection is not established.
Off	Solid on	10Mbps or 100Mbps connection is established.
Solid on	Solid on	1000Mbps connection is established.
X	Flashes	Data transfer is in progress.

Table 38 – Light indication of SFP/SFP+/QSFP+ interface status

RX/ACT indicator is lit	TX/ACT indicator is lit	Ethernet interface state
Off	Off	The port is disabled or connection is not established.
Solid on	Solid on	Connection established.
Flashes	X	Data reception in progress.
X	Flashes	Data transfer is in progress.

The following table lists description of system indicator statuses and meanings.

Table 39 – Status of system indicators

Indicator name	Indicator function	LED State	Device State
Status	Current device status LED.	Green	Device is in normal operation state.
		Red	The device is in the firmware download state.
Alarm	Alarm LED.	-	-
VPN	Active VPN sessions indicator.	-	-
Flash	Activity indicator of exchange with data storages: SD-card or USB Flash.	Green	Read/write operation execution with 'copy' command.

Indicator name	Indicator function	LED State	Device State
Power	Device power LED.	Green	Device power is normal. Main power supply, if installed, is operational.
		Orange	Main power supply failure, fault, or the primary network is missing.
		Off	Device internal power supply failure.
Master	Indicator of failover modes operation.	-	-
Fan	Cooling fan status.	Off	All fans are operational.
		Red	One or more fans has failed. Possible cause of failure: at least one of the fans has stopped or is working at lower rpm.
RPS	Backup power supply operation mode.	Green	Backup power supply is installed and operational.
		Off	Backup power supply is not installed.
		Red	Backup power supply is missing or failed.

ESR-3200, ESR-3100, ESR-1511, ESR-1500 light indication

Gigabit Ethernet copper interface and SFP interface statuses are represented by two LEDs – green *LINK/ACT* LED and amber *SPEED* LED. Location of the copper interface LEDs is depicted in figure 52. SFP interface status is depicted in figure 53. For light indication meaning, see Tables 40 and 41 respectively.

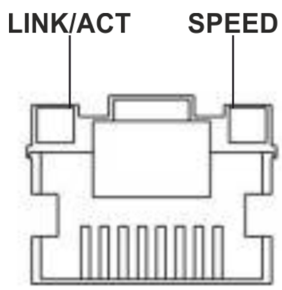


Figure 52 – Location of RJ-45 connector indicators

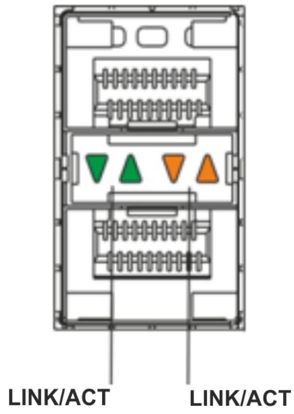


Figure 53 – Location of optical interface indicators

Table 40 – Light indication of copper interfaces status

SPEED indicator is lit	LINK/ACT indicator is lit	Ethernet interface state
Off	Off	The port is disabled or connection is not established.
Off	Solid on	10 Mbps or 100 Mbps connection is established.
Solid on	Solid on	1000 Mbps connection is established.
X	Flashes	Data transfer is in progress.

Table 41 – Light indication of SFP/SFP+/QSFP+ status

SPEED indicator is lit	LINK/ACT indicator is lit	Ethernet interface state
Off	Off	The port is disabled or connection is not established.
Solid on	Solid on	Connection is established.
Flashes	X	Data reception is in progress.
X	Flashes	Data transfer is in progress.

The following table describes the states of the system LEDs on the device and their meanings.

Table 42 – Status of system indicators

Indicator name	Indicator function	LED State	Device State
Status	Current device status LED.	Green	Device is in normal operation state.
		Orange	Device is booting up the software.

Indicator name	Indicator function	LED State	Device State
Alarm	Device alarm presence and level indicator.	-	-
VPN	Active VPN sessions indicator.	-	-
Flash	Activity indicator of exchange with data storages: SD-card or USB Flash.	Orange	Read/write operation execution with 'copy' command.
Power	Device power LED.	Green	Device power is OK. Main power supply, if installed, is operational.
		Orange	Main power supply failure, fault, or the primary network is missing.
		Off	Device internal power supply failure.
Master	Indicator of failover modes operation.	-	-
Fan	Cooling fan status.	Off	All fans are operational.
		Red	One or more fans has failed. Possible cause of failure: at least one of the fans has stopped or is working at lower rpm.
RPS	Backup power supply operation mode.	Green	Backup power supply is installed and operational.
		Off	Backup power supply is not installed.
		Red	Backup power supply is missing or failed.

ESR-200/ESR-100 light indication

Gigabit Ethernet copper interface and SFP interface statuses are represented by two LEDs – green *LINK/ACT* LED and amber *SPEED* LED. Location of the copper interface LEDs is depicted in figure 50. SFP interface status is depicted in figure 54. For light indication meaning, see Table 43.

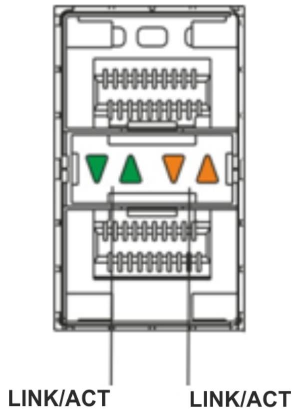


Figure 54 – Location of optical interface indicators

Table 43 – Light indication of copper and SFP interfaces status

SPEED indicator is lit	LINK/ACT indicator is lit	Ethernet interface state
Off	Off	The port is disabled or connection is not established.
Off	Solid on	10 Mbps or 100 Mbps connection is established.
Solid on	Solid on	1000 Mbps connection is established.
X	Flashes	Data transfer is in progress.

The following table describes the states of the system LEDs on the device and their meanings.

Table 44 – Status of system indicators

Indicator name	Indicator function	LED State	Device State
Status	Current device status LED.	Green	Device is in normal operation state.
		Red	Device is booting up the software.
Alarm	Device alarm presence and level indicator[1].	-	-

Indicator name	Indicator function	LED State	Device State
Power	Device power LED.	Green	Device power is OK. Main power supply, if installed, is operational.
		Red	Main power supply failure, fault, or the primary network is missing.
		Off	Device internal power supply failure.
Fan	Cooling fan status.	Off	All fans are operational.
		Red	One or more fans has failed. Possible cause of failure: at least one of the fans has stopped or is working at lower rpm.

[1] Not supported in current firmware version.

ESR-21/ESR-20 light indication

Gigabit Ethernet copper interface statuses are represented by two LEDs – green *LINK/ACT* LED and amber *SPEED* LED.

Table 45 – Light indication of copper and SFP interfaces status

SPEED indicator is lit	LINK/ACT indicator is lit	Ethernet interface state
Off	Off	The port is disabled or connection is not established.
Off	Solid on	10 Mbps or 100 Mbps connection is established.
Solid on	Solid on	1000 Mbps connection is established.
X	Flashes	Data transfer is in progress.

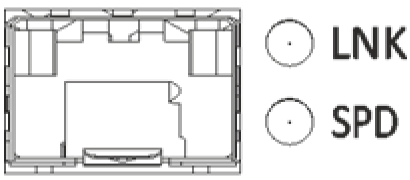


Figure 55 – Location of SFP connector indicators

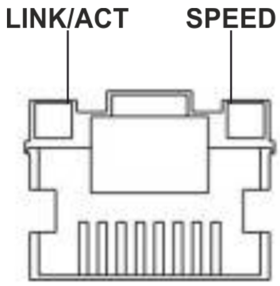


Figure 56 – Location of RJ-45 connector indicators

The following table lists description of system indicator statuses and meanings.

Table 46 – Status of system indicators

Indicator name	Indicator function	LED State	Device State
Power	Device power LED.	Green	Device power is normal operation state. Main power supply, if installed, is operational. The main software is uploaded.
		Red	The main software is not loaded.
		Off	Device internal power supply failure.
Status	Current device status LED.	Green	Device is in normal operation state.
		Orange	Device is booting up the software.
Alarm	Alarm LED.	-	-
HA	HA operation mode LED (not supported in the current version)	-	-

ESR-30 light indication

Gigabit Ethernet copper interface statuses are represented by two LEDs – green *LINK/ACT* LED and amber *SPEED* LED.

Table 47 – Light indication of copper and SFP interfaces status

SPEED indicator is lit	LINK/ACT indicator is lit	Ethernet interface state
Off	Off	The port is disabled or connection is not established.
Off	Solid on	10 Mbps or 100 Mbps connection is established.
Solid on	Solid on	1000 Mbps connection is established.
X	Flashes	Data transfer is in progress.

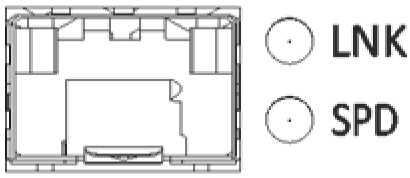


Figure 57 – Location of SFP connector indicators

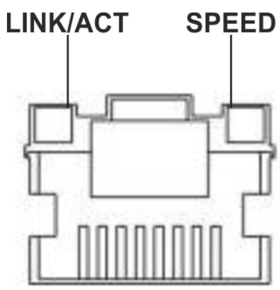


Figure 58 – Location of RJ-45 connector indicator

The following table lists description of system indicator statuses and meanings.

Table 48 – Status of system indicators

Indicator name	Indicator function	LED State	Device State
Power	Device power LED.	Green	Device power is normal operation state. Main power supply, if installed, is operational. The main software is uploaded.
		Red	The main software is not loaded.
		Off	Device internal power supply failure.

Indicator name	Indicator function	LED State	Device State
Status	Current device status LED.	Green	Device is in normal operation state.
		Flashes green	Device is booting up the software.
Alarm	Alarm LED.	-	-
HA	HA operation mode LED (not supported in the current version)	-	-

ESR-15 light indication

Gigabit Ethernet copper interfaces statuses are represented by amber *SPEED* LED.

Table 49 – Light indication of copper interface status

SPEED indicator is lit	Ethernet interface state
Off	Port is disabled or connection is not established
Solid amber on	1000 Mbps connection is established
Solid green on	10 Mbps or 100 Mbps connection is established
Flashes	Data transfer is in progress

The following table lists description of system indicator statuses and meanings.

Table 50 – Status of system indicators

Indicator name	Indicator function	LED State	Device State
Power	Device power LED.	Green	Device power is normal operation state. Main power supply, if installed, is operational. The main software is uploaded.
		Red	The main software is not loaded.
		Off	Device internal power supply failure.
USB1, USB2	Indicators of operation of external USB devices.	Green	USB device is connected.
		Flashes green	Read/write operation execution.

Indicator name	Indicator function	LED State	Device State
		Off	No devices connected or connectivity issues.

ESR-12V(F) light indication

Gigabit Ethernet copper interface statuses are represented by two LEDs – green LINK/ACT LED and amber SPEED LED.

Table 51 – Light indication of copper and SFP interface status

SPEED indicator is lit	LINK/ACT indicator is lit	Ethernet interface state
Off	Off	Port is disabled or connection is not established
Off	Solid on	10 Mbps or 100 Mbps connection is established
Solid on	Solid on	1000 Mbps connection is established
X	Flashes	Data transfer is in progress

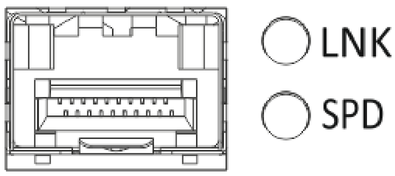


Figure 57 – Location of SFP connector indicators (only for ESR-12VF, ESR-14VF)

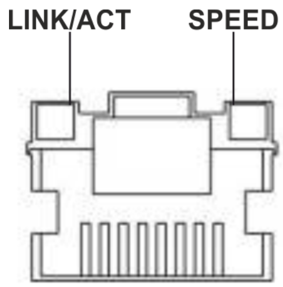


Figure 58 – Location of RJ-45 connector indicators

The following table lists description of system indicator statuses and meanings.

Table 52 – Status of system indicators

Indicator name	Indicator function	LED State	Device State
Power	Device power LED.	Green	Device power is normal. Main power supply, if installed, is operational. The main software is uploaded.
		Red	The main software is not uploaded.

Indicator name	Indicator function	LED State	Device State
		Off	Device internal power supply failure.

ESR-10 light indication

Gigabit Ethernet copper interfaces statuses are represented by amber SPEED LED.

Table 53 – Light indication of copper interface status

SPEED indicator is lit	Ethernet interface state
Off	Port is disabled or connection is not established
Solid amber on	1000 Mbps connection is established
Solid green on	10 Mbps or 100 Mbps connection is established
Flashes	Data transfer is in progress

Table 54 – Status of system indicators

Indicator name	Indicator function	LED State	Device State
Power	Device power LED.	Green	Device power is normal operation state. Main power supply, if installed, is operational. The main software is uploaded.
		Red	The main software is not loaded.
		Off	Device internal power supply failure.
USB1, USB2	Indicators of operation of external USB devices.	Green	USB device is connected.
		Flashes green	Read/write operation execution.
		Off	No devices connected or connectivity issues.

2.5 Delivery package

ESR-10 standard delivery package includes:

- ESR-10 router;
- 220 VAC/12 VDC, 1.5 A power adapter;
- Conformity certificate;
- Documentation (optional).

ESR-12V standard delivery package includes:

- ESR-12V router;
- Power cable;
- Console cable;
- 19” rack mounting kit;
- Conformity certificate;
- Documentation (optional).

ESR-12VF standard delivery package includes:

- ESR-12VF router;
- Power cable;
- Console cable;
- 19” rack mounting kit;
- Conformity certificate;
- Documentation (optional).

ESR-14VF standard delivery package includes:

- ESR-14VF router;
- Power cable;
- 19” rack mounting kit;
- Conformity certificate;
- Documentation (optional).

ESR-15 standard delivery package includes:

- ESR-15 router;
- 220 VAC/12 VDC, 1.5 A power adapter;
- Conformity certificate;
- Documentation (optional).

ESR-20 standard delivery package includes:

- ESR-20 router;
- Power cable;
- Console cable;
- 19” rack mounting kit;
- Conformity certificate;
- Documentation (optional).

ESR-21 standard delivery package includes:

- ESR-21 router;
- Power cable;
- Console cable;
- 19” rack mounting kit;
- Conformity certificate;
- Documentation (optional).

ESR-30 standard delivery package includes:

- ESR-21 router;
- Power cable;
- Console cable;
- 19” rack mounting kit;
- Conformity certificate;
- Documentation (optional).

ESR-100 standard delivery package includes:

- ESR-100 router;
- Power cable;
- Console cable;
- 19” rack mounting kit;
- Conformity certificate;
- Documentation (optional).

ESR-200 standard delivery package includes:

- ESR-200 router;
- Power cable;
- Console cable;
- 19” rack mounting kit;
- Conformity certificate;
- Documentation (optional).

ESR-1000 standard delivery package includes:

- ESR-1000 router;
- Console cable;
- 19” rack mounting kit;
- Conformity certificate;
- Documentation (optional).

ESR-1200 standard delivery package includes:

- ESR-1200 router;
- Console cable;
- 19” rack mounting kit;
- Conformity certificate;
- Documentation (optional).

ESR-1500 standard delivery package includes:

- ESR-1500 router;
- Console cable;
- 19” rack mounting kit;
- Conformity certificate;
- Documentation (optional).

ESR-1511 standard delivery package includes:

- ESR-1511 router;
- Console cable;
- 19” rack mounting kit;
- Conformity certificate;
- Documentation (optional).

ESR-1700 standard delivery package includes:

- ESR-1700 router;
- Console cable;
- 19” rack mounting kit;
- Conformity certificate;
- Documentation (optional).

ESR-3100 standard delivery package includes:

- ESR-3100 router;
- Console cable;
- 19” rack mounting kit;
- Conformity certificate;
- Documentation (optional).

ESR-3200 standard delivery package includes:

- ESR-3200 router;
- Console cable;
- 19” rack mounting kit;
- Conformity certificate;
- Documentation (optional).

Power module (PM-160-220/12 or PM-100-48/12) may be included in the ESR-1000, ESR-1200 delivery package on the customer's request.
Power module (PM-160-220/12) may be included in the ESR-1500, ESR-1511, ESR-3100, ESR-3200 delivery package on the customer's request.
Power module (PM-350-220/12 or PM-350-48/12) may be included in the ESR-1700 delivery package on the customer's request.

SFP/SFP+ transceivers may be included in the delivery package on the customer's request.

3 Installation and connection

This section describes installation of the device into a rack and connection to a power supply.

3.1 Support brackets mounting

The delivery package includes support brackets for rack installation and mounting screws to fix the device case on the brackets. To install the support brackets:

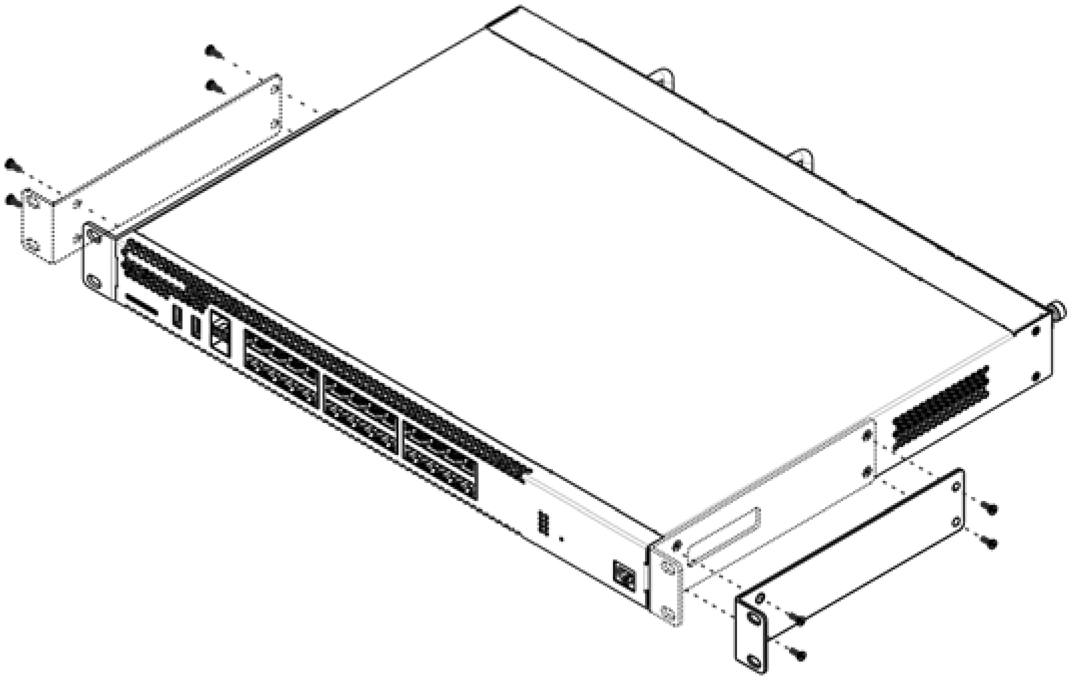


Figure 60 – Support brackets mounting

1. Align four mounting holes in the support bracket with the corresponding holes in the side panel of the device.
2. Use a screwdriver to screw the support bracket to the case.
3. Repeat steps 1 and 2 for the second support bracket.

3.2 Device rack installation

To install the device to the rack:

1. Attach the device to the vertical guides of the rack.
2. Align mounting holes in the support bracket with the corresponding holes in the rack guides. Use the holes of the same level on both sides of the guides to ensure the device horizontal installation.
3. Use a screwdriver to screw the router to the rack.

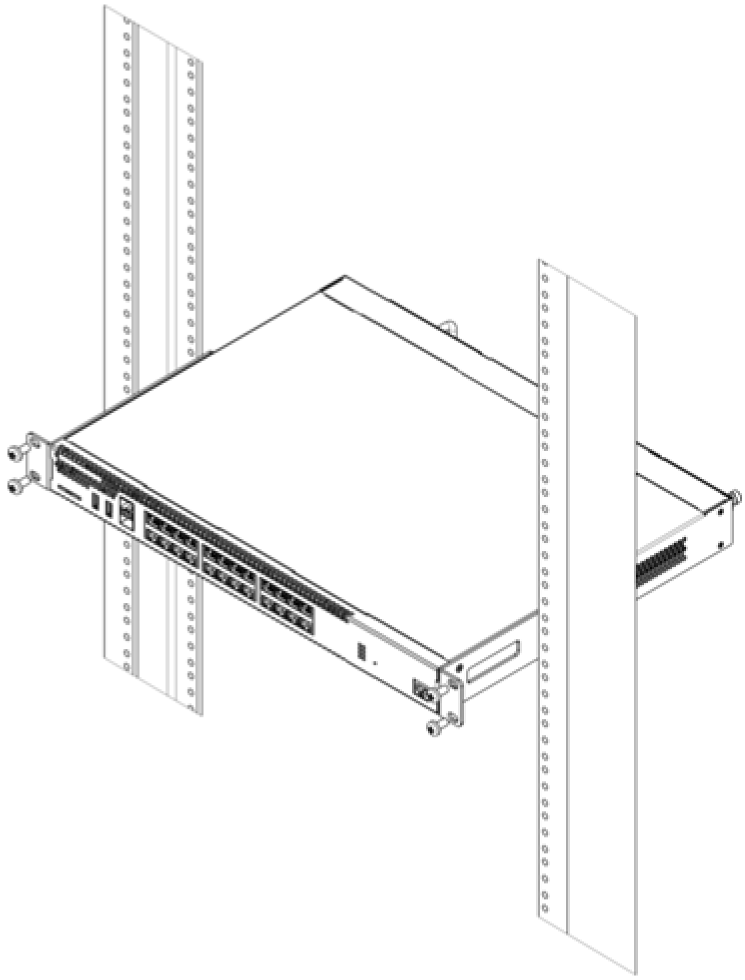


Figure 62 – Device rack installation

Device ventilation system is implemented using 'front-rear' layout. Vents are located on the front and side panels of the device; ventilation modules are located at the rear. Do not block air inlet and outlet vents to avoid components overheating and subsequent device malfunction.

3.3 ESR-1000, ESR-1200, ESR-1500, ESR-1511, ESR-1700, ESR-3100, ESR-3200 power module installation

ESR-1000/1200/1500/1511/1700/3100/3200 routers can operate with one or two power modules. The second power module installation is necessary when the device operates under strict reliability requirements.

From the electric point of view, both places for power module installation are identical. In the context of device operation, the power module can be in the main and reserve slot. For information on priority see table 'Description of connectors, LEDs and controls located on router'. Power modules can be inserted and removed without powering the device off. When additional power module is inserted or removed, the router continues operation without reboot.

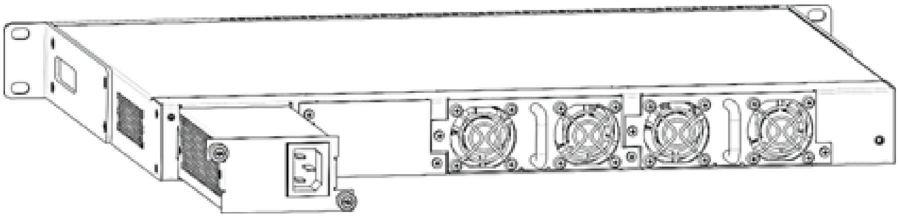


Figure 63 – Power module installation

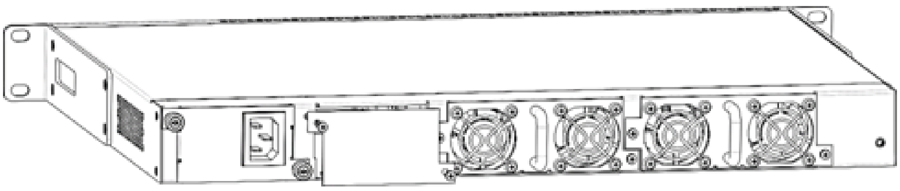


Figure 64 – Plug installation

Power module fault indication may be caused not only by the module failure, but also by the absence of the primary power supply.

The state of power modules can be checked by the indication on the front panel of the router (see Light indication section) or by diagnostics, available through the router management interfaces.

3.4 Connection to Power Supply

1. Ground the case of the device prior to connecting it to the power supply. An insulated multiconductor wire should be used for earthing. The device grounding and the earthing wire cross-section should comply with Electric Installation Code.
2. If a PC or another device is supposed to be connected to the router console port, the device should be also securely grounded.
3. Connect the power supply cable to the device. Depending on the delivery package, the device can be powered by AC or DC electrical network. To connect the device to AC power supply, use the cable from the delivery package. To connect the device to DC power supply, use wires with a minimum cross-section of 1 mm^2.
4. Turn the device on and check the front panel LEDs to make sure the terminal is in normal operating conditions.

3.5 SFP transceiver installation and removal

> ⚠ **Optical modules can be installed when the terminal is turned on or off.**

3.5.1 Transceiver installation

1. Insert the top SFP module into a slot with its open side down, and the bottom SFP module with its open side up.

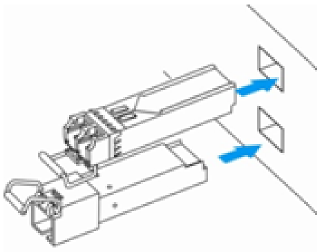


Figure 65 – SFP transceivers installation

2. Push the module into the device housing until it is secured with a clicking sound.

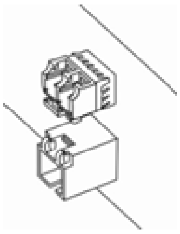


Figure 66 – Installed SFP transceivers

3.5.2 Transceiver removal

1. Flip the module handle to unlock the latch.

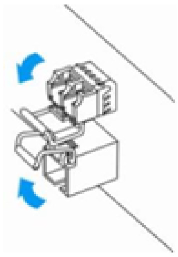


Figure 67 – Opening SFP transceiver latch

2. Remove the module from the slot.

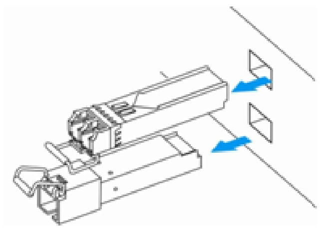


Figure 68 – SFP transceivers removal

4 Management interfaces

To control and monitor the device, various management interfaces can be used.

To access the device, you may use network connection via Telnet or SSH as well as direct connection via RS-232 compliant console port. For Telnet, SSH or console port connections, the command line interface is used for device management.

Factory settings contain trusted zone description and IP address for device management access – 192.168.1.1/24.
Trusted zone includes the following interfaces:

- **for ESR-10: GigabitEthernet 1/0/2-6;**
- **for ESR-12V(F), ESR-14VF: GigabitEthernet 1/0/2-8;**
- **for ESR-15: GigabitEthernet 1/0/2-5;**
- **for ESR-20: GigabitEthernet 1/0/2-4;**
- **for ESR-21: GigabitEthernet 1/0/2-12;**
- **for ESR-30: GigabitEthernet 1/0/3-4;**
- **for ESR-100: GigabitEthernet 1/0/2-4;**
- **for ESR-200: GigabitEthernet 1/0/2-8;**
- **for ESR-1000: GigabitEthernet 1/0/2-24;**
- **for ESR-1200: GigabitEthernet 1/0/2-16, TengigabitEthernet 1/0/3-8;**
- **for ESR-1500: GigabitEthernet 1/0/2-8, TengigabitEthernet 1/0/2-4;**
- **for ESR-1511: GigabitEthernet 1/0/2-8, TengigabitEthernet 1/0/1-4;**
- **for ESR-1700: GigabitEthernet 1/0/2-4, TengigabitEthernet 1/0/3-12;**
- **for ESR-3100: GigabitEthernet 1/0/2-8, TengigabitEthernet 1/0/3-8;**
- **for ESR-3200: GigabitEthernet 1/0/2-8, TengigabitEthernet 1/0/3-12.**

By default, the user 'admin' with the password 'password' is defined in factory settings.

For each management interface provided, there are unified configuration operating principles. When modifying and applying the configuration, you should follow the specific sequence described herein that is intended to protect the device from misconfiguration.

4.1 Command line interface (CLI)

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).

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

You can create as many users as you like, access rights will be assigned individually to each user.

To ensure command line interface security, all commands are divided into 2 categories–privileged and unprivileged. Privileged commands basically include configuration commands. Unprivileged commands include monitoring commands.

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

4.2 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 55 – 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: **fortygigabitethernet 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
Sub-interfaces	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** • **fortygigabitethernet 1/0/2.1024** • **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** • **fortygigabitethernet 1/0/2.408.507** • **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, • **<SLOT>** – number of device E1 module, • **<STREAM>** – E1 flow sequence number. 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 <CHANNEL_ID>**
Logical interfaces	Designation of logical interface is the interface sequence number: Designation examples: • **loopback 4** • **bridge 60** • **service-port 1**

Interface type	Designation
Serial interfaces	Designation of serial interface includes its type and identifier. Serial 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
```

4.3 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 56 – Types and naming procedure of router tunnels

Tunnel type	Designation
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**
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 examples: **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**

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

5 Initial router configuration

5.1 ESR router factory configuration

The device is shipped to the consumer with the factory configuration installed that includes essential basic settings. Factory configuration allows using the router as a gateway with SNAT without applying any additional settings. Also, factory configuration contains settings that allow you to obtain network access to the device for advanced configuration.

5.1.1 Description of factory settings

To establish network connection, the configuration features 2 security zones named 'Trusted' for local area network and 'Untrusted' for public network. All interfaces are divided between two security zones:

1. **'Untrusted' zone** is meant for a public network (WAN) connection. In this zone, DHCP ports are open in order to obtain dynamic IP address from the provider. All incoming connections from this zone to the router are blocked.
   This security zone includes the following interfaces:
   - for ESR-10/12V: GigabitEthernet 1/0/1;
   - for ESR-12VF/ESR-14VF: GigabitEthernet 1/0/1; GigabitEthernet 1/0/9;
   - for ESR-15: GigabitEthernet 1/0/1; GigabitEthernet 1/0/6;
   - for ESR-20: GigabitEthernet 1/0/1;
   - for ESR-21: GigabitEthernet 1/0/1;
   - for ESR-30: GigabitEthernet 1/0/1; GigabitEthernet 1/0/1-2;
   - for ESR-100/200: GigabitEthernet 1/0/1;
   - for ESR-1000/1500/3100: GigabitEthernet 1/0/1, TengigabitEthernet 1/0/1-2;
   - for ESR-1200/1700: GigabitEthernet 1/0/1, TengigabitEthernet 1/0/1, TengigabitEthernet 1/0/2;
   - for ESR-1511: GigabitEthernet 1/0/1, FortygigabitEthernet 1/0/1-2;
   - for ESR-3200: GigabitEthernet 1/0/1-2.

     Zone interfaces are grouped into a single L2 segment via *Bridge 2* network bridge.
2. **'Trusted' zone** is meant for a local area network (LAN) connection. Telnet and SSH ports for remote access, ICMP ports for router availability test, DHCP ports for clients obtaining IP addresses from the router. Outgoing connections from this zone into the Untrusted zone are allowed.
   This security zone includes the following interfaces:
   - for ESR-10: GigabitEthernet 1/0/2-6;
   - for ESR-12V(F)/ESR-14VF: GigabitEthernet 1/0/2-8;
   - for ESR-15: GigabitEthernet 1/0/2-5;
   - for ESR-20: GigabitEthernet 1/0/2-4;
   - for ESR-21: GigabitEthernet 1/0/2-12;

- for ESR-30: GigabitEthernet 1/0/3-4;
- for ESR-100: GigabitEthernet 1/0/2-4;
- for ESR-200: GigabitEthernet 1/0/2-8;
- for ESR-1000: GigabitEthernet 1/0/2-24;
- for ESR-1200: GigabitEthernet 1/0/2-16, TengigabitEthernet 1/0/3-8;
- for ESR-1500: GigabitEthernet 1/0/2-8, TengigabitEthernet 1/0/3-4;
- for ESR-1511: GigabitEthernet 1/0/2-8, TengigabitEthernet 1/0/1-4;
- for ESR-1700: GigabitEthernet 1/0/2-4, TengigabitEthernet 1/0/3-12;
- for ESR-3100: GigabitEthernet 1/0/2-8, TengigabitEthernet 1/0/3-8;
- for ESR-3200: Twentyfivegigabitethernet 1/0/3-12.

Zone interfaces are grouped into a single L2 segment via *Bridge 2* network bridge.

On the *Bridge 2* interface, DHCP client is enabled to obtain dynamic IP address from the provider. On *Bridge 1* interface, static IP address 192.168.1.1/24 is configured. Created IP address acts as a gateway for LAN clients. For LAN clients, DHCP address pool 192.168.1.2-192.168.1.254 is configured with the mask 255.255.255.0. For clients in order to access the Internet, the router should have Source NAT service enabled.

Security zone policies have the following configuration:

Table 57 – Security zone policy description

Traffic origin zone	Traffic destination zone	Traffic type	Action
Trusted	Untrusted	TCP, UDP, ICMP	enabled
Trusted	Trusted	TCP, UDP, ICMP	enabled
Trusted	self	TCP/22 (SSH), ICMP, UDP/67 (DHCP Server), UDP/123 (NTP)	enabled
Untrusted	self	UDP/68 (DHCP Client)	enabled

To enable device configuration on the first startup, 'admin' user with 'password' password has been created in the router configuration. The user will be prompted to change administrator password during the initial configuration of the router.

To enable network access to the router on the first startup, static IP address 192.168.1.1/24 has been configured on Bridge 1 interface.

5.2 Router connection and configuration

ESR series routers are intended to perform border gateway functions and securing the user network when it is connected to public data networks.

Basic router configuration should include:

- Assigning IP addresses (static or dynamic) to the interfaces that participate in data routing;
- Creation of security zones and distribution of interfaces between these zones;
- Creation of policies governing data transfer through these zones;
- Configuration of services that accompany the data routing (NAT, Firewall, etc.).

Advanced settings depend on the requirements of the specific device application pattern and may be easily added or modified with the existing management interfaces.

5.2.1 Connection to the router

There are several device connection options:

Ethernet LAN connection

> **Upon the initial startup, the router starts with the factory configuration. The factory configuration is described in the** ESR router factory configuration **section of this manual.**

Connect the network data cable (patch cord) to any port within the ***'Trusted'*** zone and to the PC intended for management tasks.

In the router factory configuration, DHCP server is enabled with IP address pool in **192.168.1.0/24** subnet.

When network interface is connected to the management computer, the latter should obtain the network address from the server.

If IP address is not obtained for some reason, assign the interface address manually using any address except for 192.168.1.1 in 192.168.1.0/24 subnet.

RS-232 console port connection

Using RJ-45/DBF9 cable included into device delivery package, connect the router '**Console**' port to the computer RS-232 port.

Launch terminal application (e.g. HyperTerminal or Minicom) and create a new connection. VT100 terminal emulation mode should be used.

Specify the following settings for RS-232 interface:

```
Data rate: 115200 bps
Data bits: 8 bits
Parity: none
Stop bits: 1
Flow control: none
```

5.2.2 Applying the configuration change

Any changes made in the configuration will take effect only after applying the command:

```
esr# commit
Configuration has been successfully committed
```

After applying the command above, the configuration rollback timer is started. To stop the timer and rollback mechanism, use the following command:

```
esr# confirm
Configuration has been successfully confirmed
```

The default 'rollback' timer value is 600 seconds. To change this timer, use the command:

```
esr(config)# system config-confirm timeout <TIME>
```

- <TIME> – time period of configuration confirmation pending, takes value in seconds [120..86400].

5.2.3 Basic router configuration

Upon the first startup, the router configuration procedure includes the following steps:

- Changing password for "admin" user.
- Creation of new users.
- Assigning device name (Hostname).
- Setting parameters for public network connection in accordance with the provider requirements.
- Configuring remote connection to router.
- Applying basic settings.

Changing password for 'admin' user

To ensure the secure system access, you should change the password for the privileged 'admin' user.

> ⚠ **'techsupport' account ('eltex' up to version 1.0.7) is required for service centre specialist remote access.**
> **'remote' account – RADIUS, TACACS+, LDAP authentication.**
> **'admin', 'techsupport', 'remote' users cannot be deleted. Only passwords and a privilege level can be changed.**

> ❗ **If information about 'admin' user is not displayed in the configuration, then the parameters of this user are configured by default ('password' password, privilege level 15).**

Username and password are required for login during the device administration sessions.

To change 'admin' password, use the following commands:

```
esr# configure
esr(config)# username admin
esr(config-user)# password <new-password>
esr(config-user)# exit
```

Creation of new users

Use the following commands to create a new system user or configure the username, password, or privilege level:

```
esr(config)# username <name>
esr(config-user)# password <password>
esr(config-user)# privilege <privilege>
esr(config-user)# exit
```

Privilege levels 1–9 allow accessing the device and viewing its operation status, but the device configuration is disabled. Privilege levels 10-14 allow both the access to the device and configuration of majority of its functions. Privilege level 15 allows both the access to the device and configuration of all its functions.

Example of commands, that allow you to create user **'fedor'** with password **'12345678'** and privilege level **15** and create user **'ivan'** with password **'password'** and privilege level **'1'**:

```
esr# configure
esr(config)# username fedor
esr(config-user)# password 12345678
esr(config-user)# privilege 15
esr(config-user)# exit
esr(config)# username ivan
esr(config-user)# password password
esr(config-user)# privilege 1
esr(config-user)# exit
```

Assigning device name

To assign the device name, use the following commands:

```
esr# configure
esr(config)# hostname <new-name>
```

When a new configuration is applied, command prompt will change to the value specified by **<new-name>** parameter.

Configuration of public network parameters

To configure router network interface in the public network, you should assign parameters defined by the network provider – default IP address, subnet mask and gateway address – to the device.

Example of static IP address configuration commands for **Gigabit Ethernet 1/0/2.150** sub-interface used for obtaining access to the router via **VLAN 150**.

Interface parameters:

- IP address: 192.168.16.144;
- Subnet mask: 255.255.255.0;
- Default gateway IP address: 192.168.16.1.

```
esr# configure
esr(config)# interface gigabitethernet 1/0/2.150
esr(config-subif)# ip address 192.168.16.144/24
esr(config-subif)# exit
esr(config)# ip route 0.0.0.0/0 192.168.16.1
```

To ensure the correct IP address assigning for the interface, enter the following command when the configuration is applied:

```
esr# show ip interfaces
IP address           Interface                          Type
-------------------  ---------------------------------  -------
192.168.16.144/24    gigabitethernet 1/0/2.150          static
```

Provider may use dynamically assigned addresses in their network. If the there is DHCP server in the network, you can obtain the IP address via DHCP.

Configuration example for obtaining dynamic IP address from DHCP server on **Gigabit Ethernet 1/0/10** interface:

```
esr# configure
esr(config)# interface gigabitethernet 1/0/10
esr(config-if)# ip address dhcp
esr(config-if)# exit
```

To ensure the correct IP address assigning for the interface, enter the following command when the configuration is applied:

```
esr# show ip interfaces
IP address           Interface                          Type
-------------------  ---------------------------------  -------
192.168.11.5/25      gigabitethernet 1/0/10             DHCP
```

Configuring remote connection to router

In the factory configuration, remote access to the router may be established via Telnet or SSH from the **'trusted'** zone. To enable remote access to the router from other zones, e.g. from the public network, you should create the respective rules in the firewall.

When configuring access to the router, rules should be created for the following pair of zones:

- **source-zone** – zone that the remote access will originate from;
- **self** – zone which includes router management interface.

Use the following commands to create the allowing rule:

```
esr# configure
esr(config)# security zone-pair <source-zone> self
esr(config-zone-pair)# rule <number>
esr(config-zone-rule)# action permit
esr(config-zone-rule)# match protocol tcp
esr(config-zone-rule)# match source-address <network object-group>
esr(config-zone-rule)# match destination-address <network object-group>
esr(config-zone-rule)# match destination-port <service object-group>
esr(config-zone-rule)# enable
esr(config-zone-rule)# exit
esr(config-zone-pair)# exit
```

Example of commands that allow users from **'untrusted'** zone with IP addresses in range **132.16.0.5-132.16.0.10** to connect to the router with IP address **40.13.1.22** via SSH:

```
esr# configure
esr(config)# object-group network clients
esr(config-addr-set)# ip address-range 132.16.0.5-132.16.0.10
esr(config-addr-set)# exit
esr(config)# object-group network gateway
esr(config-addr-set)# ip address-range 40.13.1.22
esr(config-addr-set)# exit
esr(config)# object-group service ssh
esr(config-port-set)# port-range 22
esr(config-port-set)# exit
esr(config)# security zone-pair untrusted self
esr(config-zone-pair)# rule 10
esr(config-zone-rule)# action permit
esr(config-zone-rule)# match protocol tcp
esr(config-zone-rule)# match source-address clients
esr(config-zone-rule)# match destination-address gateway
esr(config-zone-rule)# match destination-port ssh
esr(config-zone-rule)# enable
esr(config-zone-rule)# exit
esr(config-zone-pair)# exit
```

6 Firmware update

- Updating firmware via system resources
- Updating firmware via bootloader
- Secondary bootloader update (U-Boot)

6.1 Updating firmware via system resources

To update the firmware, use any of the following servers: TFTP, FTP, SCP. Router firmware files obtained from the manufacturer should be allocated on the server.
The router stores two copies of the firmware. To ensure the reliability of the firmware update procedure, only the copy that was not used for the last device startup is available for the update.

When update the firmware, the router configuration is converted according to a new version.
When loading a router with an older software version than the previously loaded configuration, the configuration is not converted and is subsequently deleted.

Firmware from earlier versions can be updated using instructions in the Updating firmware via bootloader **section.**

To update the firmware for the device running the operating system, follow procedure described below.

1. Prepare the selected server for operation. The server address must be known; also firmware distributive file should be loaded onto the server.
2. The router should be prepared for operation according to the documentary requirements. Router configuration should allow for data exchange with the server via TFTP/FTP/SCP and ICMP protocols. Server's belonging to the security zones of the router must be taken into account.
3. Connect to the router locally via Console port or remotely via Telnet or SSH.
   Check the server availability for the router using *ping* command on the router. If the server is not available, check the router settings and the status of the server network interfaces.
4. To update the router firmware, enter the following command. Specify IP address of the server being used as *<server>* For updates that utilize FTP or SCP server, enter a username (*<user>* parameter) and a password (*<password>* parameter). Specify the name of the firmware file loaded onto the server as *<file_name>* parameter (when using SCP, the full path must be as *<folder>* parameter). When the command is executed, router will copy the file into its internal memory, perform data integrity check and save it into non-volatile memory.
   TFTP:

```
esr# copy tftp://<server>:/<file_name> system:firmware
```

   FTP:

```
esr# copy ftp://[<user>[:<password>]@]<server>:/<file_name> system:firmware
```

   SCP:

```
esr# copy scp://[<user>[:<password>]@]<server>://<folder>/<file_name>system:firmware
```

   SFTP:

```
esr# copy sftp://[<user>[:<password>]@]<server>:/<file_name> system:firmware
```

Example of updating main firmware via SCP:

```
esr# copy scp://adm:password123@192.168.16.168://home/tftp/firmware system:firmware
```

5. To start the device with the new firmware version, switch the active image. With *show bootvar* command, locate the image number, containing updated firmware.

```
esr# show bootvar
Image   Version                     Date                   Status         After reboot
-----  --------------              --------------------   ------------   ------------
1       1.0.7 build 141[f812808]    date 18/02/2015 time    Active          *
                                    16:12:54

2       1.0.7 build 141[f812808]    date 18/02/2015 time   Not Active
                                    16:12:54
```

Use the following command to select the image:

```
esr# boot system image-[1|2]
```

6. To update the secondary bootloader (U-Boot), enter the following command: Specify IP address of the server being used as *<server>* parameter. For updates that utilize FTP or SCP server, enter a username (*<user>* parameter) and a password (*<password>* parameter). Specify the name of the secondary bootloader onto the server as *<file_name>* parameter (when using SCP, the full path must be as *<folder>* parameter). When the command is executed, router will copy the file into its internal memory, perform data integrity check and save it into non-volatile memory.
TFTP:

```
esr# copy tftp://<server>:/<file_name> system:boot-2
```

FTP:

```
esr# copy ftp://<server>:/<file_name> system:boot-2
```

SCP:

```
esr# copy scp://[<user>[:<password>]@]<server>://<folder>/<file_name>system:boot-2
```

SFTP:

```
esr# copy sftp://<server>:/<file_name> system:boot-2
```

6.2 Updating firmware via bootloader

Router firmware may be updated via the bootloader as follows:

1. When U-Boot finishes the router initialization, break the device startup with the ***<Esc>*** key.

```
Configuring PoE...
distribution 1 dest_threshold 0xa drop_timer 0x0
Configuring POE in bypass mode
NAE configuration done!
initializing port 0, type 2.
initializing port 1, type 2.
SMC Endian Test:b81fb81f
nae-0, nae-1
  =======Skip: Load SYS UCORE for old 8xxB1/3xxB0 revision on default.
Hit any key to stop autoboot:  2
```

2. Specify TFTP server address:

```
BRCM.XLP316Lite Rev B0.u-boot# serverip 10.100.100.1
```

For version 1.5 and newer:

```
BRCM.XLP316Lite Rev B0.u-boot# serverip 10.100.100.1
```

3. Specify router IP address:

```
BRCM.XLP316Lite Rev B0.u-boot# ipaddr 10.100.100.2
```

For version 1.5 and newer:

```
BRCM.XLP316Lite Rev B0.u-boot# ipaddr 10.100.100.2
```

4. Specify the name of the firmware file on the TFTP server:

```
BRCM.XLP316Lite Rev B0.u-boot# firmware_file firmware
```

For version 1.5 and newer:

```
BRCM.XLP316Lite Rev B0.u-boot# firmware_file firmware
```

5. The environment can be saved using 'saveenv' command for future updates.

6. Launch firmware update procedure:

```
BRCM.XLP316Lite Rev B0.u-boot# run tftp_update_image1
Using nae-0-3 device
TFTP from server 10.100.100.1; our IP address is 10.100.100.2
Filename 'esr1000/firmware'.
Load address: 0xa800000060000000
Loading: TftpStart:TftpTimeoutMsecs = 10000, TftpTimeoutCountMax = 6
#################################################################
#################################################################
#################################################################
#########################
####################################
done
Bytes transferred = 64453909 (3d77d15 hex)
Device 0: MT29F8G08ABBCAH4 ... is now current device

NAND erase: device 0 offset 0x1440000, size 0x6400000
Bad block table found at page 262080, version 0x01
Bad block table found at page 262016, version 0x01
Erasing at 0x7800000 -- 1895825408% complete..
OK

NAND write: device 0 offset 0x1440000, size 0x6400000
104857600 bytes written: OK
```

7. Set the downloaded firmware as an image to start the system and reboot the router:

```
BRCM.XLP316Lite Rev B0.u-boot# run set_bootpart_1
```

For version 1.5 and newer:

```
BRCM.XLP316Lite Rev B0.u-boot# boot_system image1
BRCM.XLP316Lite Rev B0.u-boot# reset
```

6.3 Secondary bootloader update (U-Boot)

Secondary bootloader initializes NAND and the router. During the update, a new file of the secondary bootloader is saved to the flash

To view the current version of the load file operating on the device, execute 'version' command in U-Boot CLI. Also, the version is displayed during the router startup:

```
BRCM.XLP316Lite Rev B0.u-boot# version
BRCM.XLP.U-Boot:1.1.0.47 (29/11/2016 - 19:00:24)
```

Firmware update procedure:

1. When U-Boot finishes the router initialization, break the device startup with the ***<Esc>*** key.

```
Configuring PoE...
distribution 1 dest_threshold 0xa drop_timer 0x0
Configuring POE in bypass mode
NAE configuration done!
initializing port 0, type 2.
initializing port 1, type 2.
SMC Endian Test:b81fb81f
nae-0, nae-1
  =======Skip: Load SYS UCORE for old 8xxB1/3xxB0 revision on default.
Hit any key to stop autoboot:  2
```

2. Specify TFTP server address:

```
BRCM.XLP316Lite Rev B0.u-boot# setenv serverip 10.100.100.1
```

   For version 1.5 and newer:

```
BRCM.XLP316Lite Rev B0.u-boot# serverip10.100.100.2
```

3. Specify router IP address:

```
BRCM.XLP316Lite Rev B0.u-boot# setenv ipaddr 10.100.100.2
```

   For version 1.5 and newer:

```
BRCM.XLP316Lite Rev B0.u-boot# ipaddr 10.100.100.2
```

4. Specify the name of the bootloader file on the TFTP server:

```
BRCM.XLP316Lite Rev B0.u-boot# uboot_file u-boot.bin
```

   For version 1.5 and newer:

```
BRCM.XLP316Lite Rev B0.u-boot# uboot_file u-boot.bin
```

5. The environment can be saved using 'saveenv' command for future updates.
6. Launch firmware update procedure:

```
BRCM.XLP316Lite Rev B0.u-boot# run upd_uboot
```

For version 1.5 and newer:

```
BRCM.XLP316LiteRevB0.u-boot# run tftp_update_uboot
Using nae-1 device
TFTP from server 10.100.100.1; our IP address is 10.100.100.2
Filename 'esr1000/u-boot.bin'.
Load address: 0xa800000078020000
Loading: ###########################################################
done
Bytes transferred = 852648 (d02a8 hex)
SF: Detected MX25L12805D with page size 256, total 16777216 bytes
16384 KiB MX25L12805D at 0:0 is now current device
```

7. Reboot the router:

```
BRCM.XLP316Lite Rev B0.u-boot# reset
```

7 Safe configuration recommendations

The safe configuration recommendations are general and suitable for most installations. These recommendations greatly improve the safe operation of the unit, but are not exhaustive. Depending on the application of the device, other safety parameters must also be configured. In some specific cases, the implementation of these recommendations may result in a non-functional network. When configuring the device, firstly it is necessary to follow the technical requirements and regulations of the networks in which the device will be used.

7.1 General recommendations

- It is recommended to always disable unused physical interfaces with the **shutdown** The command is described in detail in the Interface monitoring and configuration section of the CLI Command Reference.
- It is recommended to always set the system clock to synchronize with trusted network time sources (NTP). The NTP setup algorithm is described in the NTP configuration section of this manual. For detailed information on the NTP configuration commands, see System timer management in the CLI Command Reference.
- It is recommended to disable the NTP broadcast client, which is enabled by default in the factory configuration.
- It is not recommended to use the **ip firewall disable** command that disables firewalling. Always assign appropriate security zones to interfaces and configure the correct firewall rules. The firewall configuration algorithm is described in the Firewall configuration section of this manual. For detailed information on the Firewall configuration commands, see Firewall management in the CLI Command Reference.

7.2 Event logging system configuration

Event logging system configuration algorithms are described in the 'Syslog configuration' subsection of the Monitoring section of this manual.

For detailed information on the Event logging system configuration commands, see SYSLOG management section in the CLI Command Reference.

7.2.1 Recommendations

- It is recommended to configure the event message storage in a syslog file on the device and transfer these events to an external syslog server.
- It is recommended to limit the size of the syslog file on the device.
- It is recommended to configure syslog file rotation on the device.
- It is recommended to enable syslog message enumeration.
- It is recommended that timestamp msec tags be added to syslog messages on ESR-1500 and ESR-1511.

7.2.2 Warnings

- The data stored in the **tmpsys:syslog** file system is not saved when the device is rebooted. This type of file system is recommended for storing operational logs.
- It is not recommended to use the **flash:syslog** file system to store logs, as it may cause premature ESR device failure.

7.2.3 Configuration example

Objective:

Configure the storage of event messages of info level and higher in a syslog file on the device and configure transmission of these events to an external syslog server. Limit the file size to 512 KB. Enable rotation of 3 files. Enable syslog message enumeration.

Solution:

Configure the storage of syslog messages in the file:

```
esr(config)# syslog file tmpsys:syslog/default info
```

Configure size limitation and file rotation:

```
esr(config)# syslog max-files 3
esr(config)# syslog file-size 512
```

Configure the transmission of messages to an external server:

```
esr(config)# syslog host mylog 192.168.1.2 info udp 514
```

Enable syslog message enumeration:

```
esr(config)# syslog sequence-numbers
```

7.3 Password usage policy configuration

The configuration algorithms for the password usage policy are described in the AAA configuration section of this manual.

For detailed information on the configuration commands for the password usage policy, see AAA configuration in the CLI Commands Reference.

7.3.1 Recommendations

- It is recommended to always enable the default password change request for the admin user.
- It is recommended to limit the lifetime of passwords and prohibit reusing at least the previous password.
- It is recommended to set the minimum password length requirement greater than 8 characters.
- It is recommended to set requirements for the use of lowercase and uppercase letters, numbers and special characters.

7.3.2 Configuration example

Objective:

- Configure a password policy with a requirement to change the default password, a password validity period of 1 month, and a ban on using the last 12 passwords.
- Set the minimum password length to 16 characters, the maximum to 64 characters.
- The password must contain at least 3 uppercase letters, at least 5 lowercase letters, at least 4 digits and at least 2 special characters. The password must contain all 4 types of characters.

Solution:

Enables the default password reset request for admin user:

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

Set the password lifetime to 30 days and prohibit the use of the previous 12 passwords:

```
esr(config)# security passwords lifetime 30
esr(config)# security passwords history 12
```

Set a limit to the password length:

```
esr(config)# security passwords min-length 16
esr(config)# security passwords max-length 64
```

Set a limit on the minimum number of characters of the respective types:

```
esr(config)# security passwords upper-case 3
esr(config)# security passwords lower-case 5
esr(config)# security passwords special-case 2
esr(config)# security passwords numeric-count 4
esr(config)# security passwords symbol-types 4
```

7.4 AAA policy configuration

The algorithms for AAA policy are described in the AAA configuration section of this manual.

For detailed information on the commands for AAA policy, see AAA configuration in the CLI Commands Reference.

7.4.1 Recommendations

- It is recommended to use a role-based access model on the device.
- It is recommended to use personal accounts to authenticate on the device.

- It is recommended to enable logging of commands entered by the user.
- It is recommended to use several authentication methods for logging in to devices via console, remote login to devices and privilege escalation. A combination of RADIUS/TACACS/LDAP authentication and local authentication is considered optimal.
- It is recommended to lower the built-in **admin** account privileges to 1.
- It is recommended to configure logging of changes of local accounts.
- It is recommended to configure AAA policy change logging.

7.4.2 Warnings

- The built-in admin account cannot be deleted.
- The **no username admin** command does not remove the **admin** user, it resets his configuration to defaults. After applying this command, the **admin** user will not appear in the configuration.
- The **no password** command for the **admin** user also does not remove the **admin** user's password, but resets it to its default value. After applying this command, the **admin** user password is no longer displayed in the configuration and becomes 'password'.
- User with privilege level 15 or an ENABLE password must be configured before setting the admin user to downgrade privileges.

7.4.3 Configuration example

Objective:

Configure AAA policy:

- Use RADIUS authentication for remote login via SSH.
- Use RADIUS authentication for local console login, use local authentication if there is no connection to RADIUS servers.
- Use ENABLE password set via RADIUS, if there is no connection to RADIUS servers, use local ENABLE password.
- Set the admin user to a reduced privilege level.
- Configure logging of changes of local accounts.
- Configure AAA policy changes logging.
- Configure the logging of entered commands.

Solution:

Create a **local-operator** user with privilege level 8:

```
esr(config)# username local-operator
esr(config-user)# password Pa$$w0rd1
esr(config-user)# privilege 8
esr(config-user)# exit
```

Set local ENABLE password:

```
esr(config)# enable password $6e5c4r3e2t!
```

Lower the privileges of the admin user:

```
esr(config)# username admin
esr(config-user)# privilege 1
esr(config-user)# exit
```

Configure the connection to the two RADIUS servers, the primary 192.168.1.11 and the backup 192.168.2.12:

```
esr(config)# radius-server host 192.168.1.11
esr(config-radius-server)# key ascii-text encrypted 8CB5107EA7005AFF
esr(config-radius-server)# priority 100 esr(config-radius-server)# exit
esr(config)# radius-server host 192.168.2.12
esr(config-radius-server)# key ascii-text encrypted 8CB5107EA7005AFF
esr(config-radius-server)# priority 150
esr(config-radius-server)# exit
```

Configure AAA policy:

```
esr(config)# aaa authentication login CONSOLE radius local
esr(config)# aaa authentication login SSH radius
esr(config)# aaa authentication enable default radius enable
esr(config)# aaa authentication mode break
esr(config)# line console
esr(config-line-console)# login authentication CONSOLE
esr(config-line-console)# exit esr(config)# line ssh
esr(config-line-ssh)# login authentication SSH
esr(config-line-ssh)# exit
```

Configure logging:

```
esr(config)# logging userinfo
esr(config)# logging aaa
esr(config)# syslog cli-commands
```

7.5 Remote management configuration

For more information on remote access configuration commands, see SSH, Telnet access configuration in the CLI command reference.

7.5.1 Recommendations

- It is recommended to disable remote control via telnet.
- It is recommended to use crypto-resistant sha2-512 authentication algorithms and disable all others.
- It is recommended to use crypto-resistant aes256ctr encryption algorithms and disable all others.
- It is recommended to use dh-group-exchange-sha256 crypto-proof encryption key exchange algorithm and disable all others.
- It is recommended to use crypto-resistant Host-Key verification algorithm for SSH rsa and disable all others.
- It is recommended to allow access to remote control of the device only from certain IP addresses;
- It is recommended to regenerate the encryption keys before starting operation.

7.5.2 Configuration example

Objective:

Disable telnet. Generate new encryption keys. Use crypto-resistant algorithms.

Solution:

Disable remote telnet control:

```
esr(config)# no ip telnet server
```

Disable outdated and not crypto-resistant algorithms:

```
esr(config)# ip ssh server
esr(config)# ip ssh authentication algorithm md5 disable
esr(config)# ip ssh authentication algorithm md5-96 disable
esr(config)# ip ssh authentication algorithm ripemd160 disable
esr(config)# ip ssh authentication algorithm sha1 disable
esr(config)# ip ssh authentication algorithm sha1-96 disable
esr(config)# ip ssh authentication algorithm sha2-256 disable
esr(config)# ip ssh encryption algorithm 3des disable
esr(config)# ip ssh encryption algorithm aes128 disable
esr(config)# ip ssh encryption algorithm aes128ctr disable
esr(config)# ip ssh encryption algorithm aes192 disable
esr(config)# ip ssh encryption algorithm aes192ctr disable
esr(config)# ip ssh encryption algorithm aes256 disable
esr(config)# ip ssh encryption algorithm arcfour disable
esr(config)# ip ssh encryption algorithm arcfour128 disable
esr(config)# ip ssh encryption algorithm arcfour256 disable
esr(config)# ip ssh encryption algorithm blowfish disable
esr(config)# ip ssh encryption algorithm cast128 disable
esr(config)# ip ssh key-exchange algorithm dh-group-exchange-sha1 disable
esr(config)# ip ssh key-exchange algorithm dh-group1-sha1 disable
esr(config)# ip ssh key-exchange algorithm dh-group14-sha1 disable
esr(config)# ip ssh key-exchange algorithm ecdh-sha2-nistp256 disable
esr(config)# ip ssh key-exchange algorithm ecdh-sha2-nistp384 disable
esr(config)# ip ssh key-exchange algorithm ecdh-sha2-nistp521 disable
esr(config)# ip ssh host-key algorithm dsa disable
esr(config)# ip ssh host-key algorithm ecdsa256 disable
esr(config)# ip ssh host-key algorithm ecdsa384 disable
esr(config)# ip ssh host-key algorithm ecdsa521 disable
esr(config)# ip ssh host-key algorithm ed25519 disable
```

Generate new encryption keys:

```
esr# update ssh-host-key rsa
esr# update ssh-host-key rsa 2048
```

7.6 Configuration of protection against network attacks mechanisms

The algorithms for configuring the network attack protection mechanisms are described in the Logging and network protection configuration section of this manual.

For detailed information about the commands to configure the password policy, see Management of logging and protection against network attacks in the CLI Command Reference.

7.6.1 Recommendations

- It is recommended to always enable protection against ip spoofing.
- It is recommended to always enable protection against TCP packets with incorrectly set flags.
- It is recommended to always enable protection against fragmented TCP packets with the SYN flag set.
- It is recommended to always enable protection against fragmented ICMP packets.
- It is recommended to always enable protection against large ICMP packets.
- It is recommended to always enable protection against unregistered IP protocols.
- It is recommended to enable logging of the protection mechanism against network attacks.

7.6.2 Configuration example

Objective:

Configure the protection mechanism against network attacks in accordance with the recommendations.

Solution:

Enable protection against ip spoofing and logging of the protection mechanism:

```
esr(config)# ip firewall screen spy-blocking spoofing
esr(config)# logging firewall screen spy-blocking spoofing
```

Enable protection against TCP packets with incorrectly set flags and logging of the protection mechanism:

```
esr(config)# ip firewall screen spy-blocking syn-fin
esr(config)# logging firewall screen spy-blocking syn-fin
esr(config)# ip firewall screen spy-blocking fin-no-ack
esr(config)# logging firewall screen spy-blocking fin-no-ack
esr(config)# ip firewall screen spy-blocking tcp-no-flag
esr(config)# logging firewall screen spy-blocking tcp-no-flag
esr(config)# ip firewall screen spy-blocking tcp-all-flags
esr(config)# logging firewall screen spy-blocking tcp-all-flags
```

Enable protection against fragmented ICMP packets and protection mechanism logging:

```
esr(config)# ip firewall screen suspicious-packets icmp-fragment
esr(config)# logging firewall screen suspicious-packets icmp-fragment
```

Enable protection against large ICMP packets and logging of the protection mechanism:

```
esr(config)# ip firewall screen suspicious-packets large-icmp
esr(config)# logging firewall screen suspicious-packets large-icmp
```

Enable protection against unregistered ip-protocols and logging protection mechanism:

```
esr(config)# ip firewall screen suspicious-packets unknown-protocols
esr(config)# logging firewall screen suspicious-packets unknown-protocols
```

8 Interface management

8.1 VLAN Configuration

VLAN (*Virtual Local Area Network*) is a logical (virtual) local area network that represents a group of devices, which communicate on channel level regardless of their physical location. VLAN operation is based on the use of additional Ethernet header fields according to 802.1q standard. In fact, VLAN isolates the broadcast domain by limiting the switching of only those Ethernet frames which have the same VLAN-ID in the Ethernet header.

8.1.1 Configuration algorithm

Step	Description	Command	Keys
1	Create VLAN.	**esr(config)# vlan <VID>**	<VID> – VLAN identifier, set in the range of [2..4094]. It is also possible to create multiple vlan (comma separated), vlan range (hyphen separated) or combined entry containing commas and hyphens.
2	Specify vlan name (optional).	**esr(config-vlan)# name <vlan-name>**	<vlan-name> – up to 255 characters.
3	Disable monitoring of the status of interfaces on which processing of the given VLAN Ethernet frames is allowed (optional).	**esr(config-vlan)# force-up**	
4	Disable the processing of incoming untagged Ethernet frames based on the default VLAN's switching table (VLAN-ID – 1) (optional).	**esr(config-if-gi)# switchport forbidden default-vlan**	
5	Set L2 interface operation mode.	**esr(config-if-gi)# mode switchport**	
6	Set the combined mode of the physical interface.	**esr(config-if-gi)# mode hybrid**	Only for ESR-1000/1200/1500/1511/1700.
7	Set L2 interface operation mode.	**esr(config-if-gi)# switchport access**	Only for ESR-10/12V(F)/14VF/15/20/21/30/100/200/3100/3200. This mode is the default mode and is not displayed in the configuration.
		esr(config-if-gi)# switchport trunk	Only for ESR-10/12V(F)/14VF/20/21/100/200/3100.

Step	Description	Command	Keys
		esr(config-gi)# switchport general	Only for ESR-1000/1200/1500/1511/1700. This mode is the default mode and is not displayed in the configuration.
8	Configure VLAN list on the interface in tagged mode.	**esr(config-if-gi)# switchport trunk allowed vlan add <VID>**	Only for ESR-10/12V(F)/14VF/15/20/21/30/100/200/3100/3200. <VID> – VLAN identifier, set in the range of [2..4094]. It is also possible to create multiple vlan (with a comma) or vlan range (with a hyphen).
		esr(config-if-gi)# switchport general allowed vlan add <VID> tagged	For ESR-1000/1200/1500/1511/1700. <VID> – VLAN identifier, set in the range of [2..4094]. It is also possible to create multiple vlan (with a comma) or vlan range (with a hyphen).
9	Configure VLAN on the interface in tagged mode (optional).	**esr(config-if-gi)# switchport trunk native-vlan <VID>**	Only for ESR-10/12V(F)/14VF/15/20/21/30/100/200/3100/3200. <VID> – VLAN identifier, set in the range of [2..4094].
		esr(config-if-gi)# switchport general allowed vlan add <VID> untagged	For ESR-1000/1200/1500/1511/1700. <VID> – VLAN identifier, set in the range of [2..4094].
10	Enable the processing of Ethernet frames of all created VLANs on the interface (optional).	**esr(config-if-gi)# switchport trunk allowed vlan auto-all**	Only for ESR-10/12V(F)/14VF/15/20/21/30/100/200/3100/3200.
		esr(config-if-gi)# switchport general allowed vlan auto-all	Only for ESR-1000/1200/1500/1511/1700.
11	Set VLAN port identifier (PVID) for incoming untagged traffic (optional).	**esr(config-if-gi)# switchport general pvid <PVID>**	Only for ESR-1000/1200/1500/1511/1700.

8.1.2 Configuration example 1. VLAN removal from the interface

Objective:

On the basis of the factory configuration, remove gi1/0/1 port from VLAN 2.

Solution:

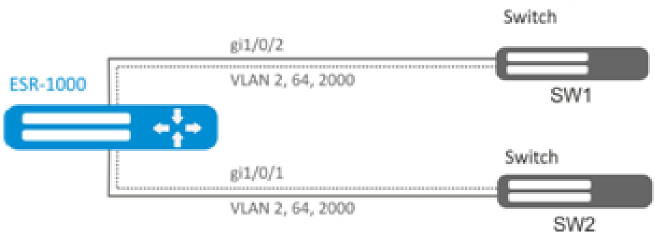
Remove VLAN2 from gi1/0/1 port:

```
esr(config)# interface gi 1/0/1
esr(config-if-gi)# switchport general allowed vlan remove 2 untagged
esr(config-if-gi)# no switchport general pvid
```

8.1.3 Configuration example 2. Enabling VLAN processing in tagged mode

Objective:

Configure gi1/0/1 and gi1/0/2 ports for packet transmission and reception in VLAN 2, VLAN 64, VLAN 2000.



Solution:

Create VLAN 2, VLAN 64, VLAN 2000 on ESR-1000:

```
esr-1000(config)# vlan 2,64,2000
```

Specify VLAN 2, VLAN 64, VLAN 2000 for gi1/0/1-2 port:

```
esr-1000(config)# interface gi1/0/1
esr-1000(config-if-gi)# mode switchport
esr-1000(config-if-gi)# switchport forbidden default-vlan
esr-1000(config-if-gi)# switchport general allowed vlan add 2,64,2000 tagged
```

8.1.4 Configuration example 3. Enabling VLAN processing in tagged and untagged modes

Objective:

Configure gi1/0/1 ports for packet transmission and reception in VLAN 2, VLAN 64, VLAN 2000 in trunk mode, configure gi1/0/2 port in access mode for VLAN 2 on ESR-100/ESR -200.

Solution:

Create VLAN 2, VLAN 64, VLAN 2000 on ESR-100/ESR-200:

```
esr(config)# vlan 2,64,2000
```

Specify VLAN 2, VLAN 64, VLAN 2000 for gi1/0/1 port:

```
esr(config)# interface gi1/0/1
esr(config-if-gi)# mode switchport
esr(config-if-gi)# switchport forbidden default-vlan
esr(config-if-gi)# switchport mode trunk
esr(config-if-gi)# switchport trunk allowed vlan add 2,64,2000
```

Specify VLAN2 to gi1/0/2 port:

```
esr(config)# interface gi1/0/2
esr(config-if-gi)# mode switchport
esr(config-if-gi)# switchport access vlan 2
```

8.2 LLDP configuration

Link Layer Discovery Protocol (LLDP) is a data link layer protocol allowing network equipment to notify the devices operating in a local network of its existence and to transmit parameters to it as well as to receive similar information.

8.2.1 Configuration algorithm

Step	Description	Command	Keys
1	Enable LLDP on the router.	**esr(config)# lldp enable**	
2	Enable the LLDPDU receiving and proceeding on the physical interface.	**esr(config-if-gi)# lldp receive**	
3	Enable LLDPDU transmission on the physical interface.	**esr(config-if-gi)# lldp transmit**	
8	Set the LLDPDU sending period (optional).	**esr(config)# lldp timer <SEC>**	<SEC> – time interval in seconds, takes values of [1..32768]. Default value: 30
4	Set the period during which the router keeps the information received via LLDP (optional)	**esr(config)# lldp hold-multiplier <SEC>**	<SEC> – time interval in seconds, takes values of [1..10]. Default value: 4
5	Set IP address which will be transmitted to LLDP TLV as the management-address (optional).	**esr(config)# lldp management-address <ADDR>**	<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. One of the existent is set by default
6	Set the system-description field which will be transmitted to LLDP TLV as the system-description (optional).	**esr(config)# lldp system-description <DESCRIPTION>**	<DESCRIPTION> – system description, set by the string of up to 255 characters. By default contains the information of the router model and firmware version.

Step	Description	Command	Keys
7	Set the system-name field which will be transmitted to LLDP TLV as the system-name (optional).	**esr(config)# lldp system-name <NAME>**	<NAME> – system name, set by the string of up to 255 characters. By default coincides with the specified hostname

8.2.2 Configuration example

Objective:

Organize the LLDPDU exchange and proceeding between ESR-1 and ESR-2 routers.

Solution:

1. R1 configuration
   Enable LLDP globally on the router:

```
esr(config)# lldp enable
```

   Enable the receiving and transmission of LLDPDU on the gi 1/0/1 interface.

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# lldp receive
esr(config-if-gi)# lldp transmit
```

2. R2 configuration
   Enable LLDP globally on the router:

```
esr(config)# lldp enable
```

   Enable the receiving and transmission of LLDPDU on the gi 1/0/1 interface.

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# lldp receive
esr(config-if-gi)# lldp transmit
```

   To view LLDP neighbors information, use the following command:

```
esr# show lldp neighbors
```

   To view more detailed information on the certain interface neighbor, use the following command:

```
esr# show lldp neighbors gigabitethernet 1/0/1
```

To view LLDP statistics, use the following command:

```
esr# show lldp statistics
```

8.3 LLDP MED configuration

LLDP MED — LLDP standard enhancement which allows to transmit network policies: VLAN ID, DSCP, priority.

8.3.1 Configuration algorithm

Step	Description	Command	Keys
1	Enable LLDP on the router	**esr(config)# lldp enable**	
2	Enable LLDPDU transmission on the physical interface.	**esr(config-if-gi)# lldp transmit**	
3	Enable MED LLDP enhancement on the router	**esr(config)# lldp med fast-start enable**	
4	Create network policy.	**esr(config)# network-policy <NAME>**	<NAME> – network-policy name, set by the string of up to 31 characters.
5	Specify the application type.	**esr(config-net-policy)# application <APP_TYPE>**	<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.
6	Set the DSCP value (optional).	**esr(config-net-policy)# dscp <DSCP>**	<DSCP> – DSCP code value, takes values in the range of [0..63].

Step	Description	Command	Keys
7	Set the CoS value (optional).	**esr(config-net-policy)# priority <PRIORITY>**	<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.
8	Set VLAN ID value.	**esr(config-net-policy)# vlan <VID> [tagged]**	<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.
9	Set a network policy on the interface.	**esr(config-if-gi)# lldp network-policy <NAME>**	<NAME> – network-policy name, set by the string of up to 31 characters.

8.3.2 Voice VLAN configuration example

Voice VLAN — VLAN ID, in receiving of which an IP phone switches to the trunk mode with the specified VLAN ID for VoIP traffic reception and transmission. VLAN ID transmission is performed by LLDP MED enhancement.

Objective:

VoIP traffic and data traffic should be grouped in different VLANs - vid 10 for data and vid 20 for VoIP - and the sending of Voice VLAN from the gi 1/0/1 ESR port should be configured. Voice VLAN should be supported and enabled on the IP phone.

Solution:

First create VLAN 10 and 20 and configure the gi 1/0/1 interface in the trunk mode:

```
esr(config)# vlan 10,20
esr(config-vlan)# exit
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# mode switchport
esr(config-if-gi)# switchport mode trunk
esr(config-if-gi)# switchport trunk allowed vlan add 10,20
esr(config-if-gi)# exit
```

Enable LLDP and MED capability in LLDP globally on the router:

```
esr(config)# lldp enable
esr(config)# lldp med fast-start enable
```

Create and configure network policy in the way that VLAN ID 20 is specified for the voice application:

```
esr(config)# network-policy VOICE_VLAN
esr(config-net-policy)# application voice
esr(config-net-policy)# vlan 20 tagged
esr(config-net-policy)# exit
```

Configure LLDP on the interface and set a network policy:

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# lldp transmit
esr(config-if-gi)# lldp receive
esr(config-if-gi)# lldp network-policy VOICE_VLAN
esr(config-if-gi)# exit
```

8.4 Sub-interface termination configuration

To terminate Ethernet frames of a certain VLAN on a specific physical interface, create a sub-interface with the number of VLAN, frames of which will be terminated. When creating two sub-interfaces having the same VLAN but located on different physical/aggregated interfaces, switching of Ethernet frames between these sub-interfaces will not be possible as external segments will be separate broadcast domains. For data exchange between subscribers of different sub-interfaces (even with the same VLAN-ID) routing will be used, i.e. data exchange will occur at the third level of the OSI model.

8.4.1 Configuration algorithm

Step	Description	Command	Keys
1	Create a sub-interface of a physical interface (possible if the physical interface is in routeport or hybrid mode).	**esr(config)# interface gigabitethernet <PORT>.<S-VLAN>** or **interface tengigabitethernet <PORT>.<S-VLAN>** or **interface port-channel <CH>.<S-VLAN>**	<PORT> – physical interface number. <CH> – aggregated interface number. <S-VLAN> – identifier of created S-VLAN. If a physical interface is included in bridge-group, it will be impossible to create sub-interface.
2	Specify sub-interface description (optional).	**esr(config-subif)# description <DESCRIPTION>**	<DESCRIPTION> – interface description, set by the string of up to 255 characters.
3	Specify VRF instance, in which the given sub-interface will operate (optional).	**esr(config-subif)# ip vrf forwarding <VRF>**	<VRF> – VRF name, set by the string of up to 31 characters.
4	Specify the IPv4/IPv6 address and subnet mask for the interface to be configured or enable IP address obtain dynamically.	**esr(config-subif)# ip address <ADDR/LEN>**	<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]. For advanced IPv4 addressing features see section IP addressing configuration.
		esr(config-subif)# ipv6 address <IPV6-ADDR/LEN>	<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]. For advanced IPv6 addressing features see section IPv6 addressing configuration. Several IPv4/IPv6 addresses can be specified separated by commas. Up to 8 IPv4/IPv6 addresses can be assigned to the interface.

Step	Description	Command	Keys
		esr(config-subif)# ip address dhcp	For advanced DHCP client operation features, see section DHCP Client management.
5	Disable the Firewall features on the interface or enable the interface in the security zone (see Firewall configuration).	**esr(config-subif)# ip firewall disable**	
		esr(config-subif)# security-zone <NAME>	<NAME> – security zone name, set by the string of up to 31 characters.
6	Set the time interval during which statistics on the sub-interface load is collected. (optional).	**esr(config-subif)# load-average <TIME>**	<TIME> – interval in seconds, takes values of [5..150].
7	Set the lifetime of IPv4/IPv6 entries in the ARP table studied on the given interface (optional).	**esr(config-subif)# ip arp reachable-time <TIME>** or **esr(config-subif)# ipv6 nd reachable-time <TIME>**	<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>.
8	Change MTU (MaximumTransmitionUnit) size. MTU above 1500 will be active only when using the 'system jumbo-frames' command (optional).	**esr(config-subif)# mtu <MTU>**	<MTU> – MTU value in bytes. Default value: 1500.
9	Enable recording of the current interface usage statistics (optional).	**esr(config-subif)# history statistics**	
10	Override the MSS (Maximum segment size) field in incoming TCP packets (optional).	**esr(config-subif)# ip tcp adjust-mss <MSS>** **esr(config-subif)# ipv6 tcp adjust-mss <MSS>**	<MSS> – MSS value, takes values in the range of [500..1460]. Default value: 1460

It is also possible to configure the sub-interface:

- QoS in basic or advanced mode (see section QoS management);
- proxy (see section HTTP/HTTPS traffic proxying);
- traffic monitoring (see sections Netflow configuration and sFlow configuration);
- routing protocols functionality (see section Routing management);
- VRRF protocol (see section Redundancy management);
- BRAS functionality (see section BRAS (Broadband Remote Access Server) management);
- IDS/IPS functionality (see section IPS/IDS configuration).

8.4.2 Sub-interface configuration example

Objective:

Configure subnet 192.168.3.1/24 in VLAN: 828 on the physical interface gigabitethernet 1/0/1.

Solution:

Create sub-interface for VLAN: 828

```
esr(config)# interface gigabitethernet 1/0/1.828
```

Configure IP address from necessary subnet.

```
esr(config)# interface gigabitethernet 1/0/1.828
esr(config-subif)# ip address 192.168.3.1/24
esr(config-subif)# exit
```

In addition to assigning an IP address, either disable the firewall or configure the corresponding security zone on the sub interface.

8.5 Q-in-Q termination configuration

Q-in-Q is a technology of packet transmission with two 802.1q tags. The technology is used for extending quantity of VLANs in data networks. 802.1q header, which is closer to payload, is an Inner Tag. also known as C-VLAN (Customer VLAN). 802.1q header, which is comes before C-VLAN, is an Outer Tag also known as S-VLAN (Service VLAN). Using of double tags in Ethernet frames is describing by 802.1ad protocol.

8.5.1 Configuration algorithm

Step	Description	Command	Keys
1	Create a sub-interface of a physical interface (possible if the physical interface is in routeport or hybrid mode).	**esr(config)# interface gigabitethernet <PORT>.<S-VLAN>** or **interface tengigabitethernet <PORT>.<S-VLAN>** or **interface port-channel <CH>.<S-VLAN>**	<PORT> – physical interface number. <CH> – aggregated interface number. <S-VLAN> – identifier of created S-VLAN. If a physical interface is included in bridge-group, it will be impossible to create sub-interface.

Step	Description	Command	Keys
2	Create Q-in-Q interface.	**esr(config)# interface gigabitethernet <PORT>.<S-VLAN>.<C-VLAN>** or **esr(config)# interface tengigabitethernet <PORT>.<S-VLAN>.<C-VLAN>** or **esr(config)# interface port-channel <CH>.<S-VLAN>.<C-VLAN>**	<PORT> – physical interface number. <CH> – aggregated interface number. <S-VLAN> – identifier of created S-VLAN. <C-VLAN> – identifier of created C-VLAN. If a physical interface or a sub-interface is included in bridge-group, it will be impossible to create sub-interface.
3	Specify Q-in-Q interface description (optional).	**esr(config-qinq-if)# description <DESCRIPTION>**	<DESCRIPTION> – interface description, set by the string of up to 255 characters.
4	Specify VRF instance, in which the given Q-in-Q interface will operate (optional).	**esr(config-qinq-if) # ip vrf forwarding <VRF>**	<VRF> – VRF name, set by the string of up to 31 characters.
5	Specify the IPv4/IPv6 address and subnet mask for the interface to be configured or enable IP address obtain dynamically.	**esr(config-qinq-if)# ip address <ADDR/LEN>**	<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]. For advanced IPv4 addressing features see section IP addressing configuration.
		esr(config-qinq-if)# ipv6 address <IPV6-ADDR/LEN>	<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]. For advanced IPv6 addressing features see section IPv6 addressing configuration. Several IPv4/IPv6 addresses can be specified separated by commas. Up to 8 IPv4/IPv6 addresses can be assigned to the interface.

Step	Description	Command	Keys
		esr(config-qinq-if)# ip address dhcp	For advanced DHCP client operation features, see section DHCP Client management.
6	Disable the Firewall features on the interface or enable the interface in the security zone (see Firewall configuration).	**esr(config-qinq-if)# ip firewall disable**	
		esr(config-qinq-if)# security-zone <NAME>	<NAME> – security zone name, set by the string of up to 31 characters.
7	Set the time interval during which statistics on the sub-interface load is collected. (optional).	**esr(config-subif)# load-average <TIME>**	<TIME> – interval in seconds, takes values of [5..150].
8	Set the lifetime of IPv4/IPv6 entries in the ARP table studied on the given interface (optional).	**esr(config-subif)# ip arp reachable-time <TIME>** or **esr(config-subif)# ipv6 nd reachable-time <TIME>**	<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>.
9	Change MTU (MaximumTransmitionUnit) size. MTU above 1500 will be active only when using the 'system jumbo-frames' command (optional).	**esr(config-subif)# mtu <MTU>**	<MTU> – MTU value in bytes. Default value: 1500.
10	Enable recording of the current interface usage statistics (optional).	**esr(config-subif)# history statistics**	
11	Override the MSS (Maximum segment size) field in incoming TCP packets (optional).	**esr(config-subif)# ip tcp adjust-mss <MSS>** **esr(config-subif)# ipv6 tcp adjust-mss <MSS>**	<MSS> – MSS value, takes values in the range of [500..1460]. Default value: 1460

It is also possible to configure the Q-in-Q interface:

- QoS in basic or advanced mode (see section QoS management);
- proxy (see section HTTP/HTTPS traffic proxying);
- traffic monitoring (see sections Netflow configuration and sFlow configuration);
- routing protocols functionality (see section Routing management);
- VRRF protocol (see section Redundancy management);
- BRAS functionality (see section BRAS (Broadband Remote Access Server) management);
- IDS/IPS functionality (see section IPS/IDS configuration).

8.5.2 Q-in-Q configuration example

Objective:

Configure the termination of subnet 192.168.1.1/24 combination C-VLAN: 741, S-VLAN: 828 on the physical interface gigabitethernet 1/0/1.

Solution:

Create sub-interface for S-VLAN: 828

```
esr(config)# interface gigabitethernet 1/0/1.828
esr(config-subif)# exit
```

Create a Q-in-Q interface for the S-VLAN: 741 and configure the IP address from the required subnet.

```
esr(config)# interface gigabitethernet 1/0/1.828.741
esr(config-qinq-if)# ip address 192.168.1.1/24
esr(config-qinq-if)# exit
```

Besides assigning IP address, it is necessary to disable firewall or to configure corresponding security zone on Q-in-Q interface.

8.6 USB modems configuration

The use of USB modems allows organizing additional link channel for router operation. When connecting USB modems, use USB hubs can be used. Up to 10 USB modems can be configured in the system at the same time.

8.6.1 USB modems configuration algorithm

Step	Description	Command	Keys
1	After USB modem connection, wait until the system detects the connected device.		
2	Define which number of the device is allocated to the connected USB modem.	**esr# show cellulars status modem**	The connected device identifier will be specified in 'USB port' field
3	Create parameter profile for USB modem and switch to the profile configuration mode.	**esr(config)# cellular profile <ID>**	<ID> – identifier of USB modem parameter profile, set in the range of [1..10].
4	Specify parameter profile description (optional).	**esr(config-cellular-profile)# description <DESCRIPTION>**	<DESCRIPTION> – profile description, set by the string of up to 255 characters.

Step	Description	Command	Keys
5	Set mobile network access point	**esr(config-cellular-profile)# apn <NAME>**	<NAME> – mobile network access point, set by the string of up to 31 characters.
6	Set the name of mobile network user (if authentication by login/password required by cellular carrier).	**esr(config-cellular-profile)# user <NAME>**	<NAME> – user name, set by the string of up to 31 characters.
7	Set the password of mobile network user (if authentication by login/ password required by cellular carrier).	**esr(config-user)# password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }**	<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.
8	Activate user (if authentication by login/password required by cellular carrier).	**esr(config-user)# enable**	
9	Set the dial-up number for connection to the mobile network.	**esr(config-cellular-profile)# number <WORD>**	<WORD> – dial-up number for connection to a mobile network, set by the string of up to 15 characters.
10	Set the method of user authentication in the mobile network (optional).	**esr(config-cellular-profile)# allowed-auth <TYPE>**	<TYPE> – method of user authentication in a mobile network [none, PAP, CHAP, MSCHAP, MSCHAPv2, EAP]. Default value: PAP
11	Limit the possibility of the use of IP addresses in mobile network.	**esr(config-cellular-profile)# ip-version { ipv4 | ipv6 }**	• **ipv4** – IPv4 family; • **ipv6** – IPv6 family;
12	Create USB modem in the router configuration and switch to the modem configuration mode.	**esr(config)# cellular modem <ID>**	<ID> – USB modem identifier, set in the range of [1..10].
13	Specify modem description (optional).	**esr(config-cellular-modem)# description <DESCRIPTION>**	<DESCRIPTION> – modem description, set by the string of up to 255 characters.
14	Specify VRF instance, in which the given modem will operate (optional).	**esr(config-cellular-modem)# ip vrf forwarding <VRF>**	<VRF> – VRF name, set by the string of up to 31 characters.

Step	Description	Command	Keys
15	Set USB modem identifier allocated by the system (specified in item 2).	**esr(config-cellular-modem)# device <WORD>**	<WORD> – identifier of connected modem's USB port, set in the range of [1..12].
16	Set the previously established parameter profile to the USB modem.	**esr(config-cellular-modem)# profile <ID>**	<ID> – identifier of USB modem parameter profile, set in the range of [1..10].
17	Set SIM card unlock code (if necessary).	**esr(config-cellular-modem)# pin <WORD>**	<WORD> – SIM card unblock code [4..8]. Only digits are allowed.
18	Allow the use of any USB modem operation mode (optional).	**esr(config-cellular-modem)# allowed-mode <MODE>**	<MODE> – acceptable USB modem operation mode [2g, 3g, 4g]. By default: all modes supported by the modem are allowed.
19	Set the size of the largest received packet (optional).	**esr(config-cellular-modem)# mru { <MRU> }**	<MRU> – MRU value, takes values in the range of [128..16383]. Default value: 1500.
20	Change the maximum size of processed MTU (MaximumTransmissionUnit) packets. MTU above 1500 will be active only when using the «system jumbo-frames» command (optional).	**esr(config-cellular-modem)# mtu <MTU>**	<MTU> – MTU value in bytes. Default value: 1500.
21	Set the preferable USB modem operation mode in the mobile network (optional).	**esr(config-cellular-modem)# preferred-mode { <MODE> }**	<MODE> – preferable USB modem operation mode [2g, 3g, 4g].
22	Disable the Firewall features on the interface or enable the interface in the security zone (see Firewall configuration).	**esr(config-subif)# ip firewall disable**	
		esr(config-subif)# security-zone <NAME>	<NAME> – security zone name, set by the string of up to 31 characters.
23	Activate USB modem.	**esr(config-cellular-modem)# enable**	

Step	Description	Command	Keys
It is also possible to configure a cellular network modem: • QoS in basic or advanced mode (see section QoS management); • proxy (see section HTTP/HTTPS traffic proxying); • traffic monitoring (see sections Netflow configuration and sFlow configuration); • routing protocol functionality (see sections Policy-based routing and MultiWAN).			

For the full modem mobile network functionality, additionally configure the routing and NAT functionality.

8.6.2 Configuration example

Objective:

Configure connection to the Internet by using USB modem.

Solution:

For example, consider the connection to the cellular operator MTS.

After modem connection, wait until the system detects the device. Determine the port of the device that was assigned to the connected USB modem:

```
esr# show cellular status modem
Number
device  USB port    Manufacturer    Model   Current state    Interface    Link    state
1       1-2         huawei          E3372   Disabled         --           Down
```

Create the parameter profile for USB modem:

```
esr(config)# cellular profile 1
```

Specify the required APN or any other necessary address. The example of connection to MTS APN is given below:

```
esr(config-cellular-profile)# apn internet.mts.ru
```

If necessary, create user name, password, dial-up number and authentication number:

```
esr(config-cellular-profile)# user mts
esr(config-ppp-user)# password ascii-text mts
esr(config-cellular-profile)# number *99#
esr(config-cellular-profile)# allowed-auth PAP
```

Proceed to configuring the USB modem and set the identifier corresponding to the device port that was defined at the beginning:

```
esr(config)# cellular modem 1
esr(config-cellular-modem)# device 1-2
```

Set the corresponding parameter profile and activate the modem:

```
esr(config-cellular-modem)# profile 1
esr(config-cellular-modem)# enable
```

8.7 STP/RSTP configuration

Spanning Tree Protocol is a network protocol to bring an Ethernet network with redundant connections to a tree topology that excludes loops. Network devices exchange configuration messages using frames of a special format and selectively enable and disable transmission on ports.

Rapid STP (RSTP) is an improvement of the STP protocol, characterized by a shorter time to bring the network to a tree topology and has higher stability.

8.7.1 Spanning Tree configuration algorithm

Step	Description	Command	Keys
1	Enable spanning-tree in the vlan-aware mode.	**esr(config)# spanning-tree**	
2	Set the amount of time to spend listening and learning states before switching to transmit state.	**esr(config)# spanning-tree forward-time <TIME>**	<TIME> – time in seconds, takes values [4..30]. Default value: 15 seconds.
3	Set the time interval between sending BPDU packets.	**esr(config)# spanning-tree hello-time <TIME>**	<TIME> – time in seconds, takes values [1..10]. Default value: 2 seconds.
4	Set STP lifetime.	**esr(config)# spanning-tree max-age <TIME>**	<TIME> – time in seconds, takes values [6..40]. Default value: 20 seconds.
5	Select a supported protocol from the STP family.	**esr(config)# spanning-tree mode <MODE>**	<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.
6	Set the path value determination method.	**esr(config)# spanning-tree pathcost method <short | long>**	long – value in the range of [1..200000000]; short – value in the range of [1..65535]. Default value: short.

Step	Description	Command	Keys
7	Set STP priority.	**esr(config)# spanning-tree priority <PRIORITY>**	<PRIORITY> – priority, specified in the range with increments of 4096 [0..61440]. Default value: 32768.
8	Switch to interface/tunnel/network bridge configuration mode.	**esr(config)# interface <IF-TYPE><IF-NUM>**	<IF-TYPE> – interface type; <IF-NUM> – F/S/P – F frame (1), S – slot (0), P – port.
		esr(config)# tunnel <TUN-TYPE><TUN-NUM>	<TUN-TYPE> – tunnel type; <TUN-NUM> – tunnel number.
		esr(config)# bridge <BR-NUM>	<BR-NUM> – bridge number.
9	Enable spanning-tree.	**esr(config-bridge)# spanning-tree**	
10	Set the amount of time to spend listening and learning states before switching to transmit state.	**esr(config-bridge)# spanning-tree forward-time <TIME>**	<TIME> – time in seconds, takes values [4..30]. Default value: 15 seconds.
11	Set the time interval between sending BPDU packets.	**esr(config-bridge)# spanning-tree hello-time <TIME>**	<TIME> – time in seconds, takes values [1..10]. Default value: 2 seconds.
12	Set STP lifetime.	**esr(config-bridge)# spanning-tree max-age <TIME>**	<TIME> – time in seconds, takes values [6..40]. Default value: 20 seconds.
13	Select a supported protocol from the STP family.	**esr(config-bridge)# spanning-tree mode <MODE>**	<MODE> – STP family protocol: • STP – IEEE 802.1D Spanning Tree Protocol; • RSTP – IEEE 802.1W Rapid Spanning Tree Protocol. Default value: RSTP.
14	Set STP priority.	**esr(config-bridge)# spanning-tree priority <PRIORITY>**	<PRIORITY> – priority, specified in the range with increments of 4096 [0..61440]. Default value: 32768.
15	Disable STP on the configured interface.	**esr(config-if-gi)# spanning-tree disable**	

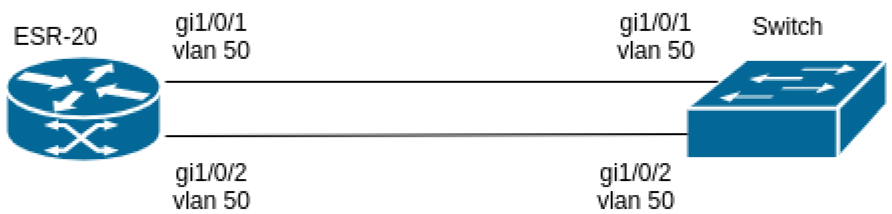
Step	Description	Command	Keys
16	Set the path value determination method.	**esr(config-if-gi)# spanning-tree cost**	<COST> – path cost in the range [1..20000000]. Default value: 4.
17	Allow this port to be set as root.	**esr(config-if-gi)# spanning-tree guard root**	
18	Set the RSTP protocol to the transmitting state and determine the type of connection for the selected port – 'point-to-point', 'branched'.	**esr(config-if-gi)# spanning-tree link-type {point-to-point|shared}**	point-to-point – the command defines the interface as 'point-to-point'; shared – the command defines the interface as 'shared'. Default value: point-to-point.
19	Set interface priority in STP spanning tree.	**esr(config-if-gi)# spanning-tree port-priority <PRIORITY>**	<PRIORITY> – priority, specified in the range with increments of 16 [0..240].
20	The mode is activated, in which the port, when the link is up, immediately switches to the transmitting state, without waiting for the timer to expire.	**esr(config-if-gi)# spanning-tree portfast**	

⚠ The commands for configuring MSTP are only available on the ESR-1000 as the MSTP protocol is currently supported solely on this device.

8.7.2 Configuration example

Objective:

Configure STP on the router to prevent a loop with a 10 second listen and learn interval and a spanning tree lifetime of 15 seconds.



Solution:

Example of a diagram with a router and a switch connected by two links is given below.

RSTP protocol on the ESR is enabled by defalut.

Go to the configuration mode:

```
esr-20# configure
```

Set STP as default:

```
esr-20(config)# spanning-tree mode stp
```

Set spanning tree lifetime to 15 seconds and network listening and learning interval to 10 seconds:

```
esr-20(config)# spanning-tree max-age 15
esr-20(config)# spanning-tree forward-time 10
```

Show spanning-tree active command output:

```
esr-20# show spanning-tree active
Protocol version: STP
          Root ID: [32768] a8:f9:4b:ad:5a:00
                    Root port: [128] gi1/0/1
                    Pathcost 32768
                    Message Age 300
                    Hello time: 2 Max age time: 20 Forward delay: 15
        Bridge ID: [32768] a8:f9:4b:ad:8e:5d
                    Hello time: 2 Max age time: 15 Forward delay: 10
                    Transmit hold count: 6 Topology change: 0
                    Time since topology change: 16 Topology change count: 2
Name           State   Prio.Num   Cost        Status     Role       PortFast   Type
------------   -----   --------   ---------   --------   --------   --------   -----
gi1/0/1        en      128.2      32768       FRW        Root       No         STP
gi1/0/2        en      128.3      32768       BLK        Altr       No         STP
```

8.8 PPP through E1 configuration

PPP (*Point-to-Point Protocol*) — point-to-point link layer protocol, used to establish direct communication between two network nodes. It can provide connection authentication, encryption and data compression.

To establish a PPP connection through the E1 stream, there must be TOPGATE-WAN-E1 media converter in the ESR router.

8.8.1 Configuration algorithm

Step	Description	Command	Keys
Preconfiguration:			
1	Enable support for Jumbo frames. To apply changes, the device restart is required.	**esr(config)# system jumbo-frames**	
Physical interface configuration:			
2	Select the interface where TOPGATE-WAN-E1 is installed.	**esr(config)# interface gigabitethernet 1/0/3**	
3	Put physical interface in switch mode.	**esr(config-if-gi)# mode switchport**	

Step	Description	Command	Keys
4	Set the operation mode of the E1 interface.	**esr(config-if-gi)# switchport mode e1**	
5	Set the synchronization source (optional).	**esr(config-if-gi)# switchport e1 clock source <SOURCE>**	<SOURCE> – synchronization source: • **Internal** (default) – synchronize with an internal source; • **line** – synchronize with a linear signal.
6	Specify MTU (Maximum Transmission Unit) size for physical interfaces.	**esr(config-if-gi)# mtu <MTU>**	<MTU> – MTU value, for E1 and Multilink interfaces may take values in the range of [1510..9600].
7	Specify frame check hash algorithm (optional).	**esr(config-if-gi)# switchport e1 crc <FCS>**	<FCS> – frame check sequence: • **16** (default) – FCS16; • **32** – FCS32.
8	Set check for transmission errors (optional).	**esr(config-if-gi)# switchport e1 framing <CRC>**	<CRC> – cyclic redundancy check: • **crc-4** – use CRC-4 algorithm; • **no-crc4** (default) – do not use check.
9	Set transmitting bits inversion (optional).	**esr(config-if-gi)# switchport e1 invert data**	
10	Set linear encoding type (optional).	**esr(config-if-gi)# switchport e1 linecode <CODE>**	<CODE> – linear encoding type; • **ami** – alternate mark inversion; • **hdb3** (default) – high density bipolar of order 3.
11	Set amount of timeslots.	**esr(config-if-gi)# switchport e1 timeslots <RANGE>**	<RANGE> – amount of timeslots.
12	Use E1 as a single entity, without time slots (optional).	**esr(config-if-gi)# switchport e1 unframed**	
E1 interface configuration:			
13	Select E1 interface.	**esr(config)# interface e1 1/<SLOT>/1**	<SLOT> – slot number.

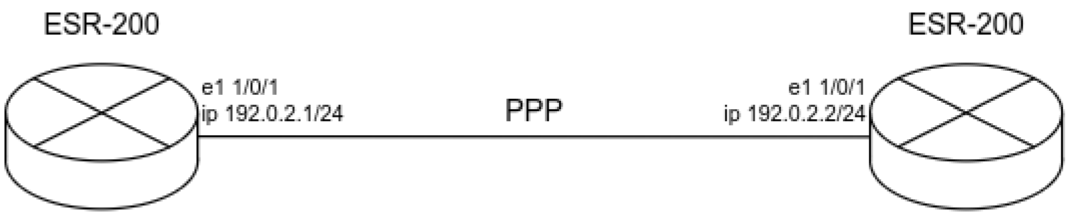
Step	Description	Command	Keys
14	Specify IPv4 and network mask for the configured interface.	**esr(config-e1)# ip address <ADDR/LEN>**	<ADDR/LEN> – IP address network mask length specified as AAA.BBB.CCC.DDD/EE, where each AAA – DDD part takes values [0..255] and EE takes values [1..32].
15	Disable Firewall functions on the interface or include the interface in the security zone (see the Firewall configuration section).	**esr(config-e1)# ip firewall disable**	
		esr(config-e1)# security-zone <NAME>	<NAME> – name of the security zone, specified as a string of up to 31 characters.
Additional PPP settings for E1:			
16	Enable CHAP authentication for PPP (optional).	**esr(config-e1)# ppp authentication chap**	
17	Specify router name that is sent to a remote party for CHAP authentication (optional).	**esr(config-e1)# ppp chap hostname <NAME>**	<NAME> – router name.
18	Set authentication password (optional).	**esr(config-e1)# ppp chap password ascii-text <CLEAR-TEXT>**	<CLEAR-TEXT> – unencrypted password, set by the string of [1..64] characters, may include [0-9a-fA-F] characters.
19	Enable authentication override (optional).	**esr(config-e1)# ppp chap refuse**	
20	Set authentication username (optional).	**esr(config-e1)# ppp chap username <NAME>**	<NAME> – user name.
21	Allow any non-null IP address to be accepted as a local IP address from the neighbour (optional).	**esr(config-e1)# ppp ipcp accept-address**	
22	Set IP address that is sent to a remote party for the further allocation (optional).	**esr(config-e1)# ppp ipcp remote-address <ADDR>**	<ADDR> – IP address of a remote gateway.
23	Set the amount of attempts to send Configure-Request packets before the remote peer is found to be unable to respond (optional).	**esr(config-e1)# ppp max-configure <VALUE>**	<VALUE> – number of retries.
24	Set the amount of attempts to send Configure-NAK packets before all options are confirmed (optional).	**esr(config-e1)# ppp max-failure <VALUE>**	<VALUE> – number of retries.

Step	Description	Command	Keys
25	Set the amount of attempts to send Terminate-Request packets before the session is aborted (optional).	**esr(config-e1)# ppp max-terminate <VALUE>**	<VALUE> – number of retries.
26	Set MRU (Maximum Receive Unit) size for the interface (optional).	**esr(config-e1)# ppp mru <MRU>**	<MRU> – MRU value.
27	Specify the time interval in seconds after which the router sends a keepalive message (optional).	**esr(config-e1)# ppp timeout keepalive <TIME>**	<TIME> – time in seconds.
28	Specify the interval after which the router sends a keepalive message (optional).	**esr(config-e1)# ppp timeout retry <TIME>**	<TIME> – time in seconds.
Enable E1 interface in Multilink PPP:			
29	Add to MLPPP group (optional).	**esr(config-e1)# ppp multilink-group <GROUP-ID>**	<GROUP-ID> – group number.
30	Enable MLPPP mode (optional).	**esr(config-e1)# ppp multilink**	

8.8.2 Configuration example

Objective:

Configure PPP connection to the opposite side with IP address 192.0.2.2/24 via TOPGATE-WAN-E1 using 1-8 channel slots for data transmission; the clock source is the opposite side.



Solution:

First, configure system jumbo-frames, save changes to the configuration and reboot the router:

```
esr(config)# system jumbo-frames
esr(config)# exit
esr# commit
esr# confirm
esr# reload system
Do you really want to reload system ? (y/N): y
```

Configure the physical interface gigabitethernet 1/0/3, in which TOPGATE-WAN-E1 is installed:

- Set mtu equal to 1510 or more;
- Switch interface to the e1 operation mode;
- Specify channel e1 – 0;
- Specify e1 channel range from 1 to 8.

```
esr# configure
esr(config)# interface gigabitethernet 1/0/3
esr(config-if-gi)# mode switchport
esr(config-if-gi)# mtu 1510
esr(config-if-gi)# switchport mode e1
esr(config-if-gi)# switchport e1 slot 0
esr(config-if-gi)# switchport e1 timeslots 1-8
esr(config-if-gi)# exit
```

Configure e1 interface:

```
esr(config)# interface e1 1/0/1
esr(config-e1)# ip address 192.0.2.1/24
esr(config-e1)# security-zone trusted
esr(config-e1)# exit
```

Information about the physical state of e1 can be obtained using the following command:

```
esr# show controllers e1 gigabitethernet 1/0/3
Interface 'gi1/0/3':
 SFP present:     Yes
 SFP Vendor name: --
 is te:     No
 SFP Vendor PN:   --
 SFP SW Version:  LPOS 1.0.9.4SR42 (20.12.2017) [
 Line code:       HDB3
 Clock source:    Internal
 Timeslot:        1-8
 Invert Data:     No
 Framing CRC4:    No
 Loopback:        --
 CRC algorithm:   FCS16
 E1 Link:         Up
 E1 Synced:       Yes
 E1 RX AIS:       No
 E1 RX RAI:       No
 E1 TX AIS:       No
```

Information about the state of e1 can be obtained using the following command:

```
esr# show interfaces status e1 1/0/1
Interface 'e1 1/0/1' status information:
 Description:          --
 Operational state:    Up
 Administrative state: Up
 Track ID:             0
 Supports broadcast:   No
 Supports multicast:   Yes
 MTU:                  1492
 MAC address:          none
 Last change:          1 minute and 3 seconds
 Mode:                 routerport
```

8.9 MLPPP Configuration

Multilink PPP (MLPPP) is an aggregated channel that encompasses methods of traffic transition via multiple physical channels while having a single logical connection. This option allows enhancing bandwidth and enables load balancing.

8.9.1 Configuration algorithm

Step	Description	Command	Keys
1	Configure aggregation group.	**esr(config)# interface multilink <IF>**	<IF> – interface name.
2	Specify the description of configured aggregation group (optional).	**esr(config-multilink)# description <DESCRIPTION>**	<DESCRIPTION> – aggregation group description, set by the string of up to 255 characters.
3	Specify the time interval during which the statistics on the aggregation group load is averaged (optional).	**esr(config-multilink)# load-average <TIME>**	<TIME> – interval in seconds, takes values of [5..150]. Default value: 5.
4	Specify MTU (Maximum Transmission Unit) size for the aggregation group (optional). MTU above 1500 will be active only when using the 'system jumbo-frames' command.	**esr(config-multilink)# mtu <MTU>**	<MTU> – MTU value, takes values in the range of [1280..1500]. Default value: 1500.
5	Enable CHAP authentication.	**esr(config-multilink)# ppp authentication chap**	
6	Enable authentication override (optional).	**esr(config-multilink)# ppp chap refuse**	

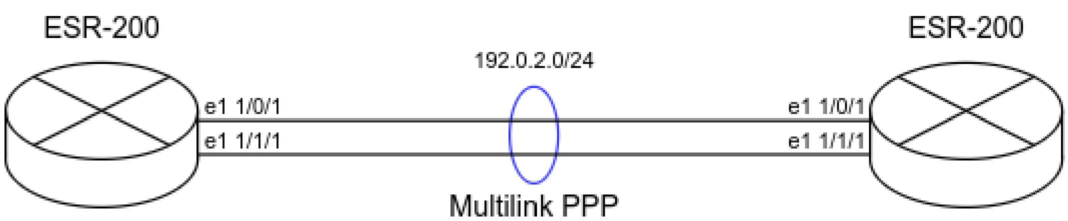
Step	Description	Command	Keys
7	Specify the router name that is sent to a remote party for CHAP authentication.	**esr(config-multilink)# ppp chap hostname <NAME>**	<NAME> – router name, set by the string of up to 31 characters
8	Specify the password that is sent with the router name to a remote party for CHAP authentication.	**esr(config-multilink)# ppp chap password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }**	<CLEAR-TEXT> – unencrypted password, set by the string of [8..64] characters, may include [0-9a-fA-F] characters. <ENCRYPTED-TEXT> – unencrypted password, set by the string of [16..128] characters.
9	Allow any non-null IP address to be accepted as a local IP address from the neighbour (optional).	**esr(config-multilink)# ppp ipcp accept-address**	
10	Set IP address that is sent to a remote party for the further allocation.	**esr(config-multilink)# ppp iccp remote-address <ADDR>**	<ADDR> – IP address of a remote gateway.
11	Specify a user for remote party authentication and switch to the specified user configuration mode.	**esr(config-multilink)# chap username <NAME>**	<NAME> – user name, set by the string of up to 31 characters.
12	Set encrypted or unencrypted password for a specific user to authenticate the remote party.	**esr(config-ppp-user)# password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }**	<CLEAR-TEXT> – unencrypted password, set by the string of [8..64] characters, may include [0-9a-fA-F] characters. <ENCRYPTED-TEXT> – unencrypted password, set by the string of [16..128] characters.
13	Set the amount of attempts to send Configure-Request packets before the remote peer is found to be unable to respond (optional).	**esr(config-multilink)# ppp max-configure <VALUE>**	<VALUE> – time in seconds, takes values of [1..255]. Default value: 10.
14	Set the amount of attempts to send Configure-NAK packets before all options are confirmed (optional).	**esr(config-multilink)# ppp max-failure <VALUE>**	<VALUE> – time in seconds, takes values of [1..255].
15	Set the amount of attempts to send Terminate-Request packets before the session is aborted (optional).	**esr(config-multilink)# ppp max-terminate <VALUE>**	<VALUE> – time in seconds, takes values of [1..255]. Default value: 2.

Step	Description	Command	Keys
16	Set MRU (Maximum Receive Unit) size for the interface.	**esr(config-multilink)# ppp mru <MRU>**	<MRU> – MRU value, takes values in the range of [128..1485]. Default value: 1500.
17	Specify the time interval in seconds after which the router sends a keepalive message (optional).	**esr(config-multilink)# ppp timeout keepalive <TIME>**	<TIME> – time in seconds, takes values of [1..32767]. Default value: 10.
18	Specify the time interval in seconds after which the router sends a keepalive message (optional).	**esr(config-multilink)# ppp timeout retry <TIME>**	<TIME> – time in seconds, takes values of [1..255]. Default value: 3.
19	Specify the maximum packet size for MLPP interface.	**esr(config-multilink)# mrru <MRRU>**	<MRRU> – maximum size of a received packet for MLPP interface, takes value in the range of [1500..10000].
20	Bind e1 port to the physical interface.	**esr(config-if-gi)# switchport e1 <SLOT>**	<SLOT> – slot identifier, takes values in the range of [0..3].
21	Put the physical port into SFPe1 module operation mode.	**esr(config-if-gi)# switchport mode e1**	
22	Enable MLPPP mode on E1 interface.	**esr(config-e1)# ppp multilink**	
23	Include E1 interface in the aggregation group.	**esr(config-e1)# ppp multilink-group <GROUP-ID>**	<GROUP-ID> – group identifier, takes values in the range of [1..4].

8.9.2 Configuration example

Objective:

Configure MLPPP connection to the opposite side with IP address 192.0.2.2/24 via e1 1/0/1 and e1 1/1/1 interfaces. To build an aggregated PPP link, gi 1/0/3 and gi 1/0/4 interfaces are used, into which TOPGATE-WAN-E1 is inserted.



Solution:

First, configure system jumbo-frames, save changes to the configuration and reboot the router:

```
esr# configure
esr(config)# system jumbo-frames
esr(config)# exit
esr# commit
esr# confirm
esr# reload system
Do you really want to reload system ? (y/N): y
```

Configure the gigabitethernet 1/0/3-4 physical interfaces that have TOPGATE-WAN-E1 installed. When configuring physical interfaces, specify mtu at least 1510, set the interface to the e1 mode, and specify the e1 channel:

```
esr# configure
esr(config)# interface gigabitethernet 1/0/3
esr(config-if-gi)# mode switchport
esr(config-if-gi)# mtu 1510
esr(config-if-gi)# switchport mode e1
esr(config-if-gi)# switchport e1 slot 0
esr(config-if-gi)# switchport e1 timeslots 1-31
esr(config-if-gi)# exit
esr(config)# interface gigabitethernet 1/0/4
esr(config-if-gi)# mode switchport
esr(config-if-gi)# mtu 1510
esr(config-if-gi)# switchport mode e1
esr(config-if-gi)# switchport e1 slot 1
esr(config-if-gi)# switchport e1 timeslots 1-31
esr(config-if-gi)# exit
```

Configure multilink interface:

```
esr(config)# interface multilink 3
esr(config-multilink)# ip address 192.0.2.1/24
esr(config-multilink)# security-zone trusted
esr(config-multilink)# exit
```

Bind E1 interfaces to Multilink PPP. When configuring E1 interface, specify multilink-group and enable multilink:

```
esr(config)# interface e1 1/0/1
esr(config-e1)# ppp multilink-group 3
esr(config-e1)# ppp multilink
esr(config-e1)# exit
esr(config)# interface e1 1/1/1
esr(config-e1)# ppp multilink-group 3
esr(config-e1)# ppp multilink
esr(config-e1)# exit
```

Information about the state of the multilink interface can be obtained using the following command:

```
esr# show interfaces status multilink 3
Interface 'mu1' status information:
 Description:           --
 Operational state:     Up
 Administrative state:  Up
 Track ID:              0
 Supports broadcast:    No
 Supports multicast:    Yes
 MTU:                   1492
 MAC address:           none
 Last change:           6 seconds
 Mode:                  routerport
```

8.10 Bridge configuration

Bridge is a method of connection for two Ethernet segments on data-link level without any higher level protocols, such as IP. Packet transmission is based on Ethernet addresses, not on IP addresses. Given that the transmission is performed on data-link level (Level 2 of the OSI model), higher level protocol traffic passes through the bridge transparently.

8.10.1 Configuration algorithm

Step	Description	Command	Keys
1	Add a network bridge to the system and switch to its configuration mode.	**esr(config)# bridge <BRIDGE-ID>**	<BRIDGE-ID> – bridge identification number, takes values in the range of: • for ESR-10/12V(F)/14VF/15 – [1..50]; • for ESR-20/21/30/100/200 – [1..250]; • for ESR-1000/1200/1500/1511/1700/3100/3200 – [1..500].
2	Enable network bridge.	**esr(config-bridge)# enable**	
3	Specify VRF instance, in which the given modem will operate (optional).	**esr(config-bridge)# ip vrf forwarding <VRF>**	<VRF> – VRF name, set by the string of up to 31 characters.
4	Specify the configured network bridge description (optional).	**esr(config-bridge)# description <DESCRIPTION>**	<DESCRIPTION> – network bridge description, set by the string of up to 255 characters.

Step	Description	Command	Keys
5	Connect sub interface, qinq interface, L2GRE tunnel or L2TPv3 tunnel with the network bridge. Connected interfaces/ tunnels and network bridges automatically become participants of the shared L2 domain (optional).	**esr(config-if-gi)# bridge-group <BRIDGE-ID>** **esr(config-if-l2tpv3)# bridge-group <BRIDGE-ID>**	<BRIDGE-ID> – bridge identification number, takes values in the range of: • for ESR-10/12V(F)/ 14VF/15 – [1..50]; • for ESR-20/21/30/100/200 – [1..250]; • for ESR-1000/1200/1500/1511/1700/3100/3200 – [1..500].
6	Connect the current network bridge with VLAN. All interfaces and L2 tunnels that are members of the assigned VLAN are automatically included in the network bridge and become members of the shared L2 domain (optional)	**esr(config-bridge)# vlan <VID>**	<VID> – VLAN identifier, set in the range of [1..4094].
7	Specify the size of MTU packets that can be passed by the bridge (optional; possible if only VLAN is included in the bridge). MTU above 1500 will be active only when using the 'system jumbo-frames' command.	**esr(config-bridge)# mtu <MTU>**	<MTU> – MTU value, takes values in the range of: • for ESR-10/12V(F)/14VF – [552..9600]; • for ESR-20/21 – [552..9500]; • for ESR-100/200/1000/1200/1500/1511/1700 – [552..10000] • for ESR-1500/1511/1700/3100 – [552..9190]. Default value: 1500
8	Specify the IPv4/IPv6 address and subnet mask for the interface to be configured or enable IP address obtain dynamically.	**esr(config-bridge)# ip address <ADDR/LEN>**	<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]. For advanced IPv4 addressing features see section IP addressing configuration.

Step	Description	Command	Keys
		esr(config-bridge)# ipv6 address <IPV6-ADDR/LEN>	<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]. For advanced IPv6 addressing features see section IPv6 addressing configuration. Several IPv4/IPv6 addresses can be specified separated by commas. Up to 8 IPv4/IPv6 addresses can be assigned to the interface.
		esr(config-bridge)# ip address dhcp	For advanced DHCP client operation features, see section DHCP Client management.
9	Disable the Firewall features on the interface or enable the interface in the security zone (see Firewall configuration).	**esr(config-bridge)# ip firewall disable**	
		esr(config-bridge)# security-zone <NAME>	<NAME> – security zone name, set by the string of up to 31 characters.
10	Enable recording of the current interface usage statistics (optional).	**esr(config-bridge)# history statistics**	
11	Specify the time interval during which the statistics on the bridge load is averaged (optional).	**esr(config-bridge)# load-average <TIME>**	<TIME> – interval in seconds, takes values of [5..150]. Default value: 5
12	Specify the network bridge MAC address different from a system one (optional).	**esr(config-bridge)# mac-address <ADDR>**	<ADDR> – network bridge MAC address, defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF].

Step	Description	Command	Keys
13	Enable interface isolation mode on the bridge. In this mode, the traffic exchange between members of the network bridge is prohibited. (Optional; relevant only for ESR-1000/1200/1500/1511/1700/3100)	**esr(config-bridge)# protected-ports [exclude vlan]**	exclude vlan – when specifying the given key, VLAN (connected with bridge) is excluded from the isolated interfaces list.
14	Prohibit unknown-unicast traffic switching (when a destination MAC address is not included in the switching table) in the given bridge. (Optional; relevant only for ESR-1000/1200/1500/1511/1700/3100)	**esr(config-bridge)# unknown-unicast-forwarding disable**	
15	Set the lifetime of IPv4/IPv6 entries in the ARP table studied on the given bridge (optional).	**esr(config-bridge)# ip arp reachable-time <TIME>** or **esr(config-bridge)# ipv6 nd reachable-time <TIME>**	<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>.

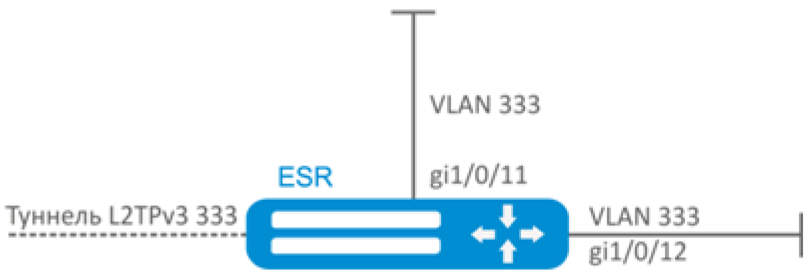
It is also possible to configure for the bridge interface:

- QoS in basic or advanced mode (see section QoS management);
- proxy (see section HTTP/HTTPS traffic proxying);
- traffic monitoring (see sections Netflow configuration and sFlow configuration);
- routing protocols functionality (see section Routing management);
- VRRF protocol (see section Redundancy management);
- BRAS functionality (see section BRAS (Broadband Remote Access Server) management);
- IDS/IPS functionality (see section IPS/IDS configuration).

8.10.2 Example of bridge configuration for VLAN and L2TPv3 tunnel

Objective:

Combine router interfaces related to LAN and L2TPv3 tunnel passing through the public network into a single L2 domain. For combining, use VLAN 333.



Solution:

Create VLAN 333:

```
esr(config)# vlan 333
esr(config-vlan)# exit
```

Create 'trusted' security zone:

```
esr(config)# security-zone trusted
esr(config-zone)# exit
```

Add gi1/0/11, gi1/0/12 interfaces to VLAN 333:

```
esr(config)# interface gigabitethernet 1/0/11-12
esr(config-if)# mode switchport
esr(config-if)# switchport general allowed vlan add 333 tagged
```

Create bridge 333, map VLAN 333 to it and specify membership in 'trusted' zone:

```
esr(config)# bridge 333
esr(config-bridge)# vlan 333
esr(config-bridge)# security-zone trusted
esr(config-bridge)# enable
```

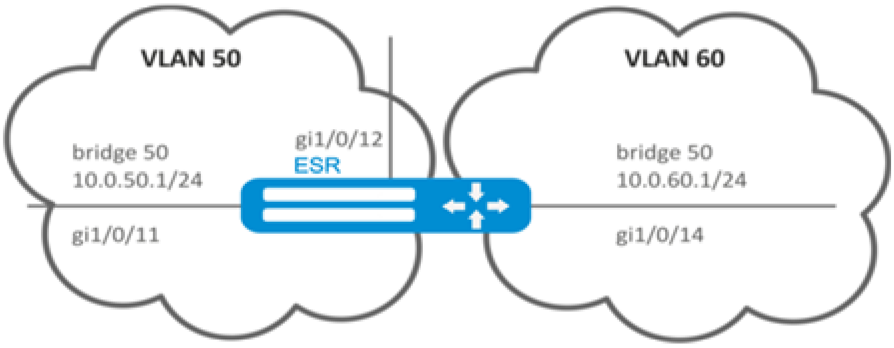
Specify the affiliation of L2TPv3 tunnel to bridge mapped to LAN (for L2TPv3 tunnel configuration, see Section L2TPv3 tunnel configuration). In general, bridge and tunnel identifiers should not match the VID, unlike this example.

```
esr(config)# tunnel l2tpv3 333
esr(config-l2tpv3)# bridge-group 333
```

8.10.3 Example of bridge configuration for VLAN

Objective:

Configure routing between VLAN 50 (10.0.50.0/24) and VLAN 60 (10.0.60.0/24). VLAN 50 should belong to 'LAN1', VLAN 60 – to 'LAN2', enable free traffic transmission between zones.



Solution:

Create VLAN 50, 60:

```
esr(config)# vlan 50,60
esr(config-vlan)# exit
```

Create 'LAN1' and 'LAN2' security zones:

```
esr(config)# security-zone LAN1
esr(config-zone)# exit
esr(config)# security-zone LAN2
esr(config-zone)# exit
```

Map VLAN 50 to gi1/0/11, gi1/0/12 interfaces:

```
esr(config)# interface gigabitethernet 1/0/11-12
esr(config-if-gi)# switchport general allowed vlan add 50 tagged
```

Map VLAN 60 to gi1/0/14 interface:

```
esr(config)# interface gigabitethernet 1/0/14
esr(config-if-gi)# switchport general allowed vlan add 60 tagged
```

Create bridge 50, map VLAN 50, define IP address 10.0.50.1/24 and membership in 'LAN1' zone:

```
esr(config)# bridge 50
esr(config-bridge)# vlan 50
esr(config-bridge)# ip address 10.0.50.1/24
esr(config-bridge)# security-zone LAN1
esr(config-bridge)# enable
```

Create bridge 60, map VLAN 60, define IP address 10.0.60.1/24 and membership in 'LAN2' zone:

```
esr(config)# bridge 60
esr(config-bridge)# vlan 60
esr(config-bridge)# ip address 10.0.60.1/24
esr(config-bridge)# security-zone LAN2
esr(config-bridge)# enable
```

Create firewall rules that enable free traffic transmission between zones:

```
esr(config)# security zone-pair LAN1 LAN2
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
esr(config)# security zone-pair LAN2 LAN1
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
esr(config)# exit
```

To view an interface membership in a bridge, use the following command:

```
esr# show interfaces bridge
```

8.10.4 Configuration example of the second VLAN tag adding/removing

Objective:

The gigabitethernet 1/0/1 interface receives Ethernet frames with various VLAN tags. It is necessary to redirect them to the gigabitethernet 1/0/2 interface, adding the second VLAN-ID 828. When Ethernet frames with VLAN-ID 828 come on the gigabitethernet 1/0/2, this tag must be removed and sent to the gigabitethernet 1/0/1 interface.

Solution:

Create the bridge without VLAN and IP address on the route:

```
esr(config)# bridge 1
esr(config-bridge)# enable
esr(config-bridge)# exit
```

Include the gigabitethernet 1/0/1 interface in bridge 1:

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# bridge-group 1
esr(config-if-gi)# exit
```

Include the gigabitethernet 1/0/2.828 sub interface in bridge 1:

```
esr(config)# interface gigabitethernet 1/0/2.828
esr(config-subif)# bridge-group 1
esr(config-subif)# exit
```

When adding the second VLAN tag to an Ethernet frame, its size is increased by 4 bytes. MTU must be increased by 4 bytes or more on the gigabitethernet 1/0/2 router interface and on all equipment transmitting Q-in-Q frames.

8.11 Dual-Homing configuration

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

Dual-Homing is a technology based on redundant links that creates a secure connection in order to prevent failures of the key network resources.

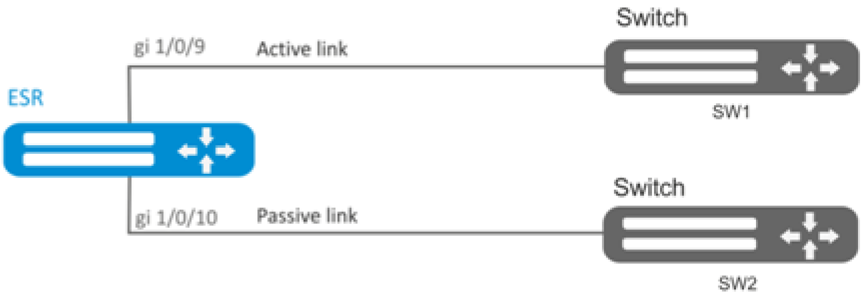
8.11.1 Configuration algorithm

Step	Description	Command	Keys
1	Specify a redundant interface to which the switching will occur when the connection is lost on a primary one.	**esr(config-if-gi)# backup interface<IF> vlan <VID>**	<IF> – interface to which the switching will be. <VID> – VLAN ID, set in the range of [2..4094]. The parameter can also be specified by the range with '-' or by comma-separated list.
2	Specify the number of packets copies with the same MAC address that will be sent to an active interface when switching (optional).	**esr(config)# backup-interface mac-duplicate <COUNT>**	<COUNT> – amount of packets copies, takes values of [1..4].
3	Specify the number of packets per second that will be sent to an active interface when switching (optional).	**esr(config)# backup-interfacemac-per-second<COUNT>**	<COUNT> – amount of MAC addresses per second, takes value of [50..400].
4	Specify that it is necessary to carry out the switching to the primary interface when restoring the communication (optional).	**esr(config)# backup-interface preemption**	

8.11.2 Configuration example

Objective:

Establish redundancy of the ESR router L2 connections for VLAN 50-55 using SW1 and SW2 devices.



Solution:

First, do the following:

Create VLAN 50, -55:

```
esr(config)# vlan 50-55
```

Disable STP for gigabitethernet 1/0/9 and gigabitethernet 1/0/10 interfaces, i.e. these protocols cannot operate simultaneously:

```
esr(config)# interface gigabitethernet 1/0/9-10
esr(config-if-gi)# spanning-tree disable
```

Add gigabitethernet 1/0/9 and gigabitethernet 1/0/10 interfaces into VLAN 50-55 in 'general' mode.

```
esr(config-if-gi)# switchport general allowed vlan add 50-55
esr(config-if-gi)# exit
```

Main configuration step:

Make gigabitethernet 1/0/10 redundant for gigabitethernet 1/0/9:

```
esr(config)# interface gigabitethernet 1/0/9
esr(config-if-gi)# backup interface gigabitethernet 1/0/10 vlan 50-55
```

To view information on redundant interfaces, use the following command:

```
esr# show interfaces backup
```

8.12 Mirroring configuration (SPAN/RSPAN)

In the current firmware version the RSPAN functionality is supported only by ESR-1000/1200/1500/1511/1700 routers.

Traffic mirroring is a feature of the router that allows for redirection of traffic from a specific port of the router to another port of the same router (local mirroring) or to a remote device (remote mirroring).

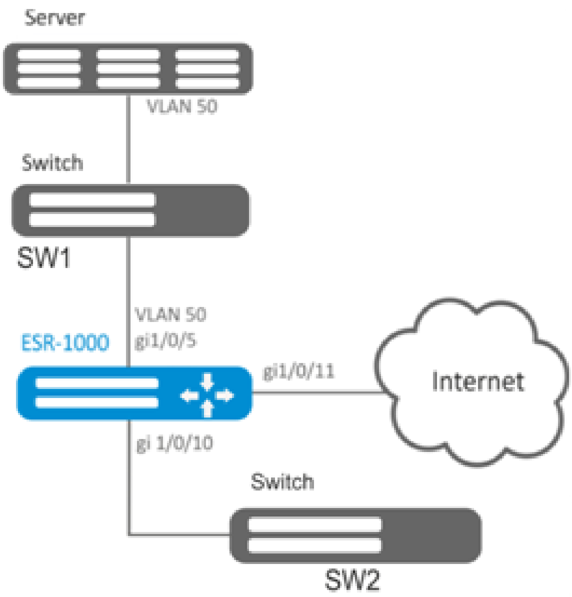
8.12.1 Configuration algorithm

Step	Description	Command	Keys
1	Define VLAN over which the mirrored traffic will be transmitted (in case of using remote mirroring).	**esr(config)# port monitor remote vlan <VID> <DIRECTION>**	<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.
2	Enable the remote mirroring mode (in case of using remote mirroring).	**esr(config)# port monitor remote**	
3	Define the mode of the port transmitting mirrored traffic (optional).	**esr(config)# port monitor mode <MODE>**	<MODE> – mode: • **network** – combined data transfer and mirroring (default); • **monitor-only** – mirroring only.
4	Enable mirroring in the interface configuration mode.	**esr(config-if-gi)# port monitor interface <IF> [<DIRECTION>]**	<IF> – interface from which the frames will be mirrored; <DIRECTION> – traffic direction: • **tx** – mirroring only output traffic; • **rx** – mirroring only input traffic;

8.12.2 Configuration example

Objective:

Establish remote mirroring of traffic through VLAN 50 from gi1/0/11 interface to be sent to server for processing purposes.



Solution:

First, do the following:

- Create VLAN 50:
- On gi 1/0/5 interface, add VLAN 50 in 'general' mode.

Main configuration step:

Specify VLAN that will be used for transmission of mirrored traffic:

```
esr1000(config)# port monitor remote vlan 50
```

For gi 1/0/5 interface, specify a port for mirroring:

```
esr1000(config)# interface gigabitethernet 1/0/5
esr1000(config-if-gi)# port monitor interface gigabitethernet 1/0/11
```

For gi 1/0/5 interface, specify the remote mirroring mode:

```
esr1000(config-if-gi)# port monitor remote
```

8.13 LACP configuration

LACP is a link aggregation protocol that allows multiple physical links to be combined into a single logical link. This process allows to increase the communication link bandwidth and robustness.

8.13.1 Configuration algorithm

Step	Description	Command	Keys
1	Set the system priority for LACP.	**esr(config)# lacp system-priority <PRIORITY>**	<PRIORITY> – priority, set in the range of [1..65535]. Default value: 1.
2	Set the load balancing mechanism for channel aggregation groups.	**esr(config)# port-channel load-balance { src-dst-mac-ip | src-dst-mac | src-dst-ip | src-dst-mac-ip-port }**	• **src-dst-mac-ip** – balancing mechanism is based on source and destination MAC addresses and IP addresses; • **src-dst-mac** – balancing mechanism is based on the MAC address of a sender and receiver; • **src-dst-ip** – balancing mechanism is based on the IP address of a sender and receiver; • **src - dst - mac - ip - port** – balancing mechanism is based on source and destination MAC address, IP address and port.
3	Set LACP administration timeout.	**esr(config)# lacp timeout {short | long }**	• **long** – long timeout; • **short** – short timeout. Default value: long.
4	Create and switch to the aggregated interface configuration mode.	**esr(config)# interface port-channel <ID>**	<ID> – sequence number of a channel aggregation group, takes values of [1..12].
5	Configure the required parameters of aggregated channel.		
6	Switch to the physical interface configuration mode.	**esr(config)# interface <IF-TYPE><IF-NUM>**	<IF-TYPE> interface type (gigabitethernet or tengigabitethernet). <IF-NUM> – F/S/P – F frame (1), S – slot (0), P – port.

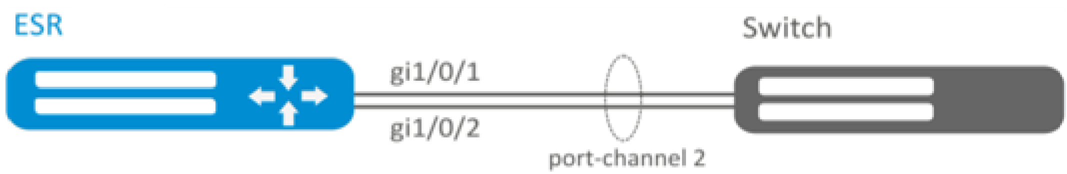
Step	Description	Command	Keys
7	Include a physical interface in the channel aggregation group specifying the mode of the channel aggregation group formation.	**esr(config-if-gi)# channel-group <ID> mode <MODE>**	<ID> – sequence number of a channel aggregation group, takes values of [1..12]. <MODE> – 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.
8	Set the Ethernet interface LACP priority.	**esr(config-if-gi)# lacp port-priority <PRIORITY>**	<PRIORITY> – priority, set in the range of [1..65535]. Default value: 1
9	Set the time interval during which statistics on the sub-interface load is collected (optional).	**esr(config-subif)# load-average <TIME>**	<TIME> – interval in seconds, takes values of [5..150].
10	Set the lifetime of IPv4/IPv6 entries in the ARP table studied on the given interface (optional).	**esr(config-subif)# ip arp reachable-time <TIME>** or **esr(config-subif)# ipv6 nd reachable-time <TIME>**	<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>.
11	Change MTU (MaximumTransmitionUnit) size. MTU above 1500 will be active only when using the 'system jumbo-frames' command (optional).	**esr(config-subif)# mtu <MTU>**	<MTU> – MTU value in bytes. Default value: 1500.
12	Enable recording of the current interface usage statistics (optional).	**esr(config-subif)# history statistics**	
13	Override the MSS (Maximum segment size) field in incoming TCP packets (optional).	**esr(config-subif)# ip tcp adjust-mss <MSS>** **esr(config-subif)# ipv6 tcp adjust-mss <MSS>**	<MSS> – MSS value, takes values in the range of [500..1460]. Default value: 1460

Step	Description	Command	Keys
It is also possible to configure the aggregated interface: · IPv4/IPv6 addressing (see sections IP addressing configuration, IPv6 addressing configuration and DHCP client management); · Firewall (see section Firewall configuration); · QoS in basic or advanced mode (see section QoS management); · proxy (see section HTTP/HTTPS traffic proxying); · traffic monitoring (see sections Netflow configuration and sFlow configuration); · routing protocols functionality (see section Routing management); · VRRF protocol (see section Redundancy management) · BRAS functionality (see section BRAS (Broadband Remote Access Server) management); · IDS/IPS functionality (see section IPS/IDS configuration).			

8.13.2 Configuration example

Objective:

Configure aggregated link between ESR router and the switch.



Solution:

First, do the following settings:

For gi1/0/1, gi1/0/2 interfaces disable security zone with 'no security-zone' command.

Main configuration step:

Create port-channel 2 interface:

```
esr(config)# interface port-channel 2
```

Add gi1/0/1, gi1/0/2 physical interfaces into the created link aggregation group:

```
esr(config)# interface gigabitethernet 1/0/1-2
esr(config-if-gi)# channel-group 2 mode auto
```

Further port-channel configuration is performed by analogy to the common physical interface.

8.14 AUX configuration

For ESR-21.

AUX configuration is used to specify parameters for interacting with external devices connected via serial interfaces to the ESR.

8.14.1 Configuration algorithm

Step	Description	Command	Keys
1	Switch to the serial interface configuration mode.	**esr(config)# line aux <NUM>**	<NUM> – a number of a serial interface from the range [1..3].

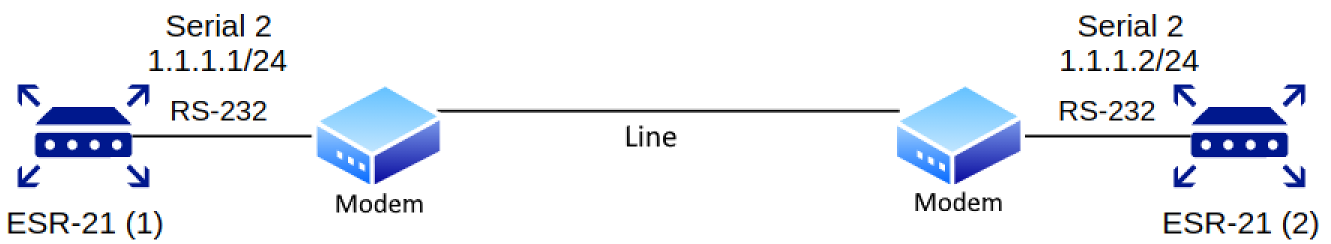
Step	Description	Command	Keys
2	Set the necessary serial interface parameters to communicate with the connected device (optional). These parameters are usually specified in the operation manual of the device to be connected. By default, the standard values will be used.	**esr(config-line-aux) databits <BITS>** **esr(config-line-aux) flowcontrol <FMODE>** **esr(config-line-aux) parity <PMODE>** **esr(config-line-aux) speed <SPEED>** **esr(config-line-aux) stopbits <STOP-BITS>**	<BITS> – a number of data bits sent [7..8]. Default is '8'. <FMODE> – data flow control mode. Takes the following values: • software – software flow control; • hardware – hardware flow control; • disabled – flow control disabled; Default is 'disabled'. <PMODE> – 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 is 'none'. <SPEED> – a speed of a serial interface in bps. Takes the following values: • 300; • 1200; • 2400; • 4800; • 9600; • 19200; • 38400; • 57600; • 115200; Default is '115200'. <STOP-BITS> – the number of stop bits transmitted[1..2]; Default is '1'.
3	Specify serial interface description (optional).	**esr(config-line-aux)# description <DESCRIPTION>**	<DESCRIPTION> – interface description, set by the string of up to 255 characters.

Step	Description	Command	Keys
4	When using the device to be connected as a modem, set the serial interface to modem mode (optional). Note: cannot be used in conjunction with the «transport telnet port» command.	**esr(config-line-aux)# modem inout**	
5	When using the ESR as a terminal server to control a connected device on the serial interface, set the TCP port number to be used as the TCP port number to connect to the ESR via telnet (optional). Note: cannot be used in conjunction with the 'modem inout' command.	**esr(config-line-aux)# transport telnet port <PORT>**	<PORT> – TCP port number for console server mode. Takes values in the range of [1..65535].

8.14.2 Configuration examples

Objective 1:

Configure IP communication between two ESRs on the serial port, using modems in Leased line mode (automatic modem mode), connected to each other by a telephone cable.



> ⚠ **Modems should be previously switched to automatic connection setting mode.**

> ⚠ **Modem compatibility verified**
> **Modem Zyxel U-336E Plus.**

Solution:

Configure the first ESR-21

Configure negotiation parameters:

```
esr-21-1(config)# line aux 2
esr-21-1(config-line-aux)# flowcontrol hardware
esr-21-1(config-line-aux)# exit
esr-21-1(config)#
```

Configure the required RS-232 interfaces:

```
esr-21-1(config)# interface serial 1/0/2
esr-21-1(config-serial)# ip address 1.1.1.1/24
esr-21-1(config-serial)# exit
esr-21-1(config)#
```

Configure firewall for security zones:

```
esr-21-1(config)# security zone xx
esr-21-1(config-zone)# exit
esr-21-1(config)# security zone-pair xx self
esr-21-1(config-zone-pair)#   rule 1
esr-21-1(config-zone-pair-rule)# action permit
esr-21-1(config-zone-pair-rule)# enable
esr-21-1(config-zone-pair-rule)# exit
esr-21-1(config-zone-pair)# exit
esr-21-1(config)#
```

Specify that the interfaces belong to the security zone:

```
esr-21-1(config)# interface serial 1/0/2
esr-21-1(config-serial)# security-zone xx
esr-21-1(config-serial)# exit
esr-21-1(config)#
```

Configure the second ESR-21

Configure negotiation parameters:

```
esr-21-2(config)# line aux 2
esr-21-2(config-line-aux)# flowcontrol hardware
esr-21-2(config-line-aux)# exit
esr-21-2(config)#
```

Configure the required RS-232 interfaces:

```
esr-21-2(config)# interface serial 1/0/2
esr-21-2(config-serial)# ip address 1.1.1.2/24
esr-21-2(config-serial)# exit
esr-21-2(config)#
```

Configure firewall for security zones:

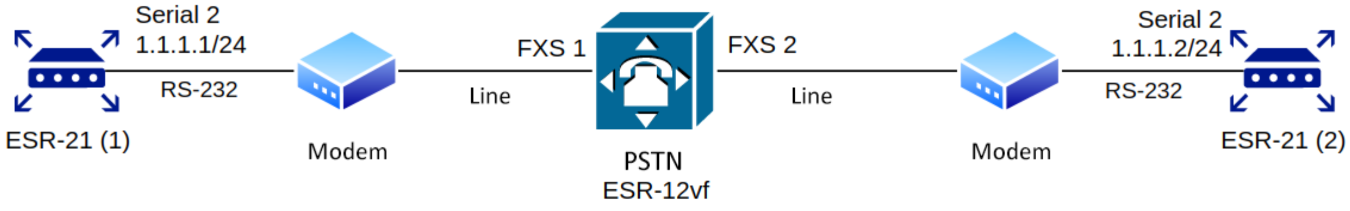
```
esr-21-2(config)# security zone xx
esr-21-2(config-zone)# exit
esr-21-2(config)# security zone-pair xx self
esr-21-2(config-zone-pair)#   rule 1
esr-21-2(config-zone-pair-rule)# action permit
esr-21-2(config-zone-pair-rule)# enable
esr-21-2(config-zone-pair-rule)# exit
esr-21-2(config-zone-pair)# exit
esr-21-2(config)#
```

Specify that the interfaces belong to the security zone:

```
esr-21-2(config)# interface serial 1/0/2
esr-21-2(config-serial)# security-zone xx
esr-21-2(config-serial)# exit
esr-21-2(config)#
```

Objective 2:

Set up IP connectivity between two ESRs on a Serial port, using Dial-Up modems and the Public Switched Telephone Network (PSTN).



The ESR-12VF with the following configuration is used as a PSTN emulation:

```
dialplan pattern factory_test
  description "dialplan for factory test"
  pattern "S5, L5 (00[1-3]@{local} | [xABCD*#].S)"
  enable
exit
sip profile 1
  dialplan pattern "factory_test"
  enable
  proxy primary
    enable
    ip address proxy-server 192.0.2.5
    registration
    ip address registration-server 192.0.2.5
  exit
exit
interface voice-port 1
  sip user phone 001
  profile sip 1
exit
interface voice-port 2
  sip user phone 002
  profile sip 1
  caller-id mode fsk-bell
exit
```

Modem compatibility verified

- **Modem ZyXEL OMNI 56K (MINI)**
- **Modem Acorp-M56SCD**

Solution:
Configure the first ESR-21

Configure the parameters for negotiation with the modem:

```
esr-21-1(config)# line aux 2
esr-21-1(config-line-aux)# flowcontrol hardware
esr-21-1(config-line-aux)# modem inout
esr-21-1(config-line-aux)# exit
esr-21-1(config)#
```

Configure the required RS-232 interfaces:

```
esr-21-1(config)# interface serial 1/0/2
esr-21-1(config-serial)# ip address 1.1.1.1/24
esr-21-1(config-serial)# exit
esr-21-1(config)#
```

Configure firewall for security zones:

```
esr-21-1(config)# security zone xx
esr-21-1(config-zone)# exit
esr-21-1(config)# security zone-pair xx self
esr-21-1(config-zone-pair)#   rule 1
esr-21-1(config-zone-pair-rule)# action permit
esr-21-1(config-zone-pair-rule)# enable
esr-21-1(config-zone-pair-rule)# exit
esr-21-1(config-zone-pair)# exit
esr-21-1(config)#
```

Specify that the interfaces belong to the security zone:

```
esr-21-1(config)# interface serial 1/0/2
esr-21-1(config-serial)# security-zone xx
esr-21-1(config-serial)# exit
esr-21-1(config)#
```

Enable dialing by number:

```
esr-21-1(config)# interface serial 1/0/2
esr-21-1(config-serial)# dialer string 002
esr-21-1(config-serial)# dialer
esr-21-1(config-serial)# exit
esr-21-1(config)#
```

Configure the second ESR-21

Configure negotiation parameters:

```
esr-21-2(config)# line aux 2
esr-21-2(config-line-aux)# flowcontrol hardware
esr-21-2(config-line-aux)# modem inout
esr-21-2(config-line-aux)# exit
esr-21-2(config)#
```

Configure the required RS-232 interfaces:

```
esr-21-2(config)# interface serial 1/0/2
esr-21-2(config-serial)# ip address 1.1.1.2/24
esr-21-2(config-serial)# exit
esr-21-2(config)#
```

Configure firewall for security zones:

```
esr-21-2(config)# security zone xx
esr-21-2(config-zone)# exit
esr-21-2(config)# security zone-pair xx self
esr-21-2(config-zone-pair)#   rule 1
esr-21-2(config-zone-pair-rule)# action permit
esr-21-2(config-zone-pair-rule)# enable
esr-21-2(config-zone-pair-rule)# exit
esr-21-2(config-zone-pair)# exit
esr-21-2(config)#
```

Specify that the interfaces belong to the security zone:

```
esr-21-2(config)# interface serial 1/0/2
esr-21-2(config-serial)# security-zone xx
esr-21-2(config-serial)# exit
esr-21-2(config)#
```

Objective 3:

Use additional modem settings for Objective 2:

- for modem 1 enable the V.22bis protocol;
- disable the speakers on both modems.

Solution

Create a line with additional modem initialization parameters for the first ESR-21, where

- AT&N1 – enable V.22bis on modem mode;
- ATM0L0 – disable modem speaker.

```
esr-21-1(config)# chat-script dial_test "ABORT 'BUSY' ABORT 'NO CARRIER' ABORT ERROR '' AT OK
AT&F OK AT&N14 OK ATM0L0 OK ATD\\T CONNECT ''"
esr-21-1(config)#
```

Enable the use of the modem initialization string:

```
esr-21-1(config)# interface serial 1/0/2
esr-21-1(config-serial)# dialer string 001 modem-script dial_test
esr-21-1(config-serial)# exit
esr-21-1(config)#
```

Create a line with additional modem initialization parameters for the second ESR-21:

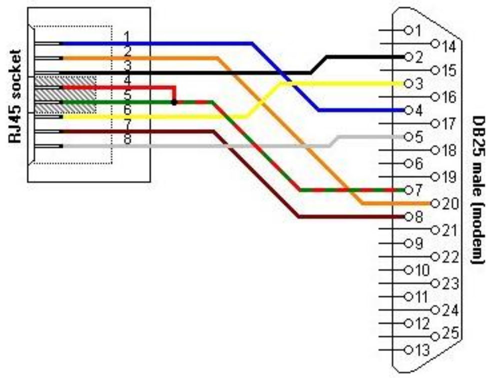
```
esr-21-2(config)# chat-script answer_test "ABORT 'BUSY' ABORT 'NO CARRIER' '' AT OK AT&F OK
ATM0L0 RING ATAr CONNECT ''"
esr-21-2(config)#
```

Enable the use of the modem initialization string:

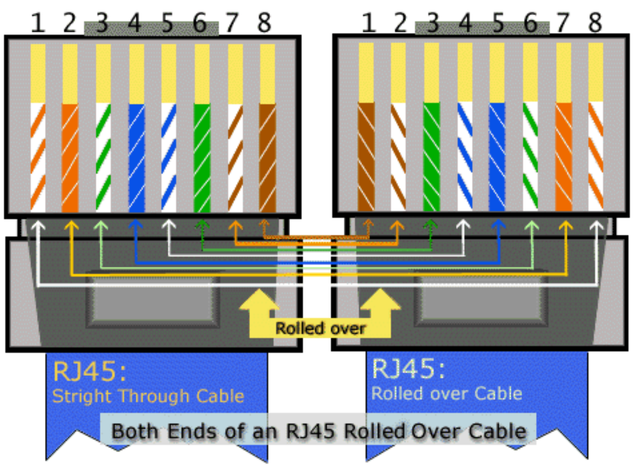
```
esr-21-2(config)# interface serial 1/0/2
esr-21-2(config-serial)# dialer string 000 modem-script answer_test
esr-21-2(config-serial)# exit
esr-21-2(config)#
```

8.14.3 Adapter soldering schemes

RJ-45 <--> DB-25 pinout



RJ-45 <--> RJ-45 pinout (rolled over cable)



9 Tunneling management

9.1 GRE tunnel configuration

GRE (*Generic Routing Encapsulation*) is a network packet tunneling protocol. Its main purpose is to encapsulate packets of the OSI model network layer into IP packets. GRE may be used for VPN establishment on 3rd level of OSI model. In ESR router implemented static unmanageable GRE tunnels, i.e. tunnels are created manually via configuration on local and remote hosts. Tunnel parameters for each side should be mutually agreeable, otherwise transferred data will not be decapsulated by the partner.

9.1.1 Configuration algorithm

Step	Description	Command	Keys
1	Configure L3 interface from which a GRE tunnel will be built.		
2	Create a GRE tunnel and switch to its configuration mode.	**esr(config)# tunnel gre <INDEX>**	<INDEX> – tunnel identifier, set in the range of: • for ESR-10/12V(F)/14VF/15 – [1..10]; • for ESR-20/21/30/100/200 – [1..250]; • for ESR-1000/1200/1500/1511/1700/3100/3200 – [1..500].

Step	Description	Command	Keys
3	Specify VRF instance, in which the given GRE tunnel will operate (optional).	**esr(config-gre)# ip vrf forwarding <VRF>**	<VRF> – VRF name, set by the string of up to 31 characters.
4	Specify the description of the configured tunnel (optional).	**esr(config-gre)# description <DESCRIPTION>**	<DESCRIPTION> – tunnel description, set by the string of up to 255 characters.
5	Set local IP address for tunnel installation.	**esr(config-gre)# local address <ADDR>**	<ADDR> – gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
		esr(config-gre)# local interface <IF>	<IF> – interface IP address of which is used for the tunnel installation.
6	Set remote IP address for tunnel installation.	**esr(config-gre)# remote address <ADDR>**	<ADDR> – gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
7	Specify the GRE tunnel encapsulation mode.	**esr(config-gre)# mode <MODE>**	<MODE> – GRE tunnel encapsulation mode: • **ip** – encapsulation of IP in GRE; • **ethernet** – encapsulation of Ethernet frames in GRE. Default value: ip
8	Set the IP address of a tunnel local side (only in ip mode).	**esr(config-gre)# ip address <ADDR/LEN>**	<ADDR/LEN> – IP address and prefix of a 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]. Up to 8 IP addresses can be specified separated by commas. For advanced IPv4 addressing features see section IP addressing configuration.

Step	Description	Command	Keys
9	Assign the broadcast domain for encapsulation in the tunnel's GRE packets (only in ethernet mode).	**esr(config-gre)# bridge-group <BRIDGE-ID>**	<BRIDGE-ID> – bridge identification number, takes values in the range of: • for ESR-10/12V(F)/14VF/15 – [1..50]; • for ESR-20/21/30/100/200 – [1..250]; • for ESR-1000/1200/1500/1511/1700/3100/3200 – [1..500]
10	Include the GRE tunnel in a security zone and configure interaction rules between zones or disable firewall (see section Firewall configuration).	**esr(config-gre)# security-zone<NAME>**	<NAME> – security zone name, set by the string of up to 12 characters.
		esr(config-gre)# ip firewall disable	
11	Specify MTU size (MaximumTransmissionUnit) for the tunnel (optional). MTU above 1500 will be active only when using the 'system jumbo-frames' command.	**esr(config-gre)# mtu <MTU>**	<MTU> – MTU value, takes values in the range of: • for ESR-10/12V(F)/14VF/15 – [1280..9600]; • for ESR-20/21/30 – [1280..9500]; • for ESR-100/200/1000/1200/1500/1511/1700/3100/3200 – [1280..10000]. Default value: 1500.
12	Specify the TTL lifetime for tunnel packets (optional).	**esr(config-gre)# ttl <TTL>**	<TTL> – TTL value, takes values in the range of [1..255]. Default value: Inherited from encapsulated packet.
13	Specify DSCP for the use in IP header of encapsulated packet (optional).	**esr(config-gre)# dscp <DSCP>**	<DSCP> – DSCP code value, takes values in the range of [0..63]. Default value: inherited from encapsulated packet.

Step	Description	Command	Keys
14	Enable key transmitting in GRE tunnel header (according to RFC 2890) and set the key value. Configured only on the both tunnel sides (optional).	**esr(config-gre)# key <KEY>**	<KEY> – KEY value, takes values in the range of [1..2000000]. Default value: key is not transmitted.
15	Enable the calculation of the checksum and entry it to the GRE header of the packets to be sent. Also it is necessary to enable verifying of the checksum on the remote side (optional).	**esr(config-gre)# local checksum**	
16	Enable verification of the presence and consistency of checksum values in the headers of GRE packets being received. Also it is necessary to enable calculation of the checksum on the remote side (optional).	**esr(config-gre)# remote checksum**	
17	Enable the check for tunnel remote gateway availability (optional).	**esr(config-gre)# keepalive enable**	
18	Change the keepalive packets timeout from the opposing party (optional).	**esr(config-gre)# keepalive timeout <TIME>**	<TIME> – time in seconds, takes values of [1..32767]. Default value: 10.
19	Change the number of attempts to check the availability of a tunnel remote gateway (optional).	**esr(config-gre)# keepalive retries <VALUE>**	<VALUE> – number of attempts, takes values in the range of [1..255]. Default value: 5.
20	Specify the IP address for the keepalive mechanism (mandatory in ethernet mode).	**esr(config-gre)# keepalive dst-address <ADDR>**	<ADDR> – IP address to check GRE tunnel capability.
21	Change the time interval during which the statistics on the tunnel load is averaged (optional).	**esr(config-gre)# load-average <TIME>**	<TIME> – interval in seconds, takes values of [5..150]. Default value: 5.
22	Enable sending snmp-trap about tunnel enabling/disabling.	**esr(config-gre)# snmp init-trap**	

Step	Description	Command	Keys
23	Enable the mechanism of IP addresses iterative query using DHCP on the specified interfaces when the GRE tunnel is disconnected via keepalive (optional).	**esr(config-gre)# keepalive dhcp dependent-interface <IF>**	<IF> – physical/logical interface on which IP address obtaining via DHCP is enabled.
24	Specify the time interval between GRE tunnel disabling and IP address iterative query on the interface/ interfaces specified by the keepalive dhcp dependent-interface command (optional).	**esr(config-gre)# keepalive dhcp link-timeout <SEC>**	<SEC> – time interval between GRE tunnel disabling and IP address requery via DHCP on the interfaces.
25	Override the MSS (Maximum segment size) field in incoming TCP packets (optional).	**esr(config-gre)# ip tcp adjust-mss <MSS>**	<MSS> – MSS value, takes values in the range of [500..1460]. Default value: 1460.
26	Enable recording of the current tunnel usage statistics (optional).	**esr(config-gre)# history statistics**	
27	Enable the tunnel.	**esr(config-gre)# enable**	

It is also possible to configure the GRE tunnel:

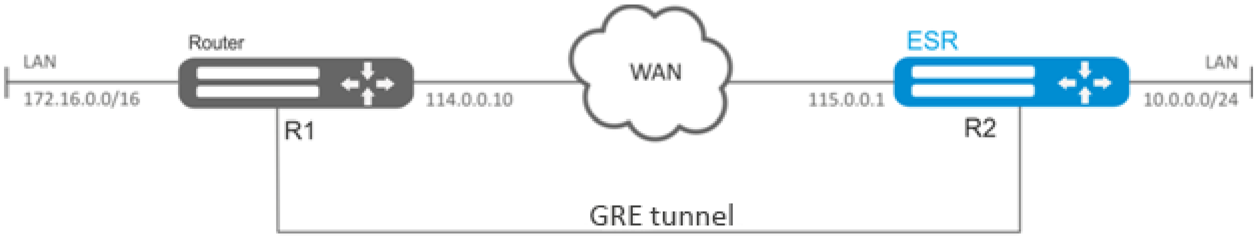
- QoS in basic or advanced mode (see section QoS management);
- proxy (see section HTTP/HTTPS traffic proxying);
- traffic monitoring (see sections Netflow configuration and sFlow configuration);
- routing protocols functionality (see section Routing management);
- BRAS functionality (see section BRAS (Broadband Remote Access Server) management).

9.1.2 IP-GRE tunnel configuration example

Objective:

Establish L3-VPN for company offices using IP network with GRE protocol for traffic tunneling.

- IP address 115.0.0.1 is used as a local gateway for the tunnel;
- IP address 114.0.0.10 is used as a remote gateway for the tunnel;
- IP address of the tunnel at the local side is 25.0.0.1/24.



Solution:

Pre-configure interfaces on the routers for connection with WAN, enable GRE packets reception from a security zone where WAN connected interfaces operate.

Create GRE 10 tunnel:

```
esr(config)# tunnel gre 10
```

Specify local and remote gateways (IP addresses of WAN border interfaces):

```
esr(config-gre)# local address 115.0.0.1
esr(config-gre)# remote address 114.0.0.10
```

Specify tunnel IP address 25.0.0.1/24:

```
esr(config-gre)# ip address 25.0.0.1/24
```

Also, the tunnel should belong to the security zone in order to create rules that allow traffic to pass through the firewall. To define the tunnel inherence to a zone, use the following command:

```
esr(config-gre)# security-zone untrusted
```

Enable tunnel:

```
esr(config-gre)# enable
esr(config-gre)# exit
```

Create route to the partner's local area network on the router. Specify previously created GRE tunnel as a destination interface.

```
esr(config)# ip route 172.16.0.0/16 tunnel gre 10
```

When settings are applied, traffic will be encapsulated into the tunnel and sent to the partner regardless of their GRE tunnel existence and settings validity.

Alternatively, the following parameters for GRE tunnel can be specified:

- Enable GRE header checksum calculation and inclusion into a packet with encapsulated packet for outbound traffic:

  ```
  esr(config-gre)# local checksum
  ```

- Enable check for GRE checksum presence and validity for inbound traffic:

  ```
  esr(config-gre)# remote checksum
  ```

- Specify a unique identifier:

```
esr(config-gre)# key 15808
```

- Specify DSCP, MTU, TTL values:

```
esr(config-gre)# dscp 44
esr(config-gre)# mtu 1426
esr(config-gre)# ttl 18
```

- Enable and configure keepalive mechanism:

```
esr(config-gre)# keepalive enable
esr(config-gre)# keepalive timeout <TIME>
esr(config-gre)# keepalive retries <VALUE>
```

To view the tunnel status, use the following command:

```
esr# show tunnels status gre 10
```

To view sent and received packet counters, use the following command:

```
esr# show tunnels counters gre 10
```

To view the tunnel configuration, use the following command:

```
esr# show tunnels configuration gre 10
```

IPv4-over-IPv4 tunnel configuration is performed in the same manner.

⚠ **During tunnel creation, enable GRE protocol (47) in the firewall.**

9.2 DMVPN configuration

DMVPN (*Dynamic Multipoint Virtual Private Network*) — technology for creating virtual private networks, with the ability to dynamically create tunnels between hosts. The advantage of this solution is its high scalability and ease of setup when connecting branches to the head office. DMVPN is used in the Hub-and-Spoke topology, and allows the construction of direct VPN Spoke-to-Spoke tunnels in addition to the usual Spoke-to-Hub tunnels. This means that branches can communicate with each other directly, without the need for traffic to pass through the Hub.

To establish such a connection, clients (NHC) over an encrypted IPsec tunnel send their internal (tunnel) address and external (NBMA) address to the NHRP server (NHS). When a client wants to connect to another NHC, it sends a request to the server to find out its external address. Having received a response from the server, the client can now independently establish a connection to the remote branch.

9.2.1 Configuration algorithm

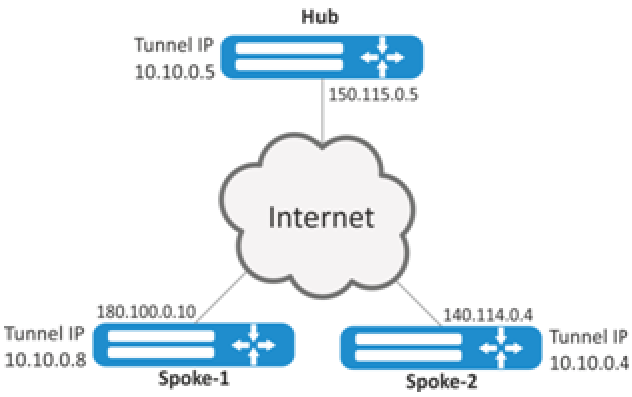
Step	Description	Command	Keys
1	Check the availability of 'external' IP addresses located on physical interfaces.		
2	Prepare IPsec tunnels for use with dynamic GRE tunnels.		See section Policy-based IPsec VPN configuration.
2	Create a GRE tunnel and switch to its configuration mode.	**esr(config)# tunnel gre <INDEX>**	<INDEX> – tunnel identifier.
3	Switch the GRE tunnel to multipoint mode.	**esr(config-gre)# multipoint**	
4	Set an open password for NHRP packets (optional).	**esr(config-gre)# ip nhrp authentication <WORD>**	<WORD> – unencrypted password, set by the string of [1..8] characters, may include [0-9a-fA-F] characters.
5	Specify the time during which a record about this client will exist on the NHS (optional).	**esr(config-gre)# ip nhrp holding-time <TIME>**	<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
6	Set the 'logic (tunnel)' address of the NHRP server.	**esr(config-gre)# ip nhrp nhs <ADDR> [no-registration]**	<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.
7	Match the 'internal' tunnel address with the 'external' NBMA address.	**esr(config-gre)# ip nhrp map <ADDR> <ADDR>**	<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Step	Description	Command	Keys
8	Define the destination of multicast traffic.	**esr(config-gre)# ip nhrp multicast { dynamic | nhs | <ADDR> }**	• **dynamic** — send to all peers with which there is a connection; • **nhs** — send to all static configured servers; <ADDR> – send to specifically configured server, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
9	Enable the ability to send NHRP Traffic Indication packets. Running on the NHS (optional).	**esr(config-gre)# ip nhrp redirect**	
10	Enable the ability to create shortest routes. Running on the NHC (optional).	**esr(config-gre)# ip nhrp shortcut**	
11	Map IPsec-VPN to the mGRE tunnel (optional).	**esr(config-gre)# ip nhrp ipsec <WORD> { static | dynamic }**	<WORD> – VPN name, set by the string of up to 31 characters. • **static** — static connection, used for connection to NHS; • **dynamic** — dynamically established connection, configured for communication between NHC.
12	Enable group attribute transmission (optional).	**esr(config-gre)# ip nhrp attribute group <WORD>**	
13	Enable NHRP.	**esr(config-gre)# ip nhrp enable**	
14	Organize IP connectivity using the dynamic routing protocol.		
Other settings are the same as for the static GRE tunnel (see section GRE tunnel configuration)			

9.2.2 Configuration example 1

Objective:

Organize DMVPN between company offices using mGRE tunnels, NHRP (Next Hop Resolution Protocol), Dynamic Routing Protocol (BGP), Ipsec. In our example, we will have a HUB router and two branches. The HUB is the DMVPN server (NHS), and the branches are DMVPN clients (NHC).



External IP addres of Hub — 150.115.0.5;

External IP address of Spoke-1 — 180.100.0.10;

External IP address of Spoke-2 — 140.114.0.4.

IPsec VPN parameters:

IKE:

- Diffie-Hellman group: 2;
- encryption algorithm: AES128;
- authentication algorithm: SHA1.

IPSEC:

- encryption algorithm: AES128;
- authentication algorithm: SHA1.

Solution:

1. Hub configuration
   Create GRE tunnel:

   ```
   esr# configure
   esr(config)# tunnel gre 5
   ```

   Specify the IP address of the interface bordering the ISP:

   ```
   esr(config-gre)# local address 150.115.0.5
   ```

   Specify MTU value:

   ```
   esr(config-gre)# mtu 1416
   ```

   Specify ttl value:

   ```
   esr(config-gre)# ttl 16
   ```

   Specify IP address of GRE tunnel:

   ```
   esr(config-gre)# ip address 10.10.0.5/24
   ```

   Switch the GRE tunnel into multipoint mode to be able to connect to multiple points:

   ```
   esr(config-gre)# multipoint
   ```

Proceed to NHRP configuration. Configure multicast to dynamically learnt addresses:

```
esr(config-gre)# ip nhrp multicast dynamic
```

Configure the dynamic routing protocol for the Hub. In our example, this will be BGP:

```
esr(config)# router bgp 65005
esr(config-bgp)# address-family ipv4
esr(config-bgp-af)# neighbor 10.10.0.8
esr(config-bgp-neighbor)# remote-as 65008
esr(config-bgp-neighbor)# enable
esr(config-bgp-neighbor)# exit
esr(config-bgp-af)# neighbor 10.10.0.4
esr(config-bgp-neighbor)# remote-as 65004
esr(config-bgp-neighbor)# enable
esr(config-bgp-neighbor)# exit
esr(config-bgp-af)# enable
```

Configure IPsec for the Hub:

```
esr(config)# security ike proposal IKEPROP
esr(config-ike-proposal)# encryption algorithm aes128
esr(config-ike-proposal)# dh-group 2
esr(config-ike-proposal)# exit
```

```
esr(config)# security ike policy IKEPOLICY
esr(config-ike-policy)# pre-shared-key ascii-text encrypted 8CB5107EA7005AFF
esr(config-ike-policy)# proposal IKEPROP
esr(config-ike-policy)# exit
```

```
esr(config)# security ike gateway IKEGW
esr(config-ike-gw)# ike-policy IKEPOLICY
esr(config-ike-gw)# local address 150.115.0.5
esr(config-ike-gw)# local network 150.115.0.5/32 protocol gre
esr(config-ike-gw)# remote address any
esr(config-ike-gw)# remote network any
esr(config-ike-gw)# mode policy-based
esr(config-ike-gw)# exit
```

```
esr(config)# security ipsec proposal IPSECPROP
esr(config-ipsec-proposal)# encryption algorithm aes128
esr(config-ipsec-proposal)# exit
```

```
esr(config)# security ipsec policy IPSECPOLICY
esr(config-ipsec-policy)# proposal IPSECPROP
esr(config-ipsec-policy)# exit
```

```
esr(config)# security ipsec vpn IPSECVPN
esr(config-ipsec-vpn)# mode ike
esr(config-ipsec-vpn)# ike establish-tunnel route
esr(config-ipsec-vpn)# ike gateway IKEGW
esr(config-ipsec-vpn)# ike ipsec-policy IPSECPOLICY
esr(config-ipsec-vpn)# enable
```

Map IPsec to the GRE tunnel so that clients can establish an encrypted connection:

```
esr(config-gre)# ip nhrp ipsec IPSECVPN dynamic
```

Enable NHRP and the tunnel:

```
esr(config-gre)# ip nhrp enable
esr(config-gre)# enable
```

2. Spoke configuration
   Perform the standard DMVPN configuration on the tunnel:

```
esr# configure
esr(config-gre)# tunnel gre 8
esr(config-gre)# mtu 1416
esr(config-gre)# ttl 16
esr(config-gre)# multipoint
esr(config-gre)# local address 180.100.0.10
esr(config-gre)# ip address 10.10.0.8/24
```

Specify the time while the client record will be stored on the server:

```
esr(config-gre)# ip nhrp holding-time 300
```

Specify the tunnel address of NHS:

```
esr(config-gre)# ip nhrp nhs 10.10.0.5/24
```

Specify the tunnel address – real:

```
esr(config-gre)# ip nhrp map 10.10.0.5 150.115.0.5
```

Configure the multicast to the NHRP server:

```
esr(config)# ip nhrp multicast nhs
```

Configure the BGP for spoke:

```
esr(config)# router bgp 65008
esr(config-bgp)# address-family ipv4
esr(config-bgp-af)# neighbor 10.10.0.5
esr(config-bgp-neighbor)# remote-as 65005
esr(config-bgp-neighbor)# enable
esr(config-bgp-neighbor)# exit
esr(config-bgp-af)# enable
```

Configure IPsec. When creating the IKE protocol gateway for NHS, specify particular destination addresses. When creating an IKE gateway for NHC – the destination address will be any:

```
esr(config)# security ike proposal IKEPROP
esr(config-ike-proposal)# encryption algorithm aes128
esr(config-ike-proposal)# dh-group 2
esr(config-ike-proposal)# exit
```

```
esr(config)# security ike policy IKEPOLICY
esr(config-ike-policy)# pre-shared-key ascii-text encrypted 8CB5107EA7005AFF
esr(config-ike-policy)# proposal IKEPROP
esr(config-ike-policy)# exit
```

```
esr(config)# security ike gateway IKEGW_HUB
esr(config-ike-gw)# ike-policy IKEPOLICY
esr(config-ike-gw)# local address 180.100.0.10
esr(config-ike-gw)# local network 180.100.0.10/32 protocol gre
esr(config-ike-gw)# remote address 150.115.0.5
esr(config-ike-gw)# remote network 150.115.0.5/32 protocol gre
esr(config-ike-gw)# mode policy-based
esr(config-ike-gw)# exit
```

```
esr(config)# security ike gateway IKEGW_SPOKE
esr(config-ike-gw)# ike-policy IKEPOLICY
esr(config-ike-gw)# local address 180.100.0.10
esr(config-ike-gw)# local network 180.100.0.10/32 protocol gre
esr(config-ike-gw)# remote address any
esr(config-ike-gw)# remote network any
esr(config-ike-gw)# mode policy-based
esr(config-ike-gw)# exit
```

```
esr(config)# security ipsec proposal IPSECPROP
esr(config-ipsec-proposal)# encryption algorithm aes128
esr(config-ipsec-proposal)# exit
```

```
esr(config)# security ipsec policy IPSECPOLICY
esr(config-ipsec-policy)# proposal IPSECPROP
esr(config-ipsec-policy)# exit
```

```
esr(config)# security ipsec vpn IPSECVPN_HUB
esr(config-ipsec-vpn)# mode ike
esr(config-ipsec-vpn)# ike establish-tunnel route
esr(config-ipsec-vpn)# ike gateway IKEGW_HUB
esr(config-ipsec-vpn)# ike ipsec-policy IPSECPOLICY
esr(config-ipsec-vpn)# enable
```

```
esr(config)# security ipsec vpn IPSECVPN_SPOKE
esr(config-ipsec-vpn)# mode ike
esr(config-ipsec-vpn)# ike establish-tunnel route
esr(config-ipsec-vpn)# ike gateway IKEGW_SPOKE
esr(config-ipsec-vpn)# ike ipsec-policy IPSECPOLICY
esr(config-ipsec-vpn)# enable
```

Map IPsec to the GRE tunnel, in order to be able to establish an encrypted connection with the server and with other network clients:

```
esr(config-gre)# ip nhrp ipsec IPSECVPN_HUB static
esr(config-gre)# ip nhrp ipsec IPSECVPN_SPOKE dynamic
```

Enable NHRP and the tunnel:

```
esr(config-gre)# ip nhrp enable
esr(config-gre)# enable
```

To view the NHRP records status, use the following command:

```
esr# show ip nhrp
```

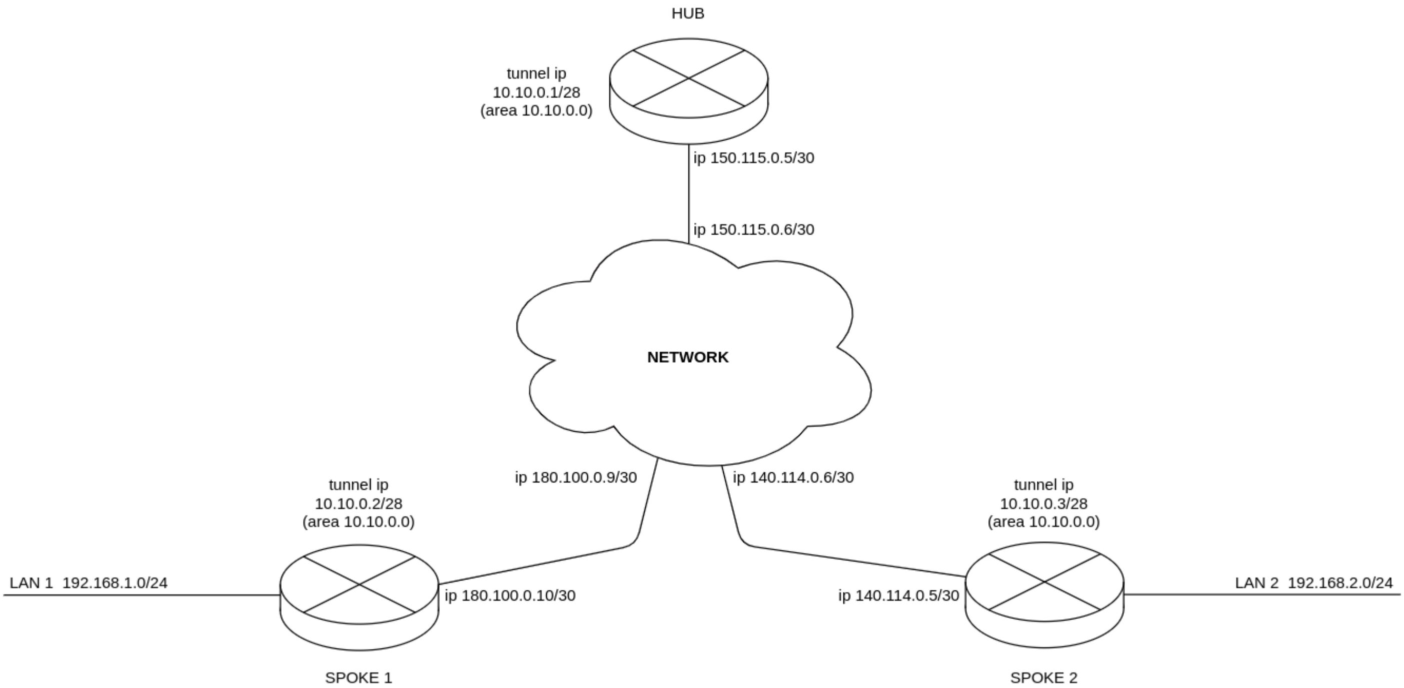
NHRP records can be cleared with the following command:

```
esr# clear ip nhrp
```

9.2.3 Configuration example 2

Objective:

Organize DMVPN between company offices with corresponding subnets LAN1 and LAN2, using mGRE tunnels, NHRP (Next Hop Resolution Protocol), Dynamic Routing Protocol (OSPF), IPsec. In our example, we will have a HUB router and two branches. The HUB is the DMVPN server (NHS), and the branches are DMVPN clients (NHC).



When using the DMVPN scheme, the HUB must be a DR router. Thus, the routes of local subnets spoke 1 and spoke 2 will be relayed through the hub.

External IP address of HUB — 150.115.0.5;
External IP address of Spoke-1 — 180.100.0.10;
External IP address of Spoke-2 — 140.114.0.4.

IPsec VPN parameters:

IKE:

- Diffie-Hellman group: 2;
- encryption algorithm: AES128;
- authentication algorithm: MD5.

IPSEC:

- Diffie-Hellman group: 2;
- encryption algorithm: AES128;
- authentication algorithm: MD5.

Solution:

1. HUB configuration:
   First, configure the OSPF protocol:

```
esr(config)# router ospf log-adjacency-changes
esr(config)# router ospf 1
esr(config-ospf)# router-id 77.77.77.77
esr(config-ospf)# area 10.10.0.0
esr(config-ospf-area)# enable
esr(config-ospf-area)# exit
esr(config-ospf)# enable
esr(config-ospf)# exit
```

   Configure the interface and identify its inherence to a security zone:

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# security-zone untrusted
esr(config-if-gi)# ip address 150.115.0.5/30
esr(config-if-gi)# exit
```

   Configure the GRE tunnel, define the security zone membership, configure OSPF on the GRE tunnel, configure NHRP and enable the tunnel and NHRP with the enable command. To make HUB DR, set the maximum priority.

```
esr(config)# tunnel gre 1
esr(config-gre)# ttl 16
esr(config-gre)# mtu 1416
esr(config-gre)# multipoint
esr(config-gre)# security-zone untrusted
esr(config-gre)# local address 150.115.0.5
esr(config-gre)# ip address 10.10.0.1/28
esr(config-gre)# ip ospf instance 1
esr(config-gre)# ip ospf area 10.10.0.0
esr(config-gre)# ip ospf priority 255
esr(config-gre)# ip ospf
esr(config-gre)# ip nhrp multicast dynamic
esr(config-gre)# ip nhrp enable
esr(config-gre)# enable
esr(config-gre)# exit
```

   Create static routes for the subnets of the spoke interfaces 180.100.0.8/30 and 140.114.0.4/30.

```
esr(config)# ip route 180.100.0.8/30 150.115.0.6
esr(config)# ip route 140.114.0.4/30 150.115.0.6
```

   Configure IPsec for the HUB.

```
esr(config)# security ike proposal ike_prop1
esr(config-ike-proposal)# authentication algorithm md5
esr(config-ike-proposal)# encryption algorithm aes128
esr(config-ike-proposal)# dh-group 2
esr(config-ike-proposal)# exit
```

```
esr(config)# security ike policy ike_pol1
esr(config-ike-policy)# pre-shared-key ascii-text password
esr(config-ike-policy)# proposal ike_prop1
esr(config-ike-policy)# exit
```

```
esr(config)# security ike gateway ike_spoke
esr(config-ike-gw)# ike-policy ike_pol1
esr(config-ike-gw)# local address 150.115.0.5
esr(config-ike-gw)# local network 150.115.0.5/32 protocol gre
esr(config-ike-gw)# remote address any
esr(config-ike-gw)# remote network any
esr(config-ike-gw)# mode policy-based
esr(config-ike-gw)# exit
```

```
esr(config)# security ipsec proposal ipsec_prop1
esr(config-ipsec-proposal)# authentication algorithm md5
esr(config-ipsec-proposal)# encryption algorithm aes128
esr(config-ipsec-proposal)# pfs dh-group 2
esr(config-ipsec-proposal)# exit
```

```
esr(config)# security ipsec policy ipsec_pol1
esr(config-ipsec-policy)# proposal ipsec_prop1
esr(config-ipsec-policy)# exit
```

```
esr(config)# security ipsec vpn ipsec_spoke
esr(config-ipsec-vpn)# mode ike
esr(config-ipsec-vpn)# ike establish-tunnel route
esr(config-ipsec-vpn)# ike gateway ike_spoke
esr(config-ipsec-vpn)# ike ipsec-policy ipsec_pol1
esr(config-ipsec-vpn)# enable
esr(config-ipsec-vpn)# exit
```

Map IPsec to the GRE tunnel so that clients can establish an encrypted connection:

```
esr(config)# tunnel gre 1
esr(config-gre)# ip nhrp ipsec ipsec_spoke dynamic
esr(config-gre)# exit
```

2. SPOKE configuration:
Preliminary configure the OSPF protocol with the advertising of the subnet LAN1:

```
esr(config)# router ospf log-adjacency-changes
esr(config)# router ospf 1
esr(config-ospf)# router-id 1.1.1.1
esr(config-ospf)# area 10.10.0.0
esr(config-ospf-area)# network 192.168.1.0/24
esr(config-ospf-area)# enable
esr(config-ospf-area)# exit
esr(config-ospf)# enable
esr(config-ospf)# exit
```

Configure the interface and identify its inherence to a security zone:

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# security-zone untrusted
esr(config-if-gi)# ip address 180.100.0.10/30
esr(config-if-gi)# exit
```

Configure the GRE tunnel, define the security zone membership, configure OSPF on the GRE tunnel, configure NHRP and enable the tunnel and NHRP with the enable command. To make HUB DR, set the minimum priority on spoke:

```
esr(config)# tunnel gre 1
esr(config-gre)# ttl 16
esr(config-gre)# mtu 1416
esr(config-gre)# multipoint
esr(config-gre)# ip firewall disable
esr(config-gre)# local address 180.100.0.10
esr(config-gre)# ip address 10.10.0.2/28
esr(config-gre)# ip ospf instance 1
esr(config-gre)# ip ospf area 10.10.0.0
esr(config-gre)# ip ospf priority 0
esr(config-gre)# ip ospf
esr(config-gre)# ip nhrp holding-time 300
esr(config-gre)# ip nhrp map 10.10.0.1 150.115.0.5
esr(config-gre)# ip nhrp nhs 10.10.0.1/28
esr(config-gre)# ip nhrp multicast nhs
esr(config-gre)# ip nhrp enable
esr(config-gre)# enable
esr(config-gre)# exit
```

Create static routes for the subnets of the spoke interfaces 180.100.0.8/30 and 140.114.0.4/30:

```
esr(config)# ip route 150.115.0.4/30 180.100.0.9
esr(config)# ip route 140.114.0.4/30 180.100.0.9
```

Configure IPsec for the HUB:

```
esr(config)# security ike proposal ike_prop1
esr(config-ike-proposal)# authentication algorithm md5
esr(config-ike-proposal)# encryption algorithm aes128
esr(config-ike-proposal)# dh-group 2
esr(config-ike-proposal)# exit
```

```
esr(config)# security ike policy ike_pol1
esr(config-ike-policy)# pre-shared-key ascii-text password
esr(config-ike-policy)# proposal ike_prop1
esr(config-ike-policy)# exit
```

```
esr(config)# security ike gateway ike_spoke
esr(config-ike-gw)# ike-policy ike_pol1
esr(config-ike-gw)# local address 180.100.0.10
esr(config-ike-gw)# local network 180.100.0.10/32 protocol gre
esr(config-ike-gw)# remote address any
esr(config-ike-gw)# remote network any
esr(config-ike-gw)# mode policy-based
esr(config-ike-gw)# exit
esr(config)# security ike gateway ike_hub
esr(config-ike-gw)# ike-policy ike_pol1
esr(config-ike-gw)# local address 180.100.0.10
esr(config-ike-gw)# local network 180.100.0.10/32 protocol gre
esr(config-ike-gw)# remote address 150.115.0.5
esr(config-ike-gw)# remote network 150.115.0.5/32 protocol gre
esr(config-ike-gw)# mode policy-based
esr(config-ike-gw)# exit
```

```
esr(config)# security ipsec proposal ipsec_prop1
esr(config-ipsec-proposal)# authentication algorithm md5
esr(config-ipsec-proposal)# encryption algorithm aes128
esr(config-ipsec-proposal)# pfs dh-group 2
esr(config-ipsec-proposal)# exit
```

```
esr(config)# security ipsec policy ipsec_pol1
esr(config-ipsec-policy)# proposal ipsec_prop1
esr(config-ipsec-policy)# exit
```

```
esr(config)# security ipsec vpn ipsec_spoke
esr(config-ipsec-vpn)# mode ike
esr(config-ipsec-vpn)# ike establish-tunnel route
esr(config-ipsec-vpn)# ike gateway ike_spoke
esr(config-ipsec-vpn)# ike ipsec-policy ipsec_pol1
esr(config-ipsec-vpn)# enable
esr(config-ipsec-vpn)# exit
esr(config)# security ipsec vpn ipsec_hub
esr(config-ipsec-vpn)# mode ike
esr(config-ipsec-vpn)# ike establish-tunnel route
esr(config-ipsec-vpn)# ike gateway ike_hub
esr(config-ipsec-vpn)# ike ipsec-policy ipsec_pol1
esr(config-ipsec-vpn)# enable
esr(config-ipsec-vpn)# exit
```

Map IPsec to the GRE tunnel, in order to be able to establish an encrypted connection with the server and with other network clients:

```
esr(config)# tunnel gre 1
esr(config-gre)# ip nhrp ipsec ipsec_hub static
esr(config-gre)# ip nhrp ipsec ipsec_spoke dynamic
esr(config-gre)# exit
```

3. To view the NHRP records status, use the following command.

```
esr# show ip nhrp
```

4. Additionally, in the security zone-pair untrusted self, the protocols for the GRE over IPSec tunnel must be allowed.

```
esr(config)# object-group service ISAKMP_PORT
esr(config-object-group-service)# port-range 500
esr(config-object-group-service)# poRt-range 4500
esr(config-object-group-service)# exit
esr(config)# security zone-pair untrusted self
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol udp
esr(config-zone-pair-rule)# match destination-port ISAKMP_PORT
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# rule 2
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol gre
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# rule 3
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol esp
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# rule 4
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol ah
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
```

9.3 L2TPv3 tunnel configuration

L2TPv3 (Layer 2 Tunnelling Protocol Version 3) is a protocol used for tunneling of 2nd level OSI model packets between two IP nodes. IP or UDP is used as an encapsulation protocol. L2TPv3 may be used as an alternative to MPLS P2P L2VPN (VLL) for L2 VPN establishment. In ESR router implemented static unmanageable L2TPv3 tunnels, i.e. tunnels are created manually via configuration on local and remote hosts. Tunnel parameters for each side should be mutually agreeable, otherwise transferred data will not be decapsulated by the partner.

9.3.1 Configuration algorithm

Step	Description	Command	Keys
1	Configure L3 interface from which a L2TPv3 tunnel will be built.		
2	Create a L2TPv3 tunnel and switch to its configuration mode.	**esr(config)# tunnel l2tpv3 <INDEX>**	<INDEX> – tunnel identifier, set in the range of: • for ESR-10/12V(F)/14VF/15 – [1..10]; • for ESR-20/21/30/100/200 – [1..250]; • for ESR-1000/1200/1500/1511/1700/3100/3200 – [1..500].
3	Specify the description of the configured tunnel (optional).	**esr(config-l2tpv3)# description <DESCRIPTION>**	<DESCRIPTION> – tunnel description, set by the string of up to 255 characters.
4	Set local IP address for tunnel installation.	**esr(config-l2tpv3)# local address <ADDR>**	<ADDR> – gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
5	Set remote IP address for tunnel installation.	**esr(config-l2tpv3)# remote address <ADDR>**	<ADDR> – gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
6	Select encapsulation method for L2TPv3 tunnel.	**esr(config-l2tpv3)# protocol <TYPE>**	<TYPE> – encapsulation type, possible values: • **ip** – encapsulation in an IP packet; • **udp** – encapsulation in UDP datagrams.
7	Set local session identifier.	**esr(config-l2tpv3)# local session-id <SESSION-ID>**	<SESSION-ID> – session identifier, takes values in the range of [1..200000].

8	Set remote session identifier.	**esr(config-l2tpv3)# remote session-id <SESSION-ID>**	<SESSION-ID> – session identifier, takes values in the range of [1..200000].
9	Define local UDP port (if UDP was selected as encapsulation method).	**esr(config-l2tpv3)# local port <UDP>**	<UDP> – UDP port number in the range of [1..65535].
10	Define remote UDP port (if UDP was selected as encapsulation method).	**esr(config-l2tpv3)# remote port <UDP>**	<UDP> – UDP port number in the range of [1..65535].
11	Assign the broadcast domain for encapsulation in the tunnel's L2TPV3 packets.	**esr(config-l2tpv3)# bridge-group <BRIDGE-ID>**	<BRIDGE-ID> – bridge identification number, takes values in the range of: • for ESR-10/12V(F)/14VF/15 – [1..50]; • for ESR-20/21/30/100/200 – [1..250]; • for ESR-1000/1200/1500/1511/1700/3100/3200 – [1..500].
12	Enable the tunnel.	**esr(config-l2tpv3)# enable**	
13	Specify MTU size (MaximumTransmissionUnit) for the tunnels (optional). MTU above 1500 will be active only when using the 'system jumbo-frames' command.	**esr(config-l2tpv3)# mtu <MTU>**	<MTU> – MTU value, takes values in the range of: • for ESR-10/12V(F)/14VF/15 – [1280..9600]; • for ESR-20/21/30 – [1280..9500]; • for ESR-100/200/1000/1200/1500/1511/1700/3100/3200 – [1280..10000]. Default value: 1500.
14	Define the local cookie value to check the conformance of data being transmitted and session (optional).	**esr(config-l2tpv3)# local cookie <COOKIE>**	<COOKIE> – COOKIE value, the parameter takes values of 8 or 16 characters in hexadecimal form.
15	Define the remote cookie value to check the conformance of data being transmitted and session (optional).	**esr(config-l2tpv3)# remote cookie <COOKIE>**	<COOKIE> – COOKIE value, the parameter takes values of 8 or 16 characters in hexadecimal form.
16	Specify the time interval during which the statistics on the tunnel load is averaged (optional).	**esr(config-l2tpv3)# load-average <TIME>**	<TIME> – interval in seconds, takes values of [5..150]. Default value: 5.

17	Enable recording of the current tunnel usage statistics (optional).	**esr(config-subif)# history statistics**	

It is also possible to configure the L2TPv3 tunnel:

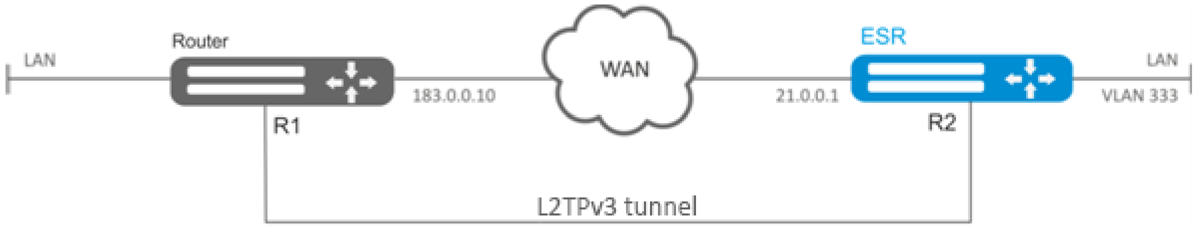
- QoS in basic or advanced mode (see section QoS management);
- BRAS functionality (see section BRAS (Broadband Remote Access Server) management).

9.3.2 L2TPv3 tunnel configuration example

Objective:

Establish L2 VPN for company offices using IP network with L2TPv3 protocol for traffic tunneling.

- UDP is used as an encapsulation protocol, port number at the local side and port number at the partner's side is 519;
- IP address 21.0.0.1 is used as a local gateway for the tunnel;
- IP address 183.0.0.10 is used as a remote gateway for the tunnel;
- Tunnel identifier at the local side equals 2, at the partner's side - 3;
- Tunnel identifier inside the tunnel equals 100, at the partner's side - 200;
- Forward traffic into the tunnel from the bridge with identifier 333.



Solution:

Create L2TPv3 333 tunnel:

```
esr# configure
esr(config)# tunnel l2tpv3 333
```

Specify local and remote gateways (IP addresses of WAN border interfaces):

```
esr(config-l2tpv3)# local address 21.0.0.1
esr(config-l2tpv3)# remote address 183.0.0.10
```

Specify the type of encapsulating protocol and UDP port numbers:

```
esr(config-l2tpv3)# protocol udp
esr(config-l2tpv3)# local port 519
esr(config-l2tpv3)# remote port 519
```

Specify identifiers for session inside the tunnel for local and remote sides:

```
esr(config-l2tpv3)# local session-id 100
esr(config-l2tpv3)# remote session-id 200
```

Define the inherence of L2TPv3 tunnel to a bridge that should be mapped to remote office network (for bridge configuration, see Section Configuration example of bridge for VLAN and L2TPv3 tunnel):

```
esr(config-l2tpv3)# bridge-group 333
```

Enable previously created tunnel and exit:

```
esr(config-l2tpv3)# enable
esr(config-l2tpv3)# exit
```

Create sub-interface for switching of traffic coming from the tunnel into LAN with VLAN id 333:

```
esr(config)# interface gi 1/0/2.333
```

Define the inherence of sub-interface to a bridge that should be mapped to LAN (for bridge configuration, see Section Configuration of PPP via E1):

```
esr(config-subif)# bridge-group 333
esr(config-subif)# exit
```

When settings are applied, traffic will be encapsulated into the tunnel and sent to the partner regardless of their L2TPv3 tunnel existence and settings validity.

Tunnel settings for the remote office should mirror local ones. IP address 183.0.0.10 should be used as a local gateway. IP address 21.0.0.1 should be used as a remote gateway for the tunnel. Encapsulation protocol port number at the local side should be 520, at the partner's side – 519. Session identifier inside the tunnel should be equal to 200, at the partner's side – 100. Also, the tunnel should belong to a bridge that should be connected with the partner's network.

To view the tunnel status, use the following command:

```
esr# show tunnels status l2tpv3 333
```

To view sent and received packet counters, use the following command:

```
esr# show tunnels counters l2tpv3 333
```

To view the tunnel configuration, use the following command:

```
esr# show tunnels configuration l2tpv3 333
```

In addition to tunnel creation, enable UDP inbound traffic in the firewall with source port 519 and destination port 519.

9.4 IPsec VPN configuration

IPsec is a set of protocols that enable security features for data transferred via IP protocol. This set of protocols allows for identity validation (authentication), IP packet integrity check and encryption, and also includes protocols for secure key exchange over the Internet.

9.4.1 Route-based IPsec VPN configuration algorithm

Step	Description	Command	Keys
1	Create a VTI tunnel and switch to its configuration mode.	**esr(config)# tunnel vti <TUN>**	<TUN> – device tunnel name.
2	Specify the local IP address of the VTI tunnel.	**esr(config-vti)#local address <ADDR>**	<ADDR> – IP address of a local gateway.
3	Specify the remote IP address of the VTI tunnel.	**esr(config-vti)#remote address <ADDR>**	<ADDR> – IP address of a remote gateway.
4	Specify the IP address of the VTI tunnel local side.	**esr(config-vti)# ip address <ADDR/LEN>**	<ADDR/LEN> – IP address and prefix of a 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].
5	Include the VTI tunnel in a security zone and configure interaction rules between zones or disable firewall for VTI tunnel.	**esr(config-vti)# security-zone<NAME>**	<NAME> – security zone name, set by the string of up to 12 characters.
		esr(config-vti)# ip firewall disable	
6	Enable the tunnel.	**esr(config-vti)#enable**	
7	Create an IKE profile and switch to its configuration mode.	**esr(config)# security ike proposal <NAME>**	<NAME> – IKE protocol name, set by the string of up to 31 characters.
8	Specify the description of the configured IKE profile (optional).	**esr(config-ike-proposal)# description<DESCRIPTION>**	<DESCRIPTION> – tunnel description, set by the string of up to 255 characters.
9	Specify IKE authentication algorithm (optional).	**esr(config-ike-proposal)# authentication algorithm <ALGORITHM>**	<ALGORITHM> – authentication algorithm, takes values of: md5, sha1, sha2-256, sha2_384, sha2-512. Default value: sha1.

Step	Description	Command	Keys
10	Specify IKE encryption algorithm (optional).	**esr(config-ike-proposal)# encryption algorithm <ALGORITHM>**	<ALGORITHM> – encryption protocol, takes the following values: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256. Default value: 3des.
11	Define Diffie-Hellman group number (optional).	**esr(config-ike-proposal)# dh-group <DH-GROUP>**	<DH-GROUP> – Diffie-Hellman group number, takes values of [1, 2, 5, 14, 15, 16, 17, 18]. Default value: 1.
12	Specify IKE authentication mode (optional)	**esr(config-ike-proposal)# authentication method <METHOD>**	<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. Default value: pre-shared-key.
13	Create an IKE policy and switch to its configuration mode.	**esr(config)# security ike policy <NAME>**	<NAME> – IKE policy name, set by the string of up to 31 characters.
14	Specify the lifetime of IKE protocol connection (optional).	**esr(config-ike-proposal)# lifetime seconds <SEC>**	<SEC> – time interval, takes values of [4..86400] seconds. Default value: 3600.
15	Bind IKE profile to IKE policy.	**esr(config-ike-policy)# proposal <NAME>**	<NAME> – IKE protocol name, set by the string of up to 31 characters.
16	Specify authentication key (mandatory if pre-shared-key is selected as authentication mode)	**esr(config-ike-policy)# pre-shared-key ascii-text<TEXT>**	<TEXT> – string [1..64] ASCII characters.
17	Create an IKE gateway and switch to its configuration mode.	**esr(config)# security ike gateway <NAME>**	<NAME> – IKE protocol gateway name, set by the string of up to 31 characters.

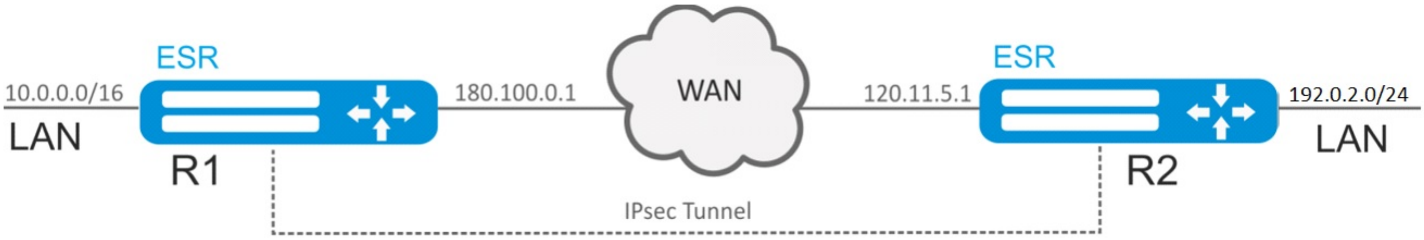
Step	Description	Command	Keys
18	Bind IKE policy to IKE gateway.	**esr(config-ike-gw)# ike-policy <NAME>**	<NAME> – IKE protocol policy name, set by the string of up to 31 characters.
19	Specify IKE version (optional).	**esr(config-ike-gw)# version <VERSION>**	<version> – IKE protocol version: v1-only or v2-only. Default value: v1-only.
20	Set the route-based mode.	**esr(config-ike-gw)# mode route-based**	
21	Specify the action for DPD (optional).	**esr(config-ike-gw)# dead-peer-detection action <MODE>**	<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.
22	Specify the interval between sending messages via DPD mechanism (optional).	**esr(config-ike-gw)# dead-peer-detection interval <SEC>**	<SEC> – interval between sending messages via DPD mechanism, takes values of [1..180] seconds. Default value: 2.
23	Specify the time period of response to DPD mechanism messages (optional).	**esr(config-ike-gw)# dead-peer-detection timeout <SEC>**	<SEC> – time interval of response to DPD mechanism messages, takes values of [1..180] seconds. Default value: 30 seconds.
24	Bind VTI tunnel to IKE gateway.	**esr(config-ike-gw)# bind-interface vti <VTI>**	<VTI> – VTI ID.
25	Create IPsec profile.	**esr(config)# security ipsec proposal <NAME>**	<NAME> – IPsec protocol profile name, set by the string of up to 31 characters.

Step	Description	Command	Keys
26	Specify IPsec authentication algorithm (optional).	**esr(config-ipsec-proposal)# authentication algorithm <ALGORITHM>**	<ALGORITHM> – authentication algorithm, takes values of: md5, sha1, sha2-256, sha2_384, sha2-512. Default value: sha1.
27	Specify IPsec encryption algorithm (route).	**esr(config-ipsec-proposal)# encryption algorithm <ALGORITHM>**	<ALGORITHM> – encryption protocol, takes the following values: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256. Default value: 3des.
28	Specify encapsulation protocol for IPsec (optional).	**esr(config-ipsec-proposal)# protocol <PROTOCOL>**	<PROTOCOL> – encapsulation protocol, takes the following values: Default value: esp.
29	Create an IPsec policy and switch to its configuration mode.	**esr(config)# security ipsec policy <NAME>**	<NAME> – IPsec policy name, set by the string of up to 31 characters.
30	Bind IPsec profile to IPsec policy.	**esr(config-ipsec-policy)# proposal <NAME>**	<NAME> – IPsec protocol profile name, set by the string of up to 31 characters.
31	Specify the lifetime of IPsec tunnel (optional).	**esr(config-ipsec- policy)# lifetime { seconds <SEC> | packets <PACKETS> | kilobytes <KB> }**	<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: 28800 seconds.

Step	Description	Command	Keys
32	Create IPsec VPN policy and switch to its configuration mode.	**esr(config)# security ipsec vpn <NAME>**	<NAME> – VPN name, set by the string of up to 31 characters.
33	Define the matching mode of data required for VPN enabling.	**esr(config-ipsec-vpn)# mode <MODE>**	<MODE> – VPN operation mode.
34	Bind IPsec policy to IPsec VPN.	**esr(config-ipsec-vpn)# ike ipsec-policy <NAME>**	<NAME> – IPsec policy name, set by the string of up to 31 characters.
35	Set the DSCP value for the use in IP headers of IKE outgoing packets (optional).	**esr(config-ipsec-vpn)# ike dscp <DSCP>**	<DSCP> – DSCP code value, takes values in the range of [0..63]. Default value: 63.
36	Set VPN activation mode.	**esr(config-ipsec-vpn)# ike establish-tunnel <MODE>**	<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.
37	Bind IKE gateway to IPsec VPN.	**esr(config-ipsec-vpn)# ike gateway <NAME>**	<NAME> – IKE gateway name, set by the string of up to 31 characters.
38	Set the time interval value in seconds after which the connection is closed, if no packet has been received or sent via SA (optional).	**esr(config-ipsec-vpn)# ike idle-time <TIME>**	<TIME> – interval in seconds, takes values of [4..86400].
39	Disable key re-approval before the IKE connection is lost due to the timeout, the number of transmitted packets or bytes (optional).	**esr(config-ipsec-vpn)# ike rekey disable**	

Step	Description	Command	Keys
40	Configure the start of IKE connection keys re-approval before the expiration of the lifetime (optional).	**esr(config-ipsec-vpn)# ike rekey margin { seconds <SEC> | packets <PACKETS> | kilobytes <KB> }**	<SEC> – time interval in seconds remaining before the connection release (set by the lifetimeseconds command, see 22.2.13). Takes values in the range of [4..86400]. <PACKETS> – number of packets remaining before the connection release (set by the lifetimepackets command). Takes values in the range of [4..86400]. <KB> – traffic volume in kilobytes remaining before the connection release (set by the lifetimekilobytes command). 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.
41	Set the level of margin seconds, margin packets, margin kilobytes values random spread (optional).	**esr(config-ipsec-vpn)# ike rekey randomization <VALUE>**	<VALUE> – maximum ratio of values spread, takes values of [1..100]. Default value: 100%
42	Specify the description for IPsec-VPN (optional).	**esr(config-ipsec-vpn)# description <DESCRIPTION>**	<DESCRIPTION> – profile description, set by the string of up to 255 characters.
43	Enable IPsec VPN.	**esr(config-ipsec-vpn)# enable**	

9.4.2 Route-based IPsec VPN configuration example



Objective:

Configure IPsec tunnel between R1 and R2.

- R1 IP address: 120.11.5.1;
- R2 IP address: 180.100.0.1.

IKE:

- Diffie-Hellman group: 2;
- encryption algorithm: AES 128 bit;
- authentication algorithm: MD5.

IP sec:

- encryption algorithm: AES 128 bit;
- authentication algorithm: MD5.

Solution:

1. R1 configuration:
   Configure external network interface and identify its inherence to a security zone:

```
esr# configure
esr(config)# interface gi 1/0/1
esr(config-if-gi)# ip address 180.100.0.1/24
esr(config-if-gi)# security-zone untrusted
esr(config-if-gi)# exit
```

   Create VTI tunnel. Traffic will be routed via VTI into IPsec tunnel. Specify IP addresses of WAN border interfaces as local and remote gateways:

```
esr(config)# tunnel vti 1
esr(config-vti)# local address 180.100.0.1
esr(config-vti)# remote address 120.11.5.1
esr(config-vti)# enable
esr(config-vti)# exit
```

   To configure security zones rules, create ISAKMP port profile:

```
esr(config)# object-group service ISAKMP
esr(config-object-group-service)# port-range 500
esr(config-object-group-service)# exit
```

Create a static route to the remote LAN. For each subnet located beyond the IPsec tunnel, specify a route via VTI tunnel:

```
esr(config)# ip route 192.0.2.0/24 tunnel vti 1
```

Create IKE protocol profile. Select Diffie-Hellman group 2, AES 128 bit encryption algorithm and MD5 authentication algorithm in the profile. The given security parameters are used for IKE connection protection:

```
esr(config)# security ike proposal ike_prop1
esr(config-ike-proposal)# dh-group 2
esr(config-ike-proposal)# authentication algorithm md5
esr(config-ike-proposal)# encryption algorithm aes128
esr(config-ike-proposal)# exit
```

Create IKE protocol policy. For the policy, specify the list of IKE protocol profiles that may be used for node and authentication key negotiation:

```
esr(config)# security ike policy ike_pol1
esr(config-ike-policy)# pre-shared-key hexadecimal 123FFF
esr(config-ike-policy)# proposal ike_prop1
esr(config-ike-policy)# exit
```

Create IKE protocol gateway. For this profile, specify VTI tunnel, policy, protocol version and mode of traffic redirection into the tunnel.

```
esr(config)# security ike gateway ike_gw1
esr(config-ike-gw)# ike-policy ike_pol1
esr(config-ike-gw)# mode route-based
esr(config-ike-gw)# bind-interface vti 1
esr(config-ike-gw)# version v2-only
esr(config-ike-gw)# exit
```

Create security parameters profile for IPsec tunnel. For the profile, select Diffie-Hellman group 2, AES 128 bit encryption algorithm and MD5 authentication algorithm. Use the following parameters to secure IPsec tunnel:

```
esr(config)# security ipsec proposal ipsec_prop1
esr(config-ipsec-proposal)# authentication algorithm md5
esr(config-ipsec-proposal)# encryption algorithm aes128
esr(config-ipsec-proposal)# exit
```

Create a policy for IPsec tunnel. For the policy, specify the list of IPsec tunnel profiles that may be used for node negotiation:

```
esr(config)# security ipsec policy ipsec_pol1
esr(config-ipsec-policy)# proposal ipsec_prop1
esr(config-ipsec-policy)# exit
```

Create IPsec VPN. For VPN, specify IKE protocol gateway, IPsec tunnel policy, key exchange mode and connection establishment method. When all parameters are entered, enable tunnel using the *enable* command.

```
esr(config)# security ipsec vpn ipsec1
esr(config-ipsec-vpn)# mode ike
esr(config-ipsec-vpn)# ike establish-tunnel route
esr(config-ipsec-vpn)# ike gateway ike_gw1
esr(config-ipsec-vpn)# ike ipsec-policy ipsec_pol1
esr(config-ipsec-vpn)# enable
esr(config-ipsec-vpn)# exit
esr(config)# exit
```

2. R2 configuration
Configure external network interface and identify its inherence to a security zone:

```
esr# configure
esr(config)# interface gi 1/0/1
esr(config-if)# ip address 120.11.5.1/24
esr(config-if)# security-zone untrusted
esr(config-if)# exit
```

Create VTI tunnel. Traffic will be routed via VTI into IPsec tunnel. Specify IP addresses of WAN border interfaces as local and remote gateways:

```
esr(config)# tunnel vti 1
esr(config-vti)# remote address 180.100.0.1
esr(config-vti)# local address 120.11.5.1
esr(config-vti)# enable
esr(config-vti)# exit
```

To configure security zones rules, create ISAKMP port profile:

```
esr(config)# object-group service ISAKMP
esr(config-object-group-service)# port-range 500
esr(config-object-group-service)# exit
```

Create a static route to the remote LAN. For each subnet located beyond the IPsec tunnel, specify a route via VTI tunnel:

```
esr(config)# ip route 10.0.0.0/16 tunnel vti 1
```

Create IKE protocol profile. Select Diffie-Hellman group 2, AES 128 bit encryption algorithm and MD5 authentication algorithm in the profile. The given security parameters are used for IKE connection protection:

```
esr(config)# security ike proposal ike_prop1
esr(config-ike-proposal)# dh-group 2
esr(config-ike-proposal)# authentication algorithm md5
esr(config-ike-proposal)# encryption algorithm aes128
esr(config-ike-proposal)# exit
esr(config)#
```

Create IKE protocol policy. For the policy, specify the list of IKE protocol profiles that may be used for node and authentication key negotiation:

```
esr(config)# security ike policy ike_pol1
esr(config-ike-policy)# pre-shared-key hexadecimal 123FFF
esr(config-ike-policy)# proposal ike_prop1
esr(config-ike-policy)# exit
```

Create IKE protocol gateway. For this profile, specify VTI tunnel, policy, protocol version and mode of traffic redirection into the tunnel.

```
esr(config)# security ike gateway ike_gw1
esr(config-ike-gw)# ike-policy ike_pol1
esr(config-ike-gw)# mode route-based
esr(config-ike-gw)# bind-interface vti 1
esr(config-ike-gw)# version v2-only
esr(config-ike-gw)# exit
```

Create security parameters profile for IPsec tunnel. For the profile, select Diffie-Hellman group 2, AES 128 bit encryption algorithm and MD5 authentication algorithm. Use the following parameters to secure IPsec tunnel:

```
esr(config)# security ipsec proposal ipsec_prop1
esr(config-ipsec-proposal)# authentication algorithm md5
esr(config-ipsec-proposal)# encryption algorithm aes128
esr(config-ipsec-proposal)# exit
```

Create a policy for IPsec tunnel. For the policy, specify the list of IPsec tunnel profiles that may be used for node negotiation:

```
esr(config)# security ipsec policy ipsec_pol1
esr(config-ipsec-policy)# proposal ipsec_prop1
esr(config-ipsec-policy)# exit
```

Create IPsec VPN. For VPN, specify IKE protocol gateway, IPsec tunnel policy, key exchange mode and connection establishment method. When all parameters are entered, enable tunnel using the *enable* command.

```
esr(config)# security ipsec vpn ipsec1
esr(config-ipsec-vpn)# mode ike
esr(config-ipsec-vpn)# ike establish-tunnel route
esr(config-ipsec-vpn)# ike gateway ike_gw1
esr(config-ipsec-vpn)# ike ipsec-policy ipsec_pol1
esr(config-ipsec-vpn)# enable
esr(config-ipsec-vpn)# exit
esr(config)# exit
```

To view the tunnel status, use the following command:

```
esr# show security ipsec vpn status ipsec1
```

To view the tunnel configuration, use the following command:

```
esr# show security ipsec vpn configuration ipsec1
```

Enable ESP and ISAKMP protocol (UDP port 500) in the firewall.

9.4.3 Policy-based IPsec VPN configuration algorithm

Step	Description	Command	Keys
1	Create an IKE instance and switch to its configuration mode.	**esr(config)# security ike proposal <NAME>**	<NAME> – IKE protocol name, set by the string of up to 31 characters.
2	Specify the description of the configured tunnel (optional).	**esr(config-ike-proposal)# description<DESCRIPTION>**	<DESCRIPTION> – tunnel description, set by the string of up to 255 characters.
3	Specify IKE authentication algorithm.	**esr(config-ike-proposal)# authentication algorithm <ALGORITHM>**	<ALGORITHM> – authentication algorithm, takes values of: md5, sha1, sha2-256, sha2_384, sha2-512.
4	Specify IKE encryption algorithm.	**esr(config-ike-proposal)# encryption algorithm <ALGORITHM>**	<ALGORITHM> – encryption protocol, takes the following values: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.
5	Define Diffie-Hellman group number.	**esr(config-ike-proposal)# dh-group <DH-GROUP>**	<DH-GROUP> – Diffie-Hellman group number, takes values of [1, 2, 5, 14, 15, 16, 17, 18].
6	Specify the authentication mode.	**esr(config-ike-proposal)# authentication method <METHOD>**	<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.
7	Create an IKE profile policy and switch to its configuration mode.	**esr(config)# security ike policy <NAME>**	<NAME> – IKE policy name, set by the string of up to 31 characters.

Step	Description	Command	Keys
8	Specify the lifetime of IKE protocol connection (optional).	**esr(config-ike-proposal)# lifetime seconds <SEC>**	<SEC> – time interval, takes values of [4..86400] seconds.
9	Bind the policy to profile.	**esr(config-ike-policy)# proposal <NAME>**	<NAME> – IKE protocol name, set by the string of up to 31 characters.
10	Specify authentication key.	**esr(config-ike-policy)#pre-shared-key ascii-text<TEXT>**	<TEXT> – string [1..64] ASCII characters.
11	Create an IKE gateway and switch to its configuration mode.	**esr(config)# security ike gateway <NAME>**	<NAME> – IKE protocol gateway name, set by the string of up to 31 characters.
12	Bind IKE policy.	**esr(config-ike-gw)# ike-policy <NAME>**	<NAME> – IKE protocol policy name, set by the string of up to 31 characters.
13	Specify IKE version (optional).	**esr(config-ike-gw)# version <VERSION>**	<version> – IKE protocol version: **v1-only** or **v2-only**.
14	Set the mode of traffic redirection into the tunnel.	**esr(config-ike-gw)#mode<MODE>**	<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.
15	Specify the action for DPD (optional).	**esr(config-ike-gw)# dead-peer-detection action <MODE>**	<MODE> – DPD operation mode: • **restart** – connection restarts; • **clear** – conection stops; • **hold** – connection holds; • **none** – the mechanism is disabled, no action is taken.
16	Specify the interval between sending messages via DPD mechanism (optional).	**esr(config-ike-gw)#dead-peer-detection interval <SEC>**	<SEC> – interval between sending messages via DPD mechanism, takes values of [1..180] seconds.

Step	Description	Command	Keys
17	Specify the time period of response to DPD mechanism messages (optional).	**esr(config-ike-gw)# dead-peer-detection timeout <SEC>**	<SEC> – time interval of response to DPD mechanism messages, takes values of [1..180] seconds.
18	Specify IKE version (optional).	**esr(config-ike-gw)# version <VERSION>**	<version> – IKE protocol version: **v1-only** or **v2-only**.
19	Set sender's IP subnets.	**esr(config-ike-gw)# local network <ADDR/LEN> [protocol { <TYPE> | <ID> } [port <PORT>]]**	<ADDR/LEN> – subnet IP address and mask 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].
20	Specify the IP address of IPsec tunnel local gateway.	**esr(config-ike-gw)#local address <ADDR>**	<ADDR> – IP address of a local gateway.
21	Specify the IP address of IPsec tunnel remote gateway.	**esr(config-ike-gw)#remote address <ADDR>**	<ADDR> – IP address of a remote gateway.
22	Set recipient's subnet IP address as well as IP and port.	**esr(config-ike-gw)# remote network <ADDR/LEN> [protocol { <TYPE> | <ID> } [port <PORT>]]**	<ADDR/LEN> – subnet IP address and mask 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].

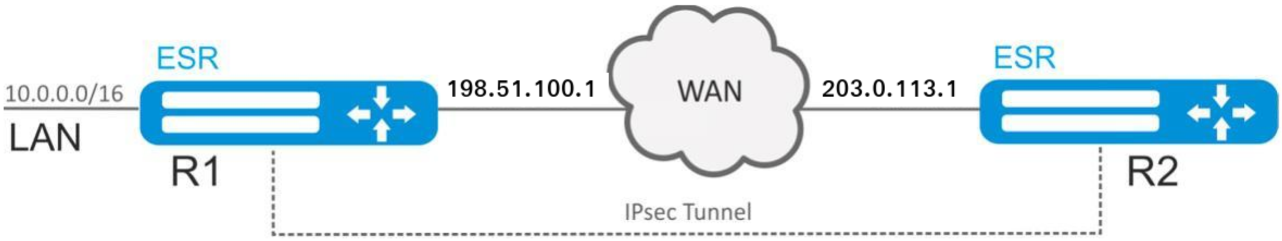
Step	Description	Command	Keys
23	Create IPsec profile.	**esr(config)# security ipsec proposal <NAME>**	<NAME> – IPsec protocol profile name, set by the string of up to 31 characters.
24	Specify IPsec authentication algorithm.	**esr(config-ipsec-proposal)# authentication algorithm <ALGORITHM>**	<ALGORITHM> – authentication algorithm, takes values of: md5, sha1, sha2-256, sha2_384, sha2-512.
26	Specify IPsec encryption algorithm.	**esr(config-ipsec-proposal)# encryption algorithm <ALGORITHM>**	<ALGORITHM> – encryption protocol, takes the following values: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.
26	Specify protocol (optional).	**esr(config-ipsec-proposal)#protocol <PROTOCOL>**	<PROTOCOL> – encapsulation protocol, takes the following values:
27	Create an IPsec profile policy and switch to its configuration mode.	**esr(config)# security ipsec policy <NAME>**	<NAME> – IPsec policy name, set by the string of up to 31 characters.
28	Bind the policy to profile.	**esr(config-ipsec-policy)# proposal <NAME>**	<NAME> – IPsec protocol profile name, set by the string of up to 31 characters.
29	Specify the lifetime of IPsec tunnel (optional).	**esr(config-ipsec-policy)# lifetime { seconds <SEC> | packets <PACKETS> | kilobytes <KB> }**	<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.

Step	Description	Command	Keys
30	Create IPsec VPN policy and switch to its configuration mode.	**esr(config)# security ipsecvpn <NAME>**	<NAME> – VPN name, set by the string of up to 31 characters.
31	Define the matching mode of data required for VPN enabling.	**esr(config-ipsec-vpn)# mode <MODE>**	<MODE> – VPN operation mode.
32	Bind IPsec policy to VPN.	**esr(config-ipsec-vpn)#ike ipsec-policy <NAME>**	<NAME> – IPsec policy name, set by the string of up to 31 characters.
33	Set the DSCP value for the use in IP headers of IKE outgoing packets (optional).	**esr(config-ipsec-vpn)#ike dscp <DSCP>**	<DSCP> – DSCP code value, takes values in the range of [0..63].
34	Set VPN activation mode.	**esr(config-ipsec-vpn)#ike establish-tunnel <MODE>**	<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.
35	Bind IKE gateway to VPN.	**esr(config-ipsec-vpn)# ike gateway <NAME>**	<NAME> – IKE gateway name, set by the string of up to 31 characters.
36	Set the time interval value in seconds after which the connection is closed, if no packet has been received or sent via SA (optional).	**esr(config-ipsec-vpn)# ike idle-time <TIME>**	<TIME> – interval in seconds, takes values of [4..86400].
37	Disable key re-approval before the IKE connection is lost due to the timeout, the number of transmitted packets or bytes (optional).	**esr(config-ipsec-vpn)# ike rekey disable**	

Step	Description	Command	Keys
38	Configure the start of IKE connection keys re-approval before the expiration of the lifetime (optional).	**esr(config-ipsec-vpn)# ike rekey margin { seconds <SEC> | packets <PACKETS> | kilobytes <KB> }**	<SEC> – time interval in seconds remaining before the connection release (set by the lifetimeseconds command). Takes values in the range of [4..86400]. <PACKETS> – number of packets remaining before the connection release (set by the lifetimepackets command). Takes values in the range of [4..86400]. <KB> – traffic volume in kilobytes remaining before the connection release (set by the lifetimekilobytes command). Takes values in the range of [4..86400].
39	Set the level of margin seconds, margin packets, margin kilobytes values random spread (optional).	**esr(config-ipsec-vpn)# ike rekey randomization <VALUE>**	<VALUE> – maximum ratio of values spread, takes values of [1..100].
40	Describe VPN (optional).	**esr(config-ipsec-vpn)# description <DESCRIPTION>**	<DESCRIPTION> – profile description, set by the string of up to 255 characters.
41	Enable IPsec VPN.	**esr(config-ipsec-vpn)# enable**	

9.4.4 Policy-based IPsec VPN configuration example

Objective:



Configure IPsec tunnel between R1 and R2.

R1 IP address – 198.51.100.1;

R2 IP address – 203.0.113.1.

IKE:

- Diffie-Hellman group: 2;
- encryption algorithm: AES 128 bit;
- authentication algorithm: MD5.

IPSEC:

- encryption algorithm: AES 128 bit;
- authentication algorithm: MD5.

Solution:

1. R1 configuration
   Configure external network interface and identify its inherence to a security zone:

```
esr# configure
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# ip address 198.51.100.1/24
esr(config-if-gi)# security-zone untrusted
esr(config-if-gi)# exit
```

   To configure security zones rules, create ISAKMP port profile:

```
esr(config)# object-group service ISAKMP
esr(config-object-group-service)# port-range 500
esr(config-object-group-service)# exit
```

   Create IKE protocol profile. Select Diffie-Hellman group 2, AES 128 bit encryption algorithm and MD5 authentication algorithm in the profile. The given security parameters are used for IKE connection protection:

```
esr(config)# security ike proposal ike_prop1
esr(config-ike-proposal)# dh-group 2
esr(config-ike-proposal)# authentication algorithm md5
esr(config-ike-proposal)# encryption algorithm aes128
esr(config-ike-proposal)# exit
```

   Create IKE protocol policy. For the policy, specify the list of IKE protocol profiles that may be used for node and authentication key negotiation:

```
esr(config)# security ike policy ike_pol1
esr(config-ike-policy)# pre-shared-key hexadecimal 123FFF
esr(config-ike-policy)# proposal ike_prop1
esr(config-ike-policy)# exit
```

Create IKE protocol gateway. For this profile, specify VTI tunnel, policy, protocol version and mode of traffic redirection into the tunnel.

```
esr(config)# security ike gateway ike_gw1
esr(config-ike-gw)# ike-policy ike_pol1
esr(config-ike-gw)# local address 198.51.100.1
esr(config-ike-gw)# local network 10.0.0.0/16
esr(config-ike-gw)# remote address 203.0.113.1
esr(config-ike-gw)# remote network 192.0.2.0/24
esr(config-ike-gw)# mode policy-based
esr(config-ike-gw)# exit
```

Create security parameters profile for IPsec tunnel. For the profile, select Diffie-Hellman group 2, AES 128 bit encryption algorithm and MD5 authentication algorithm. Use the following parameters to secure IPsec tunnel:

```
esr(config)# security ipsec proposal ipsec_prop1
esr(config-ipsec-proposal)# authentication algorithm md5
esr(config-ipsec-proposal)# encryption algorithm aes128
esr(config-ipsec-proposal)# exit
```

Create a policy for IPsec tunnel. For the policy, specify the list of IPsec tunnel profiles that may be used for node negotiation:

```
esr(config)# security ipsec policy ipsec_pol1
esr(config-ipsec-policy)# proposal ipsec_prop1
esr(config-ipsec-policy)# exit
```

Create IPsec VPN. For VPN, specify IKE protocol gateway, IPsec tunnel policy, key exchange mode and connection establishment method. When all parameters are entered, enable tunnel using the *enable* command.

```
esr(config)# security ipsec vpn ipsec1
esr(config-ipsec-vpn)# mode ike
esr(config-ipsec-vpn)# ike establish-tunnel route
esr(config-ipsec-vpn)# ike gateway ike_gw1
esr(config-ipsec-vpn)# ike ipsec-policy ipsec_pol1
esr(config-ipsec-vpn)# enable
esr(config-ipsec-vpn)# exit
esr(config)# exit
```

2. R2 configuration
   Configure external network interface and identify its inherence to a security zone:

```
esr# configure
esr(config)# interface gi 1/0/1
esr(config-if)# ip address 203.0.113.1/24
esr(config-if)# security-zone untrusted
esr(config-if)# exit
```

To configure security zones rules, create ISAKMP port profile:

```
esr(config)# object-group service ISAKMP
esr(config-addr-set)# port-range 500
esr(config-addr-set)# exit
```

Create IKE protocol profile. Select Diffie-Hellman group 2, AES 128 bit encryption algorithm and MD5 authentication algorithm in the profile. The given security parameters are used for IKE connection protection:

```
esr(config)# security ike proposal ike_prop1
esr(config-ike-proposal)# dh-group 2
esr(config-ike-proposal)# authentication algorithm md5
esr(config-ike-proposal)# encryption algorithm aes128
esr(config-ike-proposal)# exit
esr(config)#
```

Create IKE protocol policy. For the policy, specify the list of IKE protocol profiles that may be used for node and authentication key negotiation:

```
esr(config)# security ike policy ike_pol1
esr(config-ike-policy)# pre-shared-key hexadecimal 123FFF
esr(config-ike-policy)# proposal ike_prop1
esr(config-ike-policy)# exit
```

Create IKE protocol gateway. For this profile, specify VTI tunnel, policy, protocol version and mode of traffic redirection into the tunnel:

```
esr(config)# security ike gateway ike_gw1
esr(config-ike-gw)# ike-policy ike_pol1
esr(config-ike-gw)# remote address 198.51.100.1
esr(config-ike-gw)# remote network 10.0.0.0/16
esr(config-ike-gw)# local address 203.0.113.1
esr(config-ike-gw)# local network 192.0.2.0/24
esr(config-ike-gw)# mode policy-based
esr(config-ike-gw)# exit
```

Create security parameters profile for IPsec tunnel. For the profile, select Diffie-Hellman group 2, AES 128 bit encryption algorithm and MD5 authentication algorithm. Use the following parameters to secure IPsec tunnel:

```
esr(config)# security ipsec proposal ipsec_prop1
esr(config-ipsec-proposal)# authentication algorithm md5
esr(config-ipsec-proposal)# encryption algorithm aes128
esr(config-ipsec-proposal)# exit
```

Create a policy for IPsec tunnel. For the policy, specify the list of IPsec tunnel profiles that may be used for node negotiation:

```
esr(config)# security ipsec policy ipsec_pol1
esr(config-ipsec-policy)# proposal ipsec_prop1
esr(config-ipsec-policy)# exit
```

Create IPsec VPN. For VPN, specify IKE protocol gateway, IPsec tunnel policy, key exchange mode and connection establishment method. When all parameters are entered, enable tunnel using the *enable* command:

```
esr(config)# security ipsec vpn ipsec1
esr(config-ipsec-vpn)# mode ike
esr(config-ipsec-vpn)# ike establish-tunnel route
esr(config-ipsec-vpn)# ike gateway ike_gw1
esr(config-ipsec-vpn)# ike ipsec-policy ipsec_pol1
esr(config-ipsec-vpn)# enable
esr(config-ipsec-vpn)# exit
esr(config)# exit
```

To view the tunnel status, use the following command:

```
esr# show security ipsec vpn status ipsec1
```

To view the tunnel configuration, use the following command:

```
esr# show security ipsec vpn configuration ipsec1
```

Enable ESP and ISAKMP protocol (UDP port 500) in the firewall.

9.4.5 Remote Access IPsec VPN configuration algorithm

Remote Access IPsec VPN – scenario for organizing temporary VPN connections in which the IPsec VPN server is waiting for incoming connections, and clients make temporary connections to the server to gain access to network resources.

An additional feature of RA IPsec VPN is the ability to use the second IPsec authentication factor – Extended Authentication (XAUTH), where the second authentication factor is the login-password pair for the IPsec VPN client.

Step	Description	Command	Keys
1	Create an IKE instance and switch to its configuration mode.	**esr(config)# security ike proposal <NAME>**	<NAME> – IKE protocol name, set by the string of up to 31 characters.
2	Specify the description of the configured tunnel (optional).	**esr(config-ike-proposal)# description <DESCRIPTION>**	<DESCRIPTION> – tunnel description, set by the string of up to 255 characters.
3	Specify IKE authentication algorithm (optional).	**esr(config-ike-proposal)# authentication algorithm <ALGORITHM>**	<ALGORITHM> – authentication algorithm, takes values of: md5, sha1, sha2-256, sha2_384, sha2-512. Default value: sha1

Step	Description	Command	Keys
4	Specify the IP address of the VTI tunnel local side (optional).	**esr(config-vti)# ip address <ADDR/LEN>**	<ADDR/LEN> – IP address and prefix of a subnet, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..31].
5	Define Diffie-Hellman group number (optional).	**esr(config-ike-proposal)# dh-group <DH-GROUP>**	<DH-GROUP> – Diffie-Hellman group number, takes values of [1, 2, 5, 14, 15, 16, 17, 18]. Default value: 1
6	Create an IKE profile policy and switch to its configuration mode.	**esr(config)# security ike policy <NAME>**	<NAME> – IKE policy name, set by the string of up to 31 characters.
7	Specify the authentication mode.	**esr(config-ike- policy)# authentication method <METHOD>**	<METHOD> – key authentication method. May take the following values: • **xauth-psk-key** – two-factor authentication method using a login-password pair and previously obtained encryption keys.
8	Set the client mode (only for client).	**esr(config-ike- policy)# authentication mode client**	
9	Specify the lifetime of IKE protocol connection (optional).	**esr(config-ike- policy)# lifetime seconds <SEC>**	<SEC> – time interval, takes values of [4..86400] seconds. Default value: 3600
10	Bind the policy to profile.	**esr(config-ike-policy)# proposal <NAME>**	<NAME> – IKE protocol name, set by the string of up to 31 characters.
11	Specify authentication key.	**esr(config-ike-policy)#pre-shared-key ascii-text <TEXT>**	<TEXT> – string [1..64] ASCII characters.
12	Create an access profile.	**esr(config)# access profile <NAME>**	<NAME> – access profile name, set by the string of up to 31 characters.
13	Create user name.	**esr(config-access-profile)# user <LOGIN>**	<LOGIN> – login for client, set by the string of up to 31 characters.

Step	Description	Command	Keys
14	Specify a password for a user.	**esr(config-profile)# password ascii-text <TEXT>**	<TEXT> – string [8..32] ASCII characters.
15	Create a destination address pool (only for server).	**esr(config)# address-assignment pool <NAME>**	<NAME> – destination addresses pool name, set by the string of up to 31 characters.
16	Set the subnet from which IP clients will be issued (only for server).	**esr(config-pool)# ip prefix <ADDR/LEN>**	<ADDR/LEN> – address and prefix of the subnet.
17	Create an IKE gateway and switch to its configuration mode.	**esr(config)# security ike gateway <NAME>**	<NAME> – IKE protocol gateway name, set by the string of up to 31 characters.
18	Bind IKE policy.	**esr(config-ike-gw)# ike-policy <NAME>**	<NAME> – IKE protocol policy name, set by the string of up to 31 characters.
19	Set the mode of traffic redirection into the tunnel.	**esr(config-ike-gw)# mode <MODE>**	<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.
20	Specify the action for DPD (optional).	**esr(config-ike-gw)# dead-peer-detection action <MODE>**	<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
21	Specify the interval between sending messages via DPD mechanism (optional).	**esr(config-ike-gw)#dead-peer-detection interval <SEC>**	<SEC> – interval between sending messages via DPD mechanism, takes values of [1..180] seconds. Default value: 2

Step	Description	Command	Keys
22	Specify the time period of response to DPD mechanism messages (optional).	**esr(config-ike-gw)# dead-peer-detection timeout <SEC>**	<SEC> – time interval of response to DPD mechanism messages, takes values of [1..180] seconds. Default value: 30
23	Specify IKE version (optional).	**esr(config-ike-gw)# version <VERSION>**	<version> – IKE protocol version: **v1-only** or **v2-only**. Default value: v1-only
24	Set the IP subnet of the source (only for server).	**esr(config-ike-gw)# local network <ADDR/LEN> [protocol { <TYPE> | <ID> } [port <PORT>]]**	<ADDR/LEN> – subnet IP address and mask 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].
25	Specify the IP address of IPsec tunnel local gateway.	**esr(config-ike-gw)#local address <ADDR>**	<ADDR> – IP address of a local gateway.
26	Specify the IP address of IPsec tunnel remote gateway.	**esr(config-ike-gw)#remote address [any | <ADDR/LEN> [protocol { <TYPE> | <ID> } [port <PORT>]]**	Any – set as a remote address – any client address in the server configuration; <ADDR/LEN> – IP address and subnet mask of the server, in client configuration.
27	Set the pool for dynamic allocation of IP addresses to clients (only for server).	**esr(config-ike-gw)# remote network dynamic pool <NAME>**	<NAME> – destination addresses pool name, set by the string of up to 31 characters.
28	Set the dynamic establishment mode of the remote subnet (only for client).	**esr(config-ike-gw)# remote network dynamic client**	

Step	Description	Command	Keys
29	Set access profile for XAUTH parameters (only for server).	**esr(config-ike-gw)# xauth access-profile <NAME>**	<NAME> – access profile name, set by the string of up to 31 characters.
30	Set access profile and login for XAUTH parameters (only for client).	**esr(config-ike-gw)# xauth access-profile <NAME> client <LOGIN>**	<NAME> – access profile name, set by the string of up to 31 characters; <LOGIN> – login for client, set by the string of up to 31 characters.
31	Define a dedicated IP termination interface for building IPsec VPN (only for client).	**esr(config-ike-gw)# assign-interface loopback <INDEX>**	<INDEX> – interface index, takes values of [1..65535].
32	Create IPsec profile.	**esr(config)# security ipsec proposal <NAME>**	<NAME> – IPsec protocol profile name, set by the string of up to 31 characters.
33	Specify IPsec authentication algorithm (optional).	**esr(config-ipsec-proposal)# authentication algorithm <ALGORITHM>**	<ALGORITHM> – authentication algorithm, takes values of: md5, sha1, sha2-256, sha2_384, sha2-512. Default value: sha1
34	Specify IPsec encryption algorithm (optional).	**esr(config-ipsec-proposal)# encryption algorithm <ALGORITHM>**	<ALGORITHM> – encryption protocol, takes the following values: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256. Default value: 3des
35	Specify protocol (optional).	**esr(config-ipsec-proposal)#protocol <PROTOCOL>**	<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

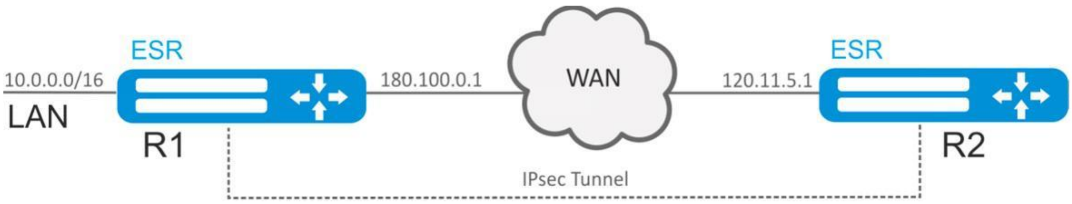
Step	Description	Command	Keys
36	Configuration config-ipsec-proposal	**esr(config)# security ipsec policy <NAME>**	<NAME> – IPsec policy name, set by the string of up to 31 characters.
37	Bind the policy to profile.	**esr(config-ipsec-policy)# proposal <NAME>**	<NAME> – IPsec protocol profile name, set by the string of up to 31 characters.
38	Specify the lifetime of IPsec tunnel (optional).	**esr(config-ipsec-policy)# lifetime { seconds <SEC> | packets <PACKETS> | kilobytes <KB> }**	<SEC> – IPsec tunnel lifetime after which the re-approval is carried out. Takes values in the range of [1140..86400] seconds. Default value: 540 <PACKETS> – number of packets after transmitting of which the IPsec tunnel re-approval is carried out. Takes values in the range of [4..86400]. Default value: disabled. <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: disabled.
39	Create IPsec VPN policy and switch to its configuration mode.	**esr(config)# security ipsec vpn <NAME>**	<NAME> – VPN name, set by the string of up to 31 characters.
40	Define the matching mode of data required for VPN enabling.	**esr(config-ipsec-vpn)# mode <MODE>**	<MODE> – VPN operation mode, takes the following values: ike, manual.
41	Bind IPsec policy to VPN.	**esr(config-ipsec-vpn)#ike ipsec-policy <NAME>**	<NAME> – IPsec policy name, set by the string of up to 31 characters.
42	Set the DSCP value for the use in IP headers of IKE outgoing packets (optional).	**esr(config-ipsec-vpn)#ike dscp <DSCP>**	<DSCP> – DSCP code value, takes values in the range of [0..63]. Default value: 63

Step	Description	Command	Keys
43	Set VPN activation mode.	**esr(config-ipsec-vpn)#ike establish-tunnel <MODE>**	<MODE> – VPN activation mode: • **by-request** – connection is activated by the opposite side, available for the server; • **route** – the connection is activated when traffic routed to the tunnel appears; it is available for the server; • **immediate** – tunnel is enabled automatically after applying the configuration, it is available for the client;
44	Bind IKE gateway to VPN.	**esr(config-ipsec-vpn)# ike gateway <NAME>**	<NAME> – IKE gateway name, set by the string of up to 31 characters.
45	Set the time interval value in seconds after which the connection is closed, if no packet has been received or sent via SA (optional).	**esr(config-ipsec-vpn)# ike idle-time <TIME>**	<TIME> – interval in seconds, takes values of [4..86400]. Default value: 0
46	Disable key re-approval before the IKE connection is lost due to the timeout, the number of transmitted packets or bytes (optional).	**esr(config-ipsec-vpn)# ike rekey disable**	Default value: disabled.
47	Configure the start of IKE connection keys re-approval before the expiration of the lifetime (optional).	**esr(config-ipsec-vpn)# ike rekey margin { seconds <SEC> | packets <PACKETS> | kilobytes <KB> }**	<SEC> – time interval in seconds remaining before the connection release (set by the lifetimeseconds command). Takes values in the range of [4..86400]. Default value: 540 <PACKETS> – number of packets remaining before the connection release (set by the lifetimepackets command). Takes values in the range of [4..86400]. Default value: disabled. <KB> – traffic volume in kilobytes remaining before the connection release (set by the lifetimekilobytes command). May take values [4..86400] Default value: disabled.

Step	Description	Command	Keys
48	Set the level of margin seconds, margin packets, margin kilobytes values random spread (optional).	**esr(config-ipsec-vpn)# ike rekey randomization <VALUE>**	<VALUE> – maximum ratio of values spread, takes values of [1..100]. Default value: 100
49	Describe VPN (route).	**esr(config-ipsec-vpn)# description <DESCRIPTION>**	<DESCRIPTION> – profile description, set by the string of up to 255 characters.
50	Enable IPsec VPN.	**esr(config-ipsec-vpn)# enable**	
51	Enable XAUTH clients reconnection mode with one login/password (server only) (optional).	**esr(config-ipsec-vpn)# security ike session uniqueids <MODE>**	<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.

9.4.6 Remote Access IPsec VPN configuration example

Objective:



Configure Remote Access IPsec VPN between R1 and R2 using the second IPsec authentication factor, XAUTH. Configure router R1 as the IPsec VPN server, and router R2 as the IPsec VPN client.

R2 IP address: 120.11.5.1;

R1 IP address: 180.100.0.1;

For IPsec VPN clients:

- issue addresses from the subnet pool 192.0.2.0/24
- provide access to the LAN subnet 10.0.0.0/16

IKE:

- Diffie-Hellman group: 2;
- encryption algorithm: 3DES;
- authentication algorithm: SHA1.

IPSEC:

- encryption algorithm: 3DES;
- authentication algorithm: SHA1.

XAUTH:

- login: client1;
- password: password123.

Solution:

1. R1 configuration
   Configure external network interface and identify its inherence to a security zone:

```
esr# configure
esr(config)# security zone untrusted
esr(config-zone)# exit
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# security-zone untrusted
esr(config-if-gi)# ip address 180.100.0.1/24
esr(config-if-gi)# exit
```

   To configure security zones rules, create ISAKMP port profile:

```
esr(config)# object-group service ISAKMP
esr(config-object-group-service)# port-range 500,4500
esr(config-object-group-service)# exit
```

Create IKE protocol profile. Select Diffie-Hellman group 2, 3DES encryption algorithm and SHA1 authentication algorithm in the profile. The given security parameters are used for IKE connection protection:

```
esr(config)# security ike proposal IKEPROP
esr(config-ike-proposal)# dh-group 2
esr(config-ike-proposal)# authentication algorithm sha1
esr(config-ike-proposal)# encryption algorithm 3des
esr(config-ike-proposal)# exit
```

Create IKE protocol policy. For the policy, specify the list of IKE protocol profiles that may be used for node, authentication key and XAUTH authentication method by key negotiation:

```
esr(config)# security ike policy IKEPOLICY
esr(config-ike-policy)# pre-shared-key hexadecimal 123FFF
esr(config-ike-policy)# authentication method xauth-psk-key
esr(config-ike-policy)# proposal IKEPROP
esr(config-ike-policy)# exit
```

Create an access profile and get in it a pair of username and password for the IPsec VPN client:

```
esr(config)# access profile XAUTH
esr(config-access-profile)# user client1
esr(config-profile)# password ascii-text password123
esr(config-profile)# exit
esr(config-access-profile)# exit
```

Create a pool of destination addresses from which IP clients will be issued IPsec VPN:

```
esr-1000(config)# address-assignment pool CLIENT_POOL
esr-1000(config-pool)# ip prefix 192.0.2.0/24
esr-1000(config-pool)# exit
```

Create IKE protocol gateway. In this profile, specify the IKE protocol policy, the local subnet, the destination address pool as the remote subnet, set the mode of traffic redirection to the tunnel according to the policy and use the second authentication factor XAUTH:

```
esr(config)# security ike gateway IKEGW
esr(config-ike-gw)# ike-policy IKEPOLICY
esr(config-ike-gw)# local address 180.100.0.1
esr(config-ike-gw)# local network 10.0.0.0/16
esr(config-ike-gw)# remote address any
esr(config-ike-gw)# remote network dynamic pool CLIENT_POOL
esr(config-ike-gw)# dead-peer-detection action clear
esr(config-ike-gw)# mode policy-based
esr(config-ike-gw)# xauth access-profile XAUTH
esr(config-ike-gw)# exit
```

Create security parameters profile for IPsec tunnel. Specify 3DES encryption algorithm and SHA1 authentication algorithm in the profile. Use the following parameters to secure IPsec tunnel:

```
esr(config)# security ipsec proposal IPSECPROP
esr(config-ipsec-proposal)# authentication algorithm sha1
esr(config-ipsec-proposal)# encryption algorithm 3des
esr(config-ipsec-proposal)# exit
```

Create a policy for IPsec tunnel. For the policy, specify the list of IPsec tunnel profiles that may be used for node negotiation:

```
esr(config)# security ipsec policy IPSECPOLICY
esr(config-ipsec-policy)# proposal IPSECPROP
esr(config-ipsec-policy)# exit
```

Create IPsec VPN. For VPN, specify IKE protocol gateway, IPsec tunnel policy, key exchange mode and waiting mode for the incoming IPsec connection – *by-request*. When all parameters are entered, enable tunnel using the *enable* command.

```
esr(config)# security ipsec IPSECVPN
esr(config-ipsec-vpn)# mode ike
esr(config-ipsec-vpn)# ike establish-tunnel by-request
esr(config-ipsec-vpn)# ike gateway IKEGW
esr(config-ipsec-vpn)# ike ipsec-policy IPSECPOLICY
esr(config-ipsec-vpn)# enable
esr(config-ipsec-vpn)# exit
```

Allow esp protocol and udp ports 500, 4500 in the firewall configuration for establishing IPsec VPN:

```
esr(config)# security zone-pair untrusted self
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol udp
esr(config-zone-pair-rule)# match destination-port ISAKMP
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)#   rule 2
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol esp
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# end
```

2. R2 configuration
Configure external network interface and identify its inherence to a security zone:

```
esr# configure
esr(config)# interface gi 1/0/1
esr(config-if)# ip address 120.11.5.1/24
esr(config-if)# security-zone untrusted
esr(config-if)# exit
```

To configure security zones rules, create ISAKMP port profile:

```
esr(config)# object-group service ISAKMP
esr(config-addr-set)# port-range 500,4500
esr(config-addr-set)# exit
```

Create IKE protocol profile. Select Diffie-Hellman group 2, 3DES encryption algorithm and SHA1 authentication algorithm in the profile. The given security parameters are used for IKE connection protection:

```
esr(config)# security ike proposal IKEPROP
esr(config-ike-proposal)# dh-group 2
esr(config-ike-proposal)# authentication algorithm sha1
esr(config-ike-proposal)# encryption algorithm 3des
esr(config-ike-proposal)# exit
```

Create IKE protocol policy. For the policy, specify the list of IKE protocol profiles that may be used for node, authentication key, XAUTH authentication method by key and client authentication mode negotiation:

```
esr(config)# security ike policy IKEPOLICY
esr(config-ike-policy)# pre-shared-key hexadecimal 123FFF
esr(config-ike-policy)# authentication method xauth-psk-key
esr(config-ike-policy)# authentication mode client
esr(config-ike-policy)# proposal IKEPROP
esr(config-ike-policy)# exit
```

Create an access profile and get in it a pair of username and password:

```
esr(config)# access profile XAUTH
esr(config-access-profile)# user client1
esr(config-profile)# password ascii-text password123
esr(config-profile)# exit
esr(config-access-profile)# exit
```

Create a loopback interface for terminating the IP address received from the IPsec VPN server:

```
esr(config)# interface loopback 8
esr(config-loopback)# exit
```

Create IKE protocol gateway. Specify the policy, the termination interface, the dynamic setting mode of the remote subnet, the access profile selection for XAUTH, and the mode of redirecting traffic to the tunnel by policy in this profile:

```
esr(config)# security ike gateway IKEGW
esr(config-ike-gw)# ike-policy IKEPOLICY
esr(config-ike-gw)# assign-interface loopback 8
esr(config-ike-gw)# local address 120.11.5.1
esr(config-ike-gw)# remote address 180.100.0.1
esr(config-ike-gw)# remote network dynamic client
esr(config-ike-gw)# mode policy-based
esr(config-ike-gw)# xauth access-profile xauth client client1
esr(config-ike-gw)# exit
```

Create security parameters profile for IPsec tunnel. Specify 3DES encryption algorithm and SHA1 authentication algorithm in the profile. Use the following parameters to secure IPsec tunnel:

```
esr(config)# security ipsec proposal IPSECPROP
esr(config-ipsec-proposal)# authentication algorithm md5
esr(config-ipsec-proposal)# encryption algorithm aes128
esr(config-ipsec-proposal)# exit
```

Create a policy for IPsec tunnel. For the policy, specify the list of IPsec tunnel profiles that may be used for node negotiation:

```
esr(config)# security ipsec policy IPSECPOLICY
esr(config-ipsec-policy)# proposal IPSECPROP
esr(config-ipsec-policy)# exit
```

Create IPsec VPN. For VPN, specify IKE protocol gateway, IPsec tunnel policy, key exchange mode and connection establishment method. When all parameters are entered, enable tunnel using *enable* command.

```
esr(config)# security ipsec vpn IPSECVPN
esr(config-ipsec-vpn)# mode ike
esr(config-ipsec-vpn)# ike establish-tunnel route
esr(config-ipsec-vpn)# ike gateway IKEGW
esr(config-ipsec-vpn)# ike ipsec-policy IPSECPOLICY
esr(config-ipsec-vpn)# enable
esr(config-ipsec-vpn)# exit
```

Allow esp protocol and udp ports 500,4500 in the firewall configuration for establishing IPsec VPN:

```
esr(config)# security zone-pair untrusted self
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol udp
esr(config-zone-pair-rule)# match destination-port ISAKMP
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)#   rule 2
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol esp
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# end
```

To view the tunnel status, use the following command:

```
esr# show security ipsec vpn status IPSECVPN
```

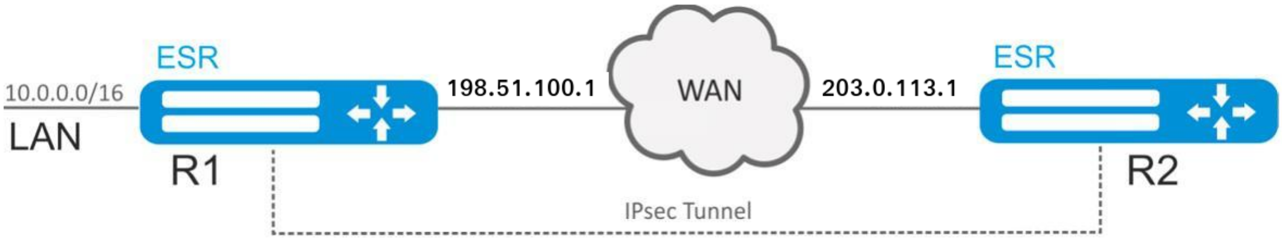
To view the tunnel configuration, use the following command:

```
esr# show security ipsec vpn configuration IPSECVPN
```

⚠ **Enable ESP and ISAKMP protocol (UDP port 500, 4500) in the firewall.**

9.4.7 DPD configuration example (Dead Peer Detection)

Objective:



Configure Dead Peed Detection on R1 for Policy-based Ipsec VPN between R1 and R2.

The initial configuration can be taken from the Policy-based IPsec VPN configuration example.

Solution:

On R1 in IKE protocol gateway specify the following: DPD operation mode – restart, polling interval – 1 second, timeout – 4 seconds:

```
esr# configure
esr(config)# security ike gateway ike_gw1
esr(config-ike-gw)# dead-peer-detection action restart
esr(config-ike-gw)# dead-peer-detection interval 1
esr(config-ike-gw)# dead-peer-detection timeout 4
esr(config-ike-gw)# exit
```

State of the tunnel can be seen with the command:

```
esr# show security ipsec vpn status ipsec1
```

Configuration of the tunnel can be seen with the command:

```
esr# show security ipsec vpn configuration ipsec1
```

After the connection between R1 and R2 on R1 is broken, the IPsec tunnel will start rebuilding 4 seconds after the break.

```
esr# show security ipsec vpn status
Name                              Local host        Remote host       Initiator spi
Responder spi         State
-------------------------------   ---------------   ---------------   ------------------
------------------   -----------
ipsec1                            198.51.100.1      203.0.113.1       0x7a77a25a55853255
0xb62fd04f2db43d08   Established
2037-10-30T07:52:53+00:00 %CLI-I-CMD: user admin from console  input: show security ipsec vpn
status
esr# show security ipsec vpn status
Name                              Local host        Remote host       Initiator spi
Responder spi         State
-------------------------------   ---------------   ---------------   ------------------
------------------   -----------
ipsec1                            198.51.100.1      203.0.113.1       0x77706e37b4e68cce
0x0000000000000000   Connecting
2037-10-30T07:52:57+00:00 %CLI-I-CMD: user admin from console  input: show security ipsec vpn
status
```

9.5 LT tunnels configuration

LT (Logical Tunnel) is a type of tunnels dedicated for transmission of routing information and traffic between different virtual routers (VRF) configured on a router. LT tunnel might be used for organization of interaction between two or more VRF using firewall restrictions.

9.5.1 Configuration algorithm

Step	Description	Command	Keys
1	Create LT tunnels for each of existing VRF.	**esr(config)# tunnel lt <ID>**	<ID> – tunnel identifier, set in the range of [1..128].
2	Specify the description of the configured tunnels (optional).	**esr(config-lt)# description <DESCRIPTION>**	<DESCRIPTION> – tunnel description, set by the string of up to 255 characters.
3	Include each LT tunnel in the corresponding VFR.	**esr(config-lt)# ip vrf forwarding <VRF>**	<VRF> – VRF name, set by the string of up to 31 characters.
4	Include each LT tunnel in a security zone and configure interaction rules between zones or disable firewall for LT tunnel.	**esr(config-lt)# security-zone<NAME>**	<NAME> – security zone name, set by the string of up to 12 characters.
		esr(config-lt)# ip firewall disable	
5	For each LT tunnel, set the opposite LT tunnel number (in another VRF).	**esr(config-lt)# peer lt <ID>**	<ID> – tunnel identifier, set in the range of [1..128].

Step	Description	Command	Keys
6	For each LT tunnel, specify IP address for packets routing. For interacting LT tunnels, IP addresses should locate in one IP subnet.	**esr(config-lt)# ip address <ADDR/ LEN>**	<ADDR/LEN> – IP address and prefix of a 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].
7	Enable the tunnels.	**esr(config-lt)# enable**	
8	For each VRF configure required routing protocols via LT tunnel.		
9	Specify the time interval during which the statistics on the tunnel load is averaged (optional).	**esr(config-lt)# load-average <TIME>**	<TIME> – interval in seconds, takes values of [5..150]. Default value: 5.
10	Specify the size of MTU packets that can be passed by the bridge (optional; possible if only VLAN is included in the bridge). MTU above 1500 will be active only when using the 'system jumbo-frames' command.	**esr(config-lt)# mtu <MTU>**	<MTU> – MTU value, takes values in the range of: • for ESR-10/12V(F)/ 14VF/15 – [1280..9600]; • for ESR-20/21/30 – [1280..9500]; • for ESR-100/200/1000/1200/1500/1511/1700/3100/3200 – [1280..10000]. Default value: 1500.

9.5.2 Configuration example

Objective:

Organize interaction between hosts terminated in two VRF vrf_1 and vrf_2.

Initial configuration:

```
hostname esr
ip vrf vrf_1
exit
ip vrf vrf_2
exit
interface gigabitethernet 1/0/1
  ip vrf forwarding vrf_1
  ip firewall disable
  ip address 10.0.0.1/24
exit
interface gigabitethernet 1/0/2
  ip vrf forwarding vrf_2
  ip firewall disable
  ip address 10.0.1.1/24
exit
```

Solution:

Create LT tunnels for each VRF, specifying IP address from one subnet:

```
esr(config)# tunnel lt 1
esr(config-lt)# ip vrf forwarding vrf_1
esr(config-lt)# ip firewall disable
esr(config-lt)# ip address 192.168.0.1/30
esr(config-lt)# exit
esr(config)# tunnel lt 2
esr(config-lt)# ip vrf forwarding vrf_2
esr(config-lt)# ip firewall disable
esr(config-lt)# ip address 192.168.0.2/30
esr(config-lt)# exit
```

Designate LT tunnel from VRF, which is necessary to establish link with, for each LT tunnel and activate them.

```
esr(config)# tunnel lt 1
esr(config-lt)# peer lt 2
esr(config-lt)# enable
esr(config-lt)# exit
esr(config)# tunnel lt 2
esr(config-lt)# peer lt 1
esr(config-lt)# enable
esr(config-lt)# exit
```

If none of dynamic routing protocols is configured in VRF, specify static routes for each VRF:

```
esr(config)# ip route vrf vrf_1 0.0.0.0/0 192.168.0.2
esr(config)# ip route vrf vrf_2 0.0.0.0/0 192.168.0.1
```

10 QoS management

QoS (Quality of Service) is a technology that provides various traffic classes with various service priorities. QoS service allows network applications to co-exist in a single network without altering the bandwidth of other applications.

10.1 Basic QoS

In basic mode on ESR routers, classification (routing traffic to the queue) and relabeling works only on the input (QoS must be enabled on the interface through which traffic arrives).

10.1.1 Configuration algorithm

Step	Description	Command	Keys
1	Enable QoS on the interface/tunnel/network bridge. If QoS policy is not assigned on the interface, the interface operates in BasicQoS mode.	**esr(config-if-gi)# qos enable**	
2	Set the trust mode for 802.1p and DSCP codes values in incoming packets (optional).	**esr(config)# qos trust <MODE>**	<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. • **cos** – trust mode for 802.1p codes values in 802.1q tag. Untagged packets will be sent to the default queue. • **cos-dscp** – trust mode for DSCP codes values in IP packets and for 802.1p codes values in other packets.

Step	Description	Command	Keys
3	Set the match between DSCP codes values of incoming packets and outgoing queues. The given match works for incoming interfaces/tunnels/bridge on which QoS is enabled (optional).	**esr(config)# qos map dscp-queue <DSCP> to <QUEUE>**	<DSCP> – service classifier in a packet IP header, takes values in the range of [0..63]; <QUEUE> – queue identifier, takes values in the range of [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
4	Set the match between 802.1p codes values of incoming packets and outgoing queues. The given match works for incoming interfaces/tunnels/bridge on which QoS is enabled (optional).	**esr(config)# qos map cos-queue <COS> to <QUEUE>**	<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
5	Set the match between DSCP codes values of incoming packets and outgoing DSCP codes (if remarking is required). The given match works for incoming interfaces/tunnels/bridge on which QoS is enabled.	**esr(config)# qos map dscp-queue <DSCP> to <DSCP>**	<DSCP> – service classifier in a packet IP header, takes values in the range of [0..63].
6	Enable DSCP codes changes according to the DSCP-Mutation table (if remarking is required).	**esr(config)# qos dscp mutation**	
7	Set the number of the default queue to which all traffic except IP falls into the trust mode for DSCP priorities.	**esr(config)# qos queue default <QUEUE>**	<QUEUE> – queue identifier, takes values in the range of [1..8].

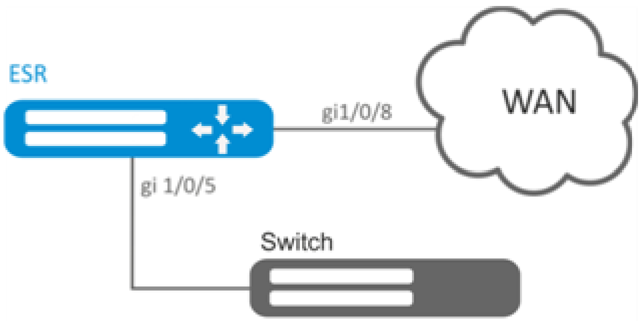
Step	Description	Command	Keys
8	Set the amount of priority queues. The remaining queues are weighted (optional).	**esr(config)# priority-queue out num-of-queues <VALUE>**	<VALUE> – amount of queues, takes values of [0..8], where: • **0** – all queues take part in WRR (WRR – weight-based queue processing mechanism); • **8** – all queues are served as «strictpriority» (strictpriority – priority queue is served as soon as the packets appear). The priority queues are allocated, starting from the 8th one, decreasing the queue number. Default value: 8.
9	Define the weights for corresponding weighted queues.	**esr(config)# qos wrr-queue <QUEUE> bandwidth <WEIGHT>**	<QUEUE> – queue identifier, takes values in the range of [1..8]; <WEIGHT> – weight value, takes values in the range of [1..255]. The default value: weight 1 for all queues.
10	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. If the incoming traffic was classified by advanced QoS, the limiting will not work (if the incoming rate limiting is required).	**esr(config-if-gi)# traffic-shape { <BANDWIDTH> [BURST] | queue <QUEUE><BANDWIDTH> [BURST] }**	<QUEUE> – queue identifier, takes values in the range of [1..8]. <BANDWIDTH> – average traffic rate in Kbps, takes the value of [3000..10000000] for TengigabitEthernet interfaces and [64..1000000] for other interfaces and tunnels; <BURST> – size of the restrictive threshold in KB, takes the value [4..16000]. 128 KB. Default value: Disabled.

Step	Description	Command	Keys
11	Set the incoming traffic rate limiting (if the outgoing rate limiting is required).	**esr(config-if-gi)# rate-limit <BANDWIDTH> [BURST]**	<BANDWIDTH> – average traffic rate in Kbps, takes the value of [3000..10000000] for TengigabitEthernet interfaces and [64..1000000] for other interfaces and tunnels; <BURST> – size of the restrictive threshold in KB, takes the value [4..16000]. 128 KB. Default value: Disabled.

10.1.2 Configuration example

Objective:

Configure the following restrictions on gigabitethernet 1/0/8 interface: transfer DSCP 22 traffic into 8th priority queue, DSCP 14 traffic into 7th weighted queue, limit transfer rate to 60Mbps for 7th queue.



Solution:

In order to make 8th queue a priority queue, and 2nd to 8th queues weighted ones, limit the quantity of priority queues to 1:

```
esr(config)# priority-queue out num-of-queues 1
```

Redirect DSCP 22 traffic into 1st priority queue:

```
esr(config)# qos map dscp-queue 22 to 8
```

Redirect DSCP 14 traffic into 7th priority queue:

```
esr(config)# qos map dscp-queue 14 to 7
```

Enable QoS on the incoming interface to correctly classify traffic and direct it to the appropriate queue on the LAN side:

```
esr(config)# interface gigabitethernet 1/0/5
esr(config-if-gi)# qos enable
esr(config-if-gi)# exit
```

Enable QoS on the WAN side interface for proper queue handling and bandwidth limitation:

```
esr(config)# interface gigabitethernet 1/0/8
esr(config-if-gi)# qos enable
```

Limit transfer rate to 60Mbps for 7th queue:

```
esr(config-if)# traffic-shape queue 7 60000
esr(config-if)# exit
```

To view QoS statistics, use the following command:

```
esr# show qos statistics gigabitethernet 1/0/8
```

10.2 Advanced QoS

10.2.1 Configuration algorithm

In advanced mode on ESR routers, classification of incoming traffic is possible on both incoming and outgoing interfaces.

Step	Description	Command	Keys
1	Create access lists to define the traffic to which the advanced QoS should be applied.		See section Access list (ACL) configuration.
2	Create QoS class and switch to the class parameters configuration mode.	**esr(config)# class-map <NAME>**	<NAME> – name of the class being created, set by the string of up to 31 characters.
3	Specify QoS class description (optional).	**esr(config-class-map)# description <description>**	<description> – up to 255 characters..
4	Specify the traffic related to the configured class by access control list (ACL).	**esr(config-class-map)# match access-group <NAME>**	<NAME> – access control list name, set by the string of up to 31 characters.

Step	Description	Command	Keys
5	Specify DSCP code value which will be set in IP packets corresponding to the class being configured (cannot be assigned simultaneously with IP Precedence and CoS fields) (if remarking is required).	**esr(config-class-map)# set dscp <DSCP>**	<DSCP> – DSCP code value, takes values in the range of [0..63].
6	Specify IP Precedence code value which will be set in IP packets corresponding to the class being configured (cannot be assigned simultaneously with DSCP and CoS fields) (if remarking is required).	**esr(config-class-map)# set ip-precedence <IPP>**	<IPP> – IP Precedence code value, takes values in the range of [0..7].
7	Specify 802.1p priority value which will be set in packets corresponding to the class being configured (cannot be assigned simultaneously with DSCP and IP Precedence fields) (if remarking is required).	**esr(config-class-map)# set cos <COS>**	<COS> – priority 802.1p value, takes values of [0..7].
8	Create QoS policy and switch to the policy parameters configuration mode.	**esr(config)# policy-map <NAME>** **esr(config-policy-map)#**	<NAME> – name of the policy being created, set by the string of up to 31 characters.
9	Specify QoS policy description (optional).	**esr(config-policy-map)# description <description>**	<description> – up to 255 characters..
10	Set the committed outgoing bandwidth for the policy in total.	**esr(config-policy-map)# shape average { <BANDWIDTH> | percent <BANDWIDTH_PERCENT> } [BURST]**	<BANDWIDTH> – committed bandwidth in Kbps, takes the value of [64..10000000]; <BANDWIDTH_PERCENT> – committed bandwidth in %, calculated from (in order from higher priority to lower priority value): • **shape average** value of the root policy; • **traffic-shape** value on the network interface, bridge, tunnel; • **speed** value of the network interface. Takes value of [1..100]. <BURST> – size of the restrictive threshold in KB, takes the value [128..16000]. 128 KB.

Step	Description	Command	Keys
11	Enable automatic bandwidth allocation between classes without bandwidth configuration, including the default class (if required).	**esr(config-policy-map)# shape auto-distribution**	
12	Include the specified QoS class in the policy and switch to the class parameters configuration mode within the policy.	**esr(config-policy-map)# class <NAME>** **esr(config-class-policy-map)#**	<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 the given class.
13	Include QoS policy in QoS class to create hierarchical QoS.	**esr(config-class-policy-map)# service-policy <NAME>**	<NAME> – policy name, set by the string of up to 31 characters. Inserted policy must already be created.
14	Set the committed outgoing bandwidth for the class within the policy (if required).	**esr(config-class-policy-map)# shape average { <BANDWIDTH> | percent <BANDWIDTH_PERCENT> } [BURST]**	<BANDWIDTH> – committed bandwidth in Kbps, takes the value of [64..10000000]; <BANDWIDTH_PERCENT> – committed bandwidth in %, calculated from (in order from higher priority to lower priority value): • **shape average** value of the root policy; • **traffic-shape** value on the network interface, bridge, tunnel; • **speed** value of the network interface. Takes value of [1..100]. <BURST> – size of the restrictive threshold in KB, takes the value [4..16000]. Default value: 128 KB.
15	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 (if required).	**esr(config-class-policy-map)# shape peak { <BANDWIDTH> | percent <BANDWIDTH_PERCENT> } [BURST]**	

Step	Description	Command	Keys
16	Specify class operation mode (optional).	**esr(config-class-policy-map)# mode <MODE>**	<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**.
17	Specify the class priority in WRR process (if required).	**esr(config-class-policy-map)# priority class <PRIORITY>**	<PRIORITY> – priority of class in WRR process, takes values of [1..8]. Classes with the highest priority are proceeded first.
18	Switch the class to the StrictPriority mode and specify the class priority (if required).	**esr(config-class-policy-map)# priority level <PRIORITY>**	<PRIORITY> – priority level in StrictPriority process, takes values of [1..8]. Classes with the highest priority are proceeded first. The default value: tha class operates in WRR mode, the priority is not specified.
19	Specify the limited number of virtual queues (optional).	**esr(config-class-policy-map)# fair-queue <QUEUE-LIMIT>**	<QUEUE-LIMIT> – limited number of virtual queues, takes values in the range of [16..4096]. Default value: 16.
20	Specify the limited number of packets for a virtual queue (optional).	**esr(config-class-policy-map)# queue-limit <QUEUE-LIMIT>**	<QUEUE-LIMIT> – limited number of packets in a virtual queue, takes values in the range of [2..4096]. Default value: 127.

Step	Description	Command	Keys
21	Specify RED (Random Early Detection) parameters (if required).	**esr(config-class-policy-map)# random-detect <LIMIT> <MIN> <MAX> <APS> <APS-NUM> <PROBABILITY>**	<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]; <APS> – average size of a queue in bytes, takes value in the range of [1..10000000]; <APS-NUM> – number of average size packets allowed for short-term transmission; <PROBABILITY> – probability of packet drop, takes values of [0..100]. When specifying the values, the following rules should be fulfilled: • <MAX>> 2 * <MIN> • <LIMIT>> 3 * <MAX>

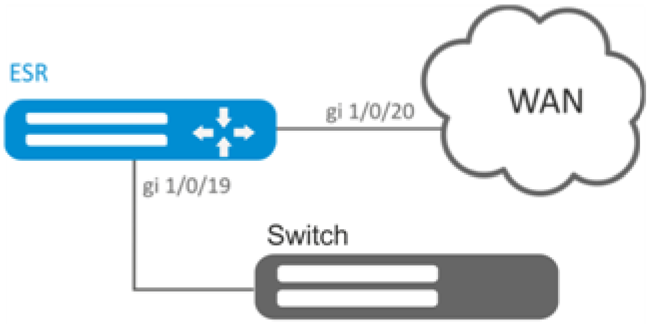
Step	Description	Command	Keys
22	Specify GRED (Generalized Random Early Detection) parameters (if required).	**esr(config-class-policy-map)# random-detect queue <QUEUE-NUM> [dscp <DSCP> | precedence <IPP>] <LIMIT> <MIN> <MAX> <APS> <APS-NUM> <PROBABILITY>**	<QUEUE-NUM> – queue number [1..16]; <DSCP> – DSCP code value, takes values in the range of [0..63]; <IPP> – IP Precedence code value, takes values in the range of [0..7]; <PRECEDENCE> – IP Precedence 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]; <APS> – average size of a queue in bytes, takes value in the range of [1..10000000]; <APS-NUM> – number of average size packets allowed for short-term transmission; <PROBABILITY> – probability of packet drop, takes values of [0..100]. When specifying the values, the following rules should be fulfilled: • <MAX>> 2 * <MIN> • <LIMIT>> 3 * <MAX>
23	Enable tcp headers compression protocol for the certain class traffic (if required).	**esr(config-class-policy-map)# compression header ip tcp**	
24	Enable QoS on the interface/tunnel/ network bridge.	**esr(config-if-gi)# qos enable**	

Step	Description	Command	Keys
25	Define the QoS policy on a configured interface/tunnel/network bridge to classify input and prioritize output traffic.	**esr(config-if-gi)# service-policy { input | output } <NAME>**	<NAME> – QoS policy name, set by the string of up to 31 characters.

10.2.2 Configuration example

Objective:

Classify incoming traffic by a subnet (10.0.11.0/24, 10.0.12.0/24), label it by DSCP (38 and 42) and segregate by a subnet (40Mbps and 60Mbps), limit general bandwidth to 250Mbps, process the rest of traffic using SFQ mechanism.



Solution:

Configure access control lists for filtering by a subnet, proceed to global configuration mode:

```
esr(config)# ip access-list extended fl1
esr(config-acl)# rule 1
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol any
esr(config-acl-rule)# match source-address 10.0.11.0 255.255.255.0
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# exit
esr(config)# ip access-list extended fl2
esr(config-acl)# rule 1
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol any
esr(config-acl-rule)# match source-address 10.0.12.0 255.255.255.0
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# exit
```

Create classes fl1 and fl2, specify the respective access control lists, configure labelling:

```
esr(config)# class-map fl1
esr(config-class-map)# set dscp 38
esr(config-class-map)# match access-group fl1
esr(config-class-map)# exit
esr(config)# class-map fl2
esr(config-class-map)# set dscp 42
esr(config-class-map)# match access-group fl2
esr(config-class-map)# exit
```

Create policy and define general bandwidth limits:

```
esr(config)# policy-map fl
esr(config-policy-map)# shape average 250000
```

Map class to policy, configure bandwidth limit and exit:

```
esr(config-policy-map)# class fl1
esr(config-class-policy-map)# shape average 40000
esr(config-class-policy-map)# exit
esr(config-policy-map)# class fl2
esr(config-class-policy-map)# shape average 60000
esr(config-class-policy-map)# exit
```

To set the bandwidth limit as a percentage, use the *shape average percent* command.

For the rest of traffic, configure a class with SFQ mode:

```
esr(config-policy-map)# class class-default
esr(config-class-policy-map)# mode sfq
esr(config-class-policy-map)# fair-queue 800
esr(config-class-policy-map)# exit
esr(config-policy-map)# exit
```

Enable QoS on the interfaces, policy on gi 1/0/19 interface ingress for classification purposes and gi1/0/20 egress for applying restrictions and SFQ mode for default class:

```
esr(config)# interface gigabitethernet 1/0/19
esr(config-if-gi)# qos enable
esr(config-if-gi)# service-policy input fl
esr(config-if-gi)# exit
esr(config)# interface gigabitethernet 1/0/20
esr(config-if-gi)# qos enable
esr(config-if-gi)# service-policy output fl
esr(config-if-gi)# exit
```

To view the statistics, use the following command:

```
esr# do show qos policy statistics gigabitethernet 1/0/20
```

11 Routing management

11.1 Routing information advertising policy

11.1.1 RIP

in/ out	Default policy	Advertising methods	Filtering methods	Filtering policy application levels
Import	Route information reception is not limited	Network, Redistribute	Route-map — the last (implicit) rule denies anything that is not explicitly allowed by the previous rules. Prefix-list — the last (implicit) rule denies anything that is not explicitly allowed by the previous rules.	RIP process
Export	Without separate advertising commands, the router does not send routing information		Prefix-list — the last (implicit) rule allows anything that is not explicitly denied by the previous rules. Prefix-list — the last (implicit) rule allows anything that is not explicitly denied by the previous rules.	

11.1.2 OSPF protocol

in/ out	Default policy	Advertising methods	Filtering methods	Filtering policy application levels
Import	Route information reception is not limited	Network, Redistribute	Route-map — the last (implicit) rule denies anything that is not explicitly allowed by the previous rules. Prefix-list — the last (implicit) rule denies anything that is not explicitly allowed by the previous rules.	OSPF process

in/ out	Default policy	Advertising methods	Filtering methods	Filtering policy application levels
Export	Information about interfaces with OSPF protocol enabled is advertised		Route-map — the last (implicit) rule allows anything that is not explicitly denied by the previous rules. Prefix-list — the last (implicit) rule allows anything that is not explicitly denied by the previous rules. *Filtering of advertised route information is possible for the following types of OSPF routes: E2, E1*	

11.1.3 IS-IS protocol

in/ out	Default policy	Advertising methods	Filtering methods	Filtering policy application levels
Import	Route information reception is not limited	Network, Redistribute	Route-map — the last (implicit) rule denies anything that is not explicitly allowed by the previous rules. Prefix-list — the last (implicit) rule denies anything that is not explicitly allowed by the previous rules.	IS-IS process
Export	Information about interfaces with IS-IS protocol enabled is advertised		Route-map — the last (implicit) rule allows anything that is not explicitly denied by the previous rules. Prefix-list — the last (implicit) rule allows anything that is not explicitly denied by the previous rules.	

11.1.4 iBPG protocol

in/ out	Default policy	Advertising methods	Filtering methods	Filtering policy application levels
Import	Route information reception is not limited	Network, Redistribute	Route-map — the last (implicit) rule denies anything that is not explicitly allowed by the previous rules. Prefix-list — the last (implicit) rule denies anything that is not explicitly allowed by the previous rules.	address-family, peer-group, neighbor
Export	All the routes that have entered the RIB via the BGP protocol are advertised		Route-map — the last (implicit) rule denies anything that is not explicitly allowed by the previous rules. Prefix-list — the last (implicit) rule denies anything that is not explicitly allowed by the previous rules.	

11.1.5 eBPG protocol

in/ out	Default policy	Advertising methods	Filtering methods	Filtering policy application levels
Import	Route information reception is not limited	Network, Redistribute	Route-map — the last (implicit) rule denies anything that is not explicitly allowed by the previous rules. Prefix-list — the last (implicit) rule denies anything that is not explicitly allowed by the previous rules.	address-family, peer-group, neighbor
Export	Routes are denied until an allowing route-map or prefix-list is applied		Route-map — the last (implicit) rule denies anything that is not explicitly allowed by the previous rules. Prefix-list — the last (implicit) rule denies anything that is not explicitly allowed by the previous rules.	

11.2 Static routes configuration

Static routing is a type of routing in which routes are defined explicitly during the router configuration without dynamic routing protocols.

11.2.1 Configuration algorithm

You can add a static route by using the following command in global configuration mode:

```
esr(config)# ip route [ vrf <VRF> ] <SUBNET> { <NEXTHOP> | interface <IF> | tunnel <TUN> | wan load-balance rule <RULE> [<METRIC>] | blackhole | unreachable | prohibit } [ <METRIC> ] [ track <TRACK-ID> ] [ bfd ]
```

- <VRF> – VRF name, set by the string of up to 31 characters.
- <SUBNET> – destination address, can be specified in the following format:
  - 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];
- <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);
- bfd – when specifying the given key, the removal of static route in case of next-hop unavailability is activated.

To add static IPv6 route to the given subnet, use the following command:

```
ipv6 route [ vrf <VRF> ] <SUBNET> { <NEXTHOP> [ resolve ] | interface <IF> | wan load-balance rule <RULE> | blackhole | unreachable | prohibit } [ <METRIC> ] [ bfd ]
```

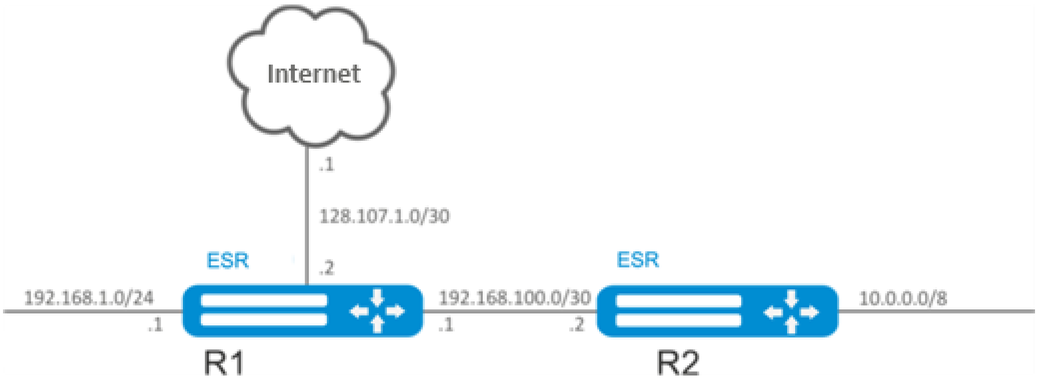
- <VRF> – VRF name, set by the string of up to 31 characters.
- <SUBNET> – destination address, can be specified in the following formats:
  - The addresses are defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].
  - <IPV6-ADDR/LEN> – IP address and mask of a subnet, defined 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 the given 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].
- bfd – when specifying the given key, the removal of static route in case of next-hop unavailability is activated.

11.2.2 Static routes configuration example

Objective:

Configure Internet access for users in LAN 192.168.1.0/24 and 10.0.0.0/8 using the static routing. On R1 device, create gateway for Internet access. Traffic within LAN should be routed within LAN zone, traffic from the Internet should belong to WAN zone.



Solution:

Specify the device name for R1 router:

```
esr# hostname R1
```

Specify 192.168.1.1/24 address and the 'LAN' zone for the gi1/0/1 interface. R1 interface will be connected to 192.168.1.0/24 network via this interface:

```
esr(config)# interface gi1/0/1
esr(config-if-gi)# security-zone LAN
esr(config-if-gi)# ip address 192.168.1.1/24
esr(config-if-gi)# exit
```

Specify 192.168.100.1/30 address and the 'LAN' zone for the gi1/0/2 interface. R1 will be connected to R2 device via the given interface for the further traffic routing:

```
esr(config)# interface gi1/0/2
esr(config-if-gi)# security-zone LAN
esr(config-if-gi)# ip address 192.168.100.1/30
esr(config-if-gi)# exit
```

Specify 128.107.1.2/30 address and the 'WAN' zone for the gi1/0/3 interface. R1 interface will be connected to the Internet via this interface:

```
esr(config)# interface gi1/0/3
esr(config-if-gi)# security-zone WAN
esr(config-if-gi)# ip address 128.107.1.2/30
esr(config-if-gi)# exit
```

Create a route for interaction with 10.0.0.0/8 network using R2 device as a gateway (192.168.100.2):

```
esr(config)# ip route 10.0.0.0/8 192.168.100.2
```

Create a route for interaction with the Internet using the provider gateway as a nexthop (128.107.1.1):

```
esr(config)# ip route 0.0.0.0/0 128.107.1.1
```

Specify the device name for R2 router:

```
esr# hostname R2
```

Specify 10.0.0.1/8 address and the 'LAN' zone for the gi1/0/1 interface. R2 interface will be connected to 10.0.0.0/8 network via this interface:

```
esr(config)# interface gi1/0/1
esr(config-if-gi)# security-zone LAN
esr(config-if-gi)# ip address 10.0.0.1/8
esr(config-if-gi)# exit
```

Specify 192.168.100.2/30 address and the 'LAN' zone for the gi1/0/2 interface. R2 will be connected to R1 device via the given interface for the further traffic routing:

```
esr(config)# interface gi1/0/2
esr(config-if-gi)# security-zone LAN
esr(config-if-gi)# ip address 192.168.100.2/30
esr(config-if-gi)# exit
```

Create a default route by specifying the IP address of R1 router gi1/0/2 interface (192.168.100.1) as a nexthop:

```
esr(config)# ip route 0.0.0.0/0 192.168.100.1
```

To check the routing table the following command can be used:

```
esr# show ip route
```

11.3 RIP configuration

RIP is a distance-vector dynamic routing protocol that uses hop count as a routing metric. The maximum amount of hops allowed for RIP is 15. By default, each RIP router transmits full routing table into the network every 30 seconds. RIP operates at 3rd level of TCP/IP stack via UDP port 520.

11.3.1 Configuration algorithm

Step	Description	Command	Keys
1	Configure RIP precedence for the main routing table (optional).	**esr(config)# ip protocols rip preference <VALUE>**	<VALUE> – protocol precedence, takes values in the range of [1..255]. Default value: RIP (100).
2	Configure RIP routing tables capacity (optional).	**esr(config)# ip protocols rip max-routes <VALUE>**	<VALUE> – amount of RIP routes in the routing table, takes values in the range of[1..10000]; Default value: 10000.
3	Create IP subnets lists that will be used for further filtration of advertised and received IP routes.	**esr(config)# ip prefix-list <NAME>**	<NAME> – name of a subnet list being configured, set by the string of up to 31 characters.

Step	Description	Command	Keys
4	Permit or deny the prefixes lists.	**esr(config-pl)# permit {object-group <OBJ-GROUP-NETWORK-NAME > <ADDR/LEN> | <IPV6-ADDR/LEN> } [{ eq <LEN> | le <LEN> | ge <LEN> [le <LEN>] }]** **esr(config-pl)# deny {object-group <OBJ-GROUP-NETWORK-NAME> <ADDR/LEN> | <IPV6-ADDR/LEN> } [{ eq <LEN> | le <LEN> | ge <LEN> [le <LEN>] }**	<OBJ-GROUP-NETWORK-NAME> – IP addresses profile name, set by the string of up to 31 characters; <LEN> – prefix length, takes values of [1..32] in prefix IP 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; • **default-route** – default route filtration.
5	Switch to the RIP process configuration mode.	**esr(config)# router rip** **esr(config-rip)#**	
6	Enable RIP.	**esr(config-rip)# enable**	
7	Specify RIP authentication algorithm (optional).	**esr(config-rip)# authentication algorithm { cleartext | md5 }**	• **cleartext** – password, transmitted in clear text; • **md5** – password is hashed by md5 algorithm.
8	Set the password for neighbor authentication (optional).	**esr(config-rip)# authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }**	<CLEAR-TEXT> – password, set by the string of 8 to 16 characters; <ENCRYPTED-TEXT> – encrypted password of 8 to 16 bytes (from 16 to 32 characters) in hexadecimal format (0xYYYY ...) or (YYYY ...).
9	Specify the list of passwords for authentication via md5 hashing algorithm (optional).	**esr(config-rip)# authentication key-chain <KEYCHAIN>**	<KEYCHAIN> – key list identifier, set by the string of up to 16 characters.

Step	Description	Command	Keys
10	Disable routes advertising on the interfaces/tunnels/bridge where it is not necessary (optional).	**esr(config-rip)# passive-interface {<IF> | <TUN> }**	<IF> – interface and identifier; <TUN> – tunnel name and number.
11	Set time interval after which the advertising is carried out (optional).	**esr(config-rip)# timers update <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 180 seconds.
12	Set time interval of route entry correctness without updating (optional).	**esr(config-rip)# timers invalid <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 180 seconds.
13	Set time interval after which the route removing is carried out (optional).	**esr(config-rip)# timers flush <TIME>**	<TIME> – time in seconds, takes values of [12..65535]. When setting the value, consider the following rule: 'timersinvalid + 60'. Default value: 240 seconds.
14	Enable subnets advertising.	**esr(config-rip)# network <ADDR/LEN>**	<ADDR/LEN> – subnet 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].
15	Add subnets filtration in incoming or outgoing updates (optional).	**esr(config-rip)# prefix-list <PREFIX-LIST-NAME> { in | out }**	<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.
16	Enable advertising of routes received in an alternative way (optional).	**esr(config-rip)# redistribute static [route-map <NAME>]**	<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.

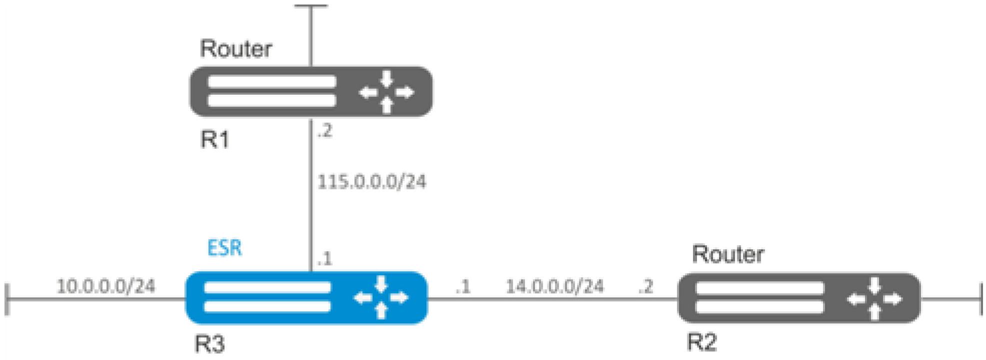
Step	Description	Command	Keys
		esr(config-rip)# redistribute connected [route-map <NAME>]	<NAME> – name of the route map that will be used for filtration and modification of advertised directly connected subnets, set by the string of up to 31 characters.
		esr(config-rip)# redistribute ospf <ID><ROUTE-TYPE> [route-map <NAME>]	<ID> – process number, takes values of [1..65535]. <ROUTE-TYPE> – route type: • **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.
		esr(config-rip)# redistribute bgp <AS> [route-map <NAME>]	<AS> – stand alone system number, takes values of [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.
17	Switch to the interface/tunnel/network bridge configuration mode.	**esr(config)# interface <IF-TYPE><IF-NUM>**	<IF-TYPE> – interface type; <IF-NUM> – F/S/P – F frame (1), S – slot (0), P – port.
		esr(config)# tunnel <TUN-TYPE><TUN-NUM>	<TUN-TYPE> – tunnel type; <TUN-NUM> – tunnel number.

Step	Description	Command	Keys
		esr(config)# bridge <BR-NUM>	<BR-NUM> – bridge number.
18	Set RIP routes metric value on the interface (optional).	**esr(config-if-gi)# ip rip metric <VALUE>**	<VALUE> – metric size, takes values of [0..32767]. Default value: 5.
19	Set the routes advertising mode via RIP (optional).	**esr(config-if-gi)# ip rip mode <MODE>**	<MODE> – routes advertising mode: • **multicast** – routes are advertised in multicast mode; • **broadcast** – routes are advertised in broadcast mode; • **unicast** – routes are advertised to the neighbors in unicast mode; Default value: multicast.
20	Specify a neighbor's IP address for establishment of a relation in routes advertising unicast mode (optional).	**esr(config-if-gi)# ip rip neighbor <ADDR>**	<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
21	Enable subnet summarization (optional).	**esr(config-if-gi)# ip rip summary-address <ADDR/LEN>**	<ADDR/LEN> – IP address and subnet mask, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32].

11.3.2 RIP configuration example

Objective:

Configure RIP on the router in order to exchange the routing information with neighboring routers. The router should advertise static routes and subnets 115.0.0.0/24, 14.0.0.0/24, 10.0.0.0/24. Routes should be advertised each 25 seconds.



Solution:

Pre-configure IP addresses on interfaces according to the network structure shown in figure above.

Switch to the RIP configuration mode:

```
esr(config)# router rip
```

Specify the networks to be advertised by protocol: 115.0.0.0/24, 14.0.0.0/24 и 10.0.0.0/24:

```
esr(config-rip)# network 115.0.0.0/24
esr(config-rip)# network 14.0.0.0/24
esr(config-rip)# network 10.0.0.0/24
```

To advertise static routes by the protocol, execute the following command:

```
esr(config-rip)# redistribute static
```

Configure timer, responsible for routing information transmission:

```
esr(config-rip)# timers update 25
```

When all required settings are done, enable the protocol:

```
esr(config-rip)# enable
```

To view the RIP routing table, use the following command:

```
esr# show ip rip
```

⚠ **In addition to RIP protocol configuration, open UDP port 520 in the firewall.**

11.4 OSFP configuration

OSPF is a dynamic routing protocol, based on link-state technology and using shortest path first Dijkstra algorithm.

11.4.1 Configuration algorithm

Step	Description	Command	Keys
1	Configure OSFP precedence for the main routing table (optional).	**esr(config)# ip protocols ospf preference <VALUE>**	<VALUE> – protocol precedence, takes values in the range of [1..255]. Default value: 150.
		esr(config-vrf)# ip protocols ospf preference <VALUE>	
2	Configure OSFP routing tables capacity (optional).	**esr(config)# ip protocols ospf max-routes <VALUE>**	<VALUE> – amount of OSPF routes in the routing table, takes values in the range of: • for ESR-1000/1200/1500/1511/1700/3100 – [1..500000]; • for ESR-20/21/100/200 – [1..300000]; • for ESR-10/12V(F)/14VF – [1..30000] Default value for the global mode: • for ESR-1000/1200/1500/1511/1700/3100 – (500000); • for ESR-20/21/100/200 – (300000); • for ESR-10/12V(F)/14VF – (30000). Default value for VRF: 0
		esr(config)# ipv6 protocols ospf max-routes <VALUE>	
3	Enable the output of OSPF neighbor state information (optional).	**esr(config)# router ospf log-adjacency-changes**	
		esr(config)# ipv6 router ospf log-adjacency-changes	
4	Create IP subnets lists that will be used for further filtration of advertised and received IP routes (optional).	**esr(config)# ip prefix-list <NAME>**	<NAME> – name of a subnet list being configured, set by the string of up to 31 characters.
		esr(config)# ipv6 prefix-list <NAME>	

Step	Description	Command	Keys
5	Permit or deny the prefixes lists (optional).	**esr(config-pl)# permit [{ object-group <OBJ-GROUP-NETWORK-NAME> | <ADDR/LEN> | <IPV6-ADDR/LEN> }] [{ eq <LEN> | le <LEN> | ge <LEN> [le <LEN>] }]** **esr(config-pl)# deny [{ object-group <OBJ-GROUP-NETWORK-NAME> | <ADDR/LEN > | <IPV6-ADDR/LEN> }] [{ eq <LEN> | le <LEN> | ge <LEN> [le <LEN>] }]**	<OBJ-GROUP-NETWORK-NAME> – IPv4/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]; <LEN> – prefix length, takes values of [1..32] in prefix IP 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;
6	Add OSFP process to the system and switch to the OSFP process parameters configuration mode.	**esr(config)# router ospf <ID> [vrf <VRF>]** **esr(config)# ipv6 router ospf <ID> [vrf <VRF>]**	<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.
7	Set router identifier for the given OSFP process.	**esr(config-ospf)# router-id <ID>** **esr(config-ipv6-ospf)# router-id <ID>**	<ID> – router identifier, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. <IF> – interface specified in form given in the Types and naming procedure of router interfaces section of CLI command reference guide. <TUN> – tunnel name specified in form given in the Types and naming procedure of router tunnels section.

Step	Description	Command	Keys
8	Define OSFP process routes precedence.	**esr(config-ospf)# preference <VALUE>**	<VALUE> – OSFP process routes precedence, takes values in the range of [1..255].
		esr(config-ipv6-ospf)# preference <VALUE>	
9	Define maximum number of equivalent routes (optional).	**esr(config-ospf)# maximum-path <PATHS>**	<PATHS> – number of equivalent routes to the target, takes values of [1..32]. Default value: 16.
		esr(config-ipv6-ospf)# maximum-path <PATHS>	
10	Enable compatibility with RFC 1583 (optional).	**esr(config-ospf)# compatible rfc1583**	
		esr(config-ipv6-ospf)# compatible rfc1583	
11	Add subnets filtration in incoming or outgoing updates (optional).	**esr(config-ospf)# prefix-list <PREFIX-LIST-NAME> { in | out }**	<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.
		esr(config-ipv6-ospf)# prefix-list <PREFIX-LIST-NAME> { in | out }	
12	Enable advertising of routes received in an alternative way (optional).	**esr(config-ospf)# redistribute static [route-map <NAME>]**	<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.
		esr(config-ipv6-ospf)# redistribute static [route-map <NAME>]	
		esr(config-ospf)# redistribute connected [route-map <NAME>]	<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.
		esr(config-ipv6-ospf)# redistribute connected [route-map <NAME>]	
		esr(config-ospf)# redistribute rip [route-map <NAME>]	<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.

Step	Description	Command	Keys
		esr(config-ospf)# redistribute bgp <AS> [route-map <NAME>]	<AS> – stand alone system number, takes values of [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.
		esr(config-ipv6-ospf)# redistribute bgp <AS> [route-map <NAME>]	
13	Enable OSFP process.	**esr(config-ospf)# enable**	
		esr(config-ipv6-ospf)# enable	
14	Create OSFP area and switch to the scope configuration mode.	**esr(config-ospf)# area <AREA_ID>**	<AREA_ID> – area identifier, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
		esr(config-ipv6-ospf)# area <AREA_ID>	
15	Enable subnets advertising (optional).	**esr(config-ospf-area)# network <ADDR/LEN>**	<ADDR/LEN> – subnet 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].
		esr(config-ipv6-ospf-area)# network <IPV6-ADDR/LEN>	<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].

Step	Description	Command	Keys
16	Specify the area type (optional).	**esr(config-ospf-area)# area-type <TYPE> [no-summary]** **esr(config-ipv6-ospf-area)# area-type <TYPE> [no-summary]**	<TYPE> – area type: • **stub** – sets stub value (stub area); no-summary – command in conjunction with the 'stub' parameter forms the 'totallystubby' area (only the default route is used to transfer information outside the area). • **nssa** – sets nssa value (NSSA area); no-summary – command in conjunction with the 'nssa' parameter forms the 'totallynssa' area (by default the route is generated as an inter-place one).
17	Enable the default route generation for NSSA area and its advertising as NSSA-LSA (optional).	**esr(config-ospf-area)# default-information-originate** **esr(config-ipv6-ospf-area)# default-information-originate**	
18	Enable the subnet summarization or hiding (optional).	**esr(config-ospf-area)# summary-address <ADDR/LEN> { advertise | not-advertise }**	<ADDR/LEN> – IP address and subnet mask, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32]; • **advertise** – if a command is specified, instead of the specified subnets, the total subnet will be advertised; • **not-advertise** – when specifying the command, the subnets included in a subnet specified will not be advertised.

Step	Description	Command	Keys
		esr(config-ipv6-ospf-area)# summary-address <IPV6-ADDR/LEN> { advertise | not-advertise }	<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** – when specifying the command instead of the subnets included in a subnet specified, a total subnet will be advertised; • **not-advertise** – the subnets included in a subnet specified will not be advertised.
19	Enable OSFP area.	**esr(config-ospf-area)# enable** **esr(config-ipv6-ospf-area)# enable**	
20	Establish a virtual connection between the main and remote areas having several areas between them (optional).	**esr(config-ospf-area)# virtual-link <ID>** **esr(config-ipv6-ospf-area)# virtual-link <ID>**	<ID> – identifier of the router with which the virtual connection is established, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
21	Set the time interval in seconds after which the router re-sends a packet that has not received a delivery confirmation (for example, a DatabaseDescription packet or LinkStateRequest packets) (optional).	**esr(config-ospf- vlink)# restransmit-interval <TIME>** **esr(config-ipv6-ospf- vlink)# restransmit-interval <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 5 seconds.
22	Set the time interval in seconds after which the router sends the next hello packet (optional).	**esr(config-ospf- vlink)# hello-interval <TIME>** **esr(config-ipv6-ospf- vlink)# hello-interval <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 10 seconds.
23	Set the time interval in seconds after which the neighbor is considered to be idle. This interval should be a multiple of the 'hello interval' value.	**esr(config-ospf- vlink)# dead-interval <TIME>** **esr(config-ipv6-ospf- vlink)# dead-interval <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 40 seconds.

Step	Description	Command	Keys
24	Set the time interval in seconds after which the router selects DR in the network (optional).	**esr(config-ospf- vlink)# wait-interval <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 40 seconds
		esr(config-ipv6-ospf- vlink)# wait-interval <TIME>	
25	Define authentication algorithm (optional).	**esr(config-ospf- vlink)# authentication algorithm <ALGORITHM>**	<ALGORITHM> – authentication algorithm: • **cleartext** – password, transmitted in unencrypted form (available only for RIP and OSPF-VLINK); • **md5** – password is hashed by md5 algorithm.
26	Set the password for neighbor authentication (optional).	**esr(config-ospf- vlink)# authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }**	<CLEAR-TEXT> – password, set by the string of 8 to 16 characters. <ENCRYPTED-TEXT> – encrypted password of 8 to 16 bytes (from 16 to 32 characters) in hexadecimal format (0xYYYY ...) or (YYYY ...).
27	Specify the list of passwords for authentication via md5 hashing algorithm.	**esr(config-ospf- vlink)# authentication key chain <KEYCHAIN>**	<KEYCHAIN> – key list identifier, set by the string of up to 16 characters.
28	Enable virtual connection.	**esr(config-ospf- vlink)# enable**	
29	Switch to the interface/tunnel/network bridge configuration mode.	**esr(config)# interface <IF-TYPE><IF-NUM>**	<IF-TYPE> – interface type; <IF-NUM> – F/S/P – F frame (1), S – slot (0), P – port.
		esr(config)# tunnel <TUN-TYPE><TUN-NUM>	<TUN-TYPE> – tunnel type; <TUN-NUM> – tunnel number.
		esr(config)# bridge <BR-NUM>	<BR-NUM> – bridge number.
30	Define the interface / tunnel / network bridge inherence to a specific OSPF process.	**esr(config-if-gi)# ip ospf instance <ID>**	<ID> – process number, takes values of [1..65535].

Step	Description	Command	Keys
		esr(config-if-gi)# ipv6 ospf instance <ID>	
31	Define the interface inherence to a specific OSPF process area.	**esr(config-if-gi)# ip ospf area <AREA_ID>** **esr(config-if-gi)# ipv6 ospf area <AREA_ID>**	<AREA_ID> – area identifier, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
32	Enable the routing via OSFP on the interface.	**esr(config-if-gi)# ip ospf** **esr(config-if-gi)# ipv6 ospf**	
33	Enable the mode in which the OSPF process will ignore MTU interface value in incoming Database Description packets (optional).	**esr(config-if-gi)# ip ospf mtu-ignore** **esr(config-if-gi)# ipv6 ospf mtu-ignore**	
34	Specify OSFP authentication algorithm (optional).	**esr(config-if-gi)# ip ospf authentication algorithm <ALGORITHM>**	<ALGORITHM> – authentication algorithm: • **cleartext** – password, transmitted in clear text; • **md5** – password is hashed by md5 algorithm.
35	Set the password for OSPF neighbor authentication when transmitting an unencrypted password (optional).	**esr(config-if-gi)# ip ospf authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }**	<CLEAR-TEXT> – password, set by the string of 8 to 16 characters; <ENCRYPTED-TEXT> – encrypted password of 8 to 16 bytes (from 16 to 32 characters) in hexadecimal format (0xYYYY ...) or (YYYY ...).
36	Specify the list of passwords for neighbor authentication via md5 hashing algorithm (optional).	**esr(config-if-gi)# ip ospf authentication key-chain <KEYCHAIN>**	<KEYCHAIN> – key list identifier, set by the string of up to 16 characters.
37	Set the time interval in seconds after which the router selects DR in the network (optional).	**esr(config-if-gi)# ip ospf wait-interval <TIME>** **esr(config-if-gi)# ipv6 ospf wait-interval <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 40 seconds.

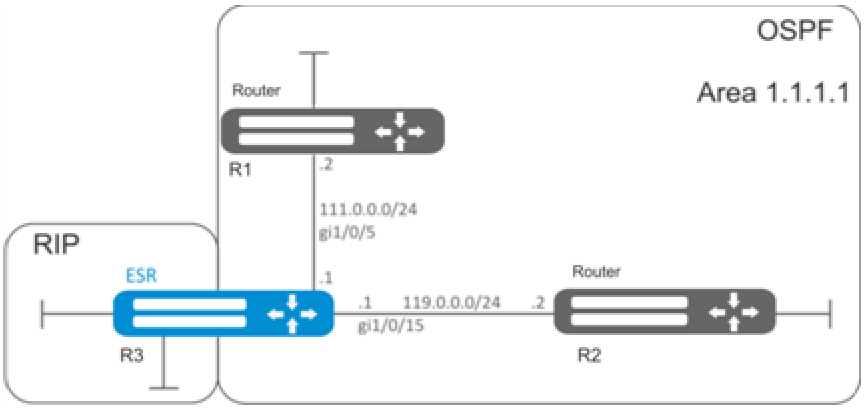
Step	Description	Command	Keys
38	Set the time interval in seconds after which the router re-sends a packet that has not received a delivery confirmation (for example, a DatabaseDescription packet or LinkStateRequest packets. Optional).	**esr(config-if-gi)# ip ospf restransmit-interval <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 5 seconds.
		esr(config-if-gi)# ipv6 ospf restransmit-interval <TIME>	
39	Set the time interval in seconds after which the router sends the next hello packet (optional).	**esr(config-if-gi)# ip ospf hello-interval <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 10 seconds.
		esr(config-if-gi)# ipv6 ospf hello-interval <TIME>	
40	Set the time interval in seconds after which the neighbor is considered to be idle (optional). This interval should be a multiple of the 'hello interval' value.	**esr(config-if-gi)# ip dead-interval <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 40 seconds.
		esr(config-if-gi)# ipv6 dead-interval <TIME>	
41	Set the time interval during which NBMA interface waits before sending a HELLO packet to a neighbor, even if the neighbor is inactive (optional).	**esr(config-if-gi)# ip poll-interval <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 120 seconds.
		esr(config-if-gi)# ipv6 poll-interval <TIME>	
42	Set static IP address of a neighbor to establish a relation in NMBA and P2MP (Point-to-MultiPoint) networks (optional).	**esr(config-if-gi)# ip ospf neighbor <IP> [eligible]**	<IP> – neighbor's 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 interface priority should be greater than zero.
		esr(config-if-gi)# ip ospf neighbor <IP> [eligible]	<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 interface priority should be greater than zero.

Step	Description	Command	Keys
43	Define the network type for OSPF neighborhood establishment (optional).	**esr(config-if-gi)# ip ospf network <TYPE>** **esr(config-if-gi)# ipv6 ospf network <TYPE>**	<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.
44	Set the router priority that is used for DR and BDR selection (optional).	**esr(config-if-gi)# ip ospf priority <VALUE>** **esr(config-if-gi)# ipv6 ospf priority <VALUE>**	<VALUE> – interface priority, takes values of [1..65535]. Default value: 120.
45	Set the metric size on the interface or tunnel (optional).	**esr(config-if-gi)# ip ospf cost <VALUE>** **esr(config-if-gi)# ipv6 ospf cost <VALUE>**	<VALUE> – metric size, takes values of [0..32767]. Default value: 150.
46	Enable BFD protocol for OSPF protocol.	**esr(config-if-gi)# ip ospf bfd-enable** **esr(config-if-gi)# ipv6 ospf bfd-enable**	

11.4.2 OSPF configuration example

Objective:

Configure OSPF protocol on the router in order to exchange the routing information with neighboring routers. The router should be in 1.1.1.1 identifier area and announce routes received via RIP.



Solution:

Pre-configure IP addresses on interfaces according to the network structure shown in figure above.

Create OSPF process with identifier 10 and proceed to the OSPF protocol configuration mode:

```
esr(config)# router ospf 10
```

Create and enable the required area:

```
esr(config-ospf)# area 1.1.1.1
esr(config-ospf-area)# enable
esr(config-ospf-area)# exit
```

Enable advertising of the routing information from RIP:

```
esr(config-ospf)# redistribute rip
```

Enable OSFP process:

```
esr(config-ospf)# enable
esr(config-ospf)# exit
```

neighboring routers are connected to gi1/0/5 and gi1/0/15 interfaces. To establish the neighboring with other routers, map them to OSPF process and the area. Next, enable OSPF routing for the interface.

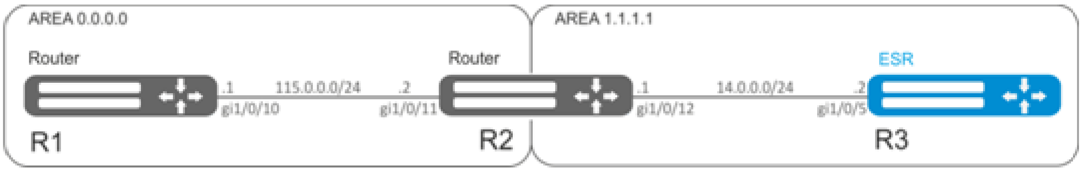
```
esr(config)# interface gigabitethernet 1/0/5
esr(config-if-gi)# ip ospf instance 10
esr(config-if-gi)# ip ospf area 1.1.1.1
esr(config-if-gi)# ip ospf
esr(config-if-gi)# exit
```

```
esr(config)# interface gigabitethernet 1/0/15
esr(config-if-gi)# ip ospf instance 10
esr(config-if-gi)# ip ospf area 1.1.1.1
esr(config-if-gi)# ip ospf
esr(config-if-gi)# exit
esr(config)# exit
```

11.4.3 OSPF stub area configuration example

Objective:

Change 1.1.1.1 area type, area should be stub.



Solution:

Pre-configure OSPF protocol and IP addresses on interfaces according to the network structure shown in figure above.

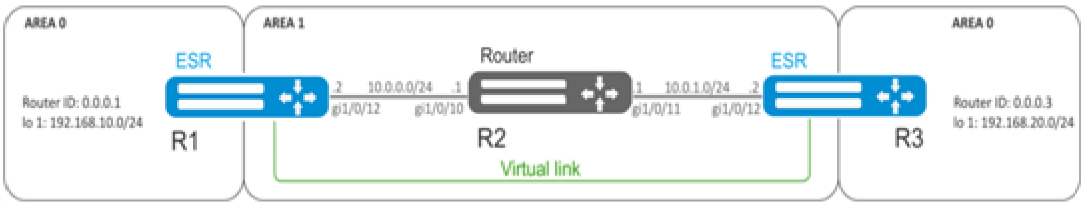
Change area type to stub. For each router from 1.1.1.1 area, execute the following command in the configuration mode:

```
esr(config-ospf-area)# area-type stub
```

11.4.4 Virtual link configuration example

Objective:

Merge two backbone areas using virtual link.



Solution:

Virtual link is a specialized connection that allows merging a split zone or connect a zone to the backbone zone trough the third zone. Virtual link is configured between two Area Border Routers (ABR).

Pre-configure OSPF protocol and IP addresses on interfaces according to the network structure shown in figure above.

For R1 router, proceed to 1.1.1.1 area configuration mode:

```
esr(config-ospf)# area 1.1.1.1
```

Create and enable virtual link with the identifier 0.0.0.3:

```
esr(config-ospf-area)# virtual-link 0.0.0.3
esr(config-ospf-vlink)# enable
```

For R3 router, proceed to 1.1.1.1 area configuration mode:

```
esr(config-ospf)# area 1.1.1.1
```

Create and enable virtual link with the identifier 0.0.0.1:

```
esr(config-ospf-area)# virtual-link 0.0.0.1
esr(config-ospf-vlink)# enable
```

Example of routing table on R1 router:

```
esr# show ip route
C     * 10.0.0.0/24        [0/0]   dev gi1/0/12,               [direct 00:49:34]
O     * 10.0.1.0/24        [150/20] via 10.0.0.1 on gi1/0/12,  [ospf1 00:49:53]  (0.0.0.3)
O     * 192.168.20.0/24    [150/30] via 10.0.0.1 on gi1/0/12,  [ospf1 00:50:15]  (0.0.0.3)
C     * 192.168.10.0/24    [0/0]   dev lo1,                    [direct 21:32:01]
```

Example of routing table on R3 router:

```
esr# show ip route
O     * 10.0.0.0/24        [150/20] via 10.0.1.1 on gi1/0/12,  [ospf1 14:38:35]  (0.0.0.2)
C     * 10.0.1.0/24        [0/0]   dev gi1/0/12,               [direct 14:35:34]
C     * 192.168.20.0/24    [0/0]   dev lo1,                    [direct 14:32:58]
O     * 192.168.10.0/24    [150/30] via 10.0.1.1 on gi1/0/12,  [ospf1 14:39:54]  (0.0.0.1)
```

Since OSPF considers virtual link as the part of the area, R1 routes received from R3 are marked as an intrazone and vice versa.

To view the neighbors, use the following command:

```
esr# show ip ospf neighbors 10
```

To view OSPF routing table, use the following command:

```
esr# show ip ospf 10
```

⚠ **Enable OSPF protocol (89) in the firewall.**

11.5 BGP configuration

BGP protocol is designed to exchange subnet reachability information among autonomous systems (AS), i.e. router groups united under a single technical control that uses interdomain routing protocol for defining packet delivery routes to other AS. Transmitted information includes a list of AS that are accessible through this system. Selection of the optimal routes is based on effective rules for the network.

11.5.1 Configuration algorithm

To establish a BGP session it is necessary to allow TCP port 179 on the firewall.

Step	Description	Command	Keys
1	Configure BGP precedence for the main routing table (optional).	**esr(config)# ip protocols bgp preference <VALUE>**	<VALUE> – protocol precedence, takes values in the range of [1..255]. Default value: BGP (170).
2	Configure the BGP routing table capacity (not required when using the global routing table).	**esr(config)# ip protocols bgp max-routes <VALUE>**	<VALUE> – amount of BGP routes in the routing table, takes values in the range of: • for ESR-1000/1200/1500/1511/1700/3100/3200 – [1..5000000]; • for ESR-20/21/30/100/200 – [1..2500000]; • for ESR-10/12V(F)/14VF/15 – [1..1000000]. The default value for the global routing table: • for ESR-1000/1200/1500/1511/1700/3100/3200 – [5000000]; • for ESR-20/21/30/100/200 – [2500000]; • for ESR-10/12V/12VF/14VF/15 – [1000000]. Default value for VRF: 0.
		esr(config)# ipv6 protocols bgp max-routes <VALUE>	
		esr(config-vrf)# ip protocols bgp max-routes <VALUE>	
		esr(config-vrf)# ipv6 protocols bgp max-routes <VALUE>	
3	Enable the output of BGP neighbor state information (optional).	**esr(config)# router bgp log-neighbor-changes**	
		esr(config)# ipv6 router bgp log-neighbor-changes	

Step	Description	Command	Keys
4	Enable ECMP and define the maximum amount of equal routes to a destination point.	**esr(config)# router bgp maximum-paths <VALUE>**	<VALUE> – amount of valid equal routes to the target, takes the values of [1..16].
5	Select the filtering method for the information transmitted between routers (mandatory when configuring eBGP to announce subnets).		
	If route-map-based filtering method is selected, create a list of rules that will be used to filter the advertised and received IP routes in the future.	**esr(config)# route-map <NAME>**	<NAME> – configured routing rule name, set by the string of up to 31 characters.
	Create rule	**(config-route-map)# rule <ORDER>**	<ORDER> – rule number, takes values of [1..10000].

Step	Description	Command	Keys
	Define the list of subnets affected by the rule.	**esr(config-route-map-rule)#match ip address { <ADDR/LEN> | object-group <OBJ-GRP-NETNAME> } [{ eq <LEN> | le <LEN> | ge <LEN 1> [le <LEN 2>] }]** **esr(config-route-map-rule)#match ipv6 address { <IPV6-ADDR/LEN> | object-group <OBJ-GRP-NETNAME> } [{ eq <LEN> | le <LEN> | ge <LEN 1> [le <LEN 2>] }]**	<ADDR/LEN> – IP address and subnet mask, in the format of. 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]; <IPV6-ADDR/LEN> – IPv6 address and subnet mask, in the format of: X:X:X:X::X/EE, where each X part takes values in hexadecimal format [0..FFFF] and EE takes values of [1..128]; <OBJ-GRP-NETNAME> – IP addresses profile name, set by the string of up to 31 characters*; <LEN>, <LEN 1>, <LEN 2> – prefix length, may take values [1..32] in prefix IP lists for IPv4 and [1..128] for IPv6; **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; **ge** <LEN 1> **le** <LEN 2> – When specifying a command, the prefix length must be greater than or equal to <LEN> but less than or equal to <LEN1>. * When using object-group filtering, they must be created in advance.
	Permit or deny action for the specified subnets in the rule.	**esr(config-route-map-rule)# action {deny | permit}**	

Step	Description	Command	Keys
	If prefix-list-based filtering method is selected, create a list of IP networks that will be used to filter the advertised and received IP routes in the future.	**esr(config)# ip prefix-list <NAME>**	<NAME> – name of a subnet list being configured, set by the string of up to 31 characters.
		esr(config)# ipv6 prefix-list <NAME>	

Step	Description	Command	Keys
	Permit or deny the prefixes lists.	**esr(config-pl)# permit { <ADDR/LEN> | object-group <OBJ-GRP-NETNAME>} [{ eq <LEN> | le <LEN> | ge <LEN 1> [le <LEN 2>] }]** **esr(config-pl)# deny {<ADDR/LEN> | object-group <OBJ-GRP-NETNAME>} [{ eq <LEN> | le <LEN> | ge <LEN 1> [le <LEN 2>] }]** **esr(config-ipv6-pl)# permit { <IPV6-ADDR/LEN> | object-group <OBJ-GRP-NETNAME>} [{ eq <LEN> | le <LEN> | ge <LEN 1> [le <LEN 2>] }]** **esr(config-ipv6-pl)# deny {<IPV6-ADDR/LEN> | object-group <OBJ-GRP-NETNAME> } [{ eq <LEN> | le <LEN> | ge <LEN 1> [le <LEN 2>] }]**	<ADDR/LEN> – IP address and subnet mask, in the format of. 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]; <IPV6-ADDR/LEN> – IPv6 address and subnet mask, in the format of: X:X:X:X::X/EE, where each X part takes values in hexadecimal format [0..FFFF] and EE takes values of [1..128]; <OBJ-GRP-NETNAME> – IP addresses profile name, set by the string of up to 31 characters*; <LEN>, <LEN 1>, <LEN 2> – prefix length, may take values [1..32] in prefix IP lists for IPv4 and [1..128] for IPv6; **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; **ge** <LEN 1> **le** <LEN 2> – When specifying a command, the prefix length must be greater than or equal to <LEN> but less than or equal to <LEN1>. * When using object-group filtering, they must be created in advance.

Step	Description	Command	Keys
6	Add BGP process to the system and switch to the BGP process parameters configuration mode.	**esr(config)# router bgp <AS>**	<AS> – stand alone system number, takes values of [1..4294967295].
7	Set the router identifier.	**esr(config-bgp)# router-id { <ID> | <IF> | <TUN> }**	<ID> – router identifier, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. <IF> – interface specified in form given in the Types and naming procedure of router interfaces section of CLI command reference guide. <TUN> – tunnel name specified in form given in the Types and naming procedure of router tunnels section.
8	Set the Route-Reflector identifier of the cluster to which the router BGP process belongs (if necessary).	**esr(config-bgp)# cluster-id <ID>**	<ID> – Route-Reflector cluster identifier, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
9	Enable generation and sending of a default route, if the default route is in the FIB routing table (optional).	**esr(config-bgp)# default-information-originate**	
10	Set the time interval after which the connection with the opposing party is checked (optional).	**esr(config-bgp-af)# timers keepalive <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 60 seconds.
11	Set time interval after which the opposing party is considered to be unavailable (optional).	**esr(config-bgp-af)# timers holdtime <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 180 seconds.
12	Set the time of minimum and maximum delay during which it is prohibited to establish a connection in order to prevent frequent disconnections (optional).	**esr(config-bgp-af)# timers error-wait <TIME1> <TIME2>**	<TIME1> – minimum delay time in seconds, takes values of [1..65535]. <TIME2> – maximum delay time in seconds, takes values of [1..65535].

Step	Description	Command	Keys
13	Define the global algorithm of neighbor authentication (if necessary).	**esr(config-bgp)# authentication algorithm <ALGORITHM>**	<ALGORITHM> – encryption algorithm: • md5 – password is encrypted by md5 algorithm. Default value: encryption is not used.
14	Set a global password for authentication with neighbors (used in conjunction with 'authentication algorithm').	**esr(config-bgp)# authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }**	<CLEAR-TEXT> – password, set by the string of 8 to 16 characters; <ENCRYPTED-TEXT> – encrypted password of 8 to 16 bytes (from 16 to 32 characters) in hexadecimal format (0xYYYY ...) or (YYYY ...).
15	Enable BGP process.	**esr(config-bgp)# enable**	
16	Define the type of configured routing information and switch to this configuration mode.	**esr(config-bgp)# address-family { ipv4 | ipv6 } unicast**	**ipv 4** – IPv4 family; **ipv 6** – IPv6 family;
17	Enable route advertising by BGP process obtained alternatively (if necessary).	**esr(config-bgp-af)# redistribute static [route-map <NAME>]**	<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.
		esr(config-bgp-af)# redistribute connected [route-map <NAME>]	<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.
		esr(config-bgp-af)# redistribute rip [route-map <NAME>]	<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.

Step	Description	Command	Keys
		esr(config-bgp-af)# redistribute ospf <ID> <ROUTE-TYPE 1> [<ROUTE-TYPE 2>] [<ROUTE-TYPE 3>] [<ROUTE-TYPE 4>] [route-map <NAME>]	<ID> – process number, takes values of {1..65535}; <ROUTE-TYPE> – route type: • **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.
		esr(config-bgp-af)# redistribute bgp <AS> [route-map <NAME>]	<AS> – stand alone system number, takes values of [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.
18	Enable subnets advertising.	**esr(config-bgp-af)# network <ADDR/LEN>**	<ADDR/LEN> – subnet address, set in one of the following formats: • 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]; • 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].

Step	Description	Command	Keys
19	Exit global BGP process route information advertisement configuration mode.	**esr(config-bgp-af)# exit**	
20	Add BGP neighbor and switch to the BGP process parameters configuration mode.	**esr(config-bgp)# neighbor <ADDR>|<IPV6-ADDR>**	<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].
21	Specify neighbor description (optional).	**esr(config-bgp-neighbor)# description <DESCRIPTION>**	<DESCRIPTION> – neighbor description, set by the string of up to 255 characters.
22	Set the time interval after which the connection with the opposing party is checked (optional).	**esr(config-bgp-neighbor)# timers keepalive <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 60 seconds.
23	Set time interval after which the opposing party is considered to be unavailable (optional).	**esr(config-bgp- neighbor)# timers holdtime <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 180 seconds.
24	Set the time of minimum and maximum delay during which it is prohibited to establish a connection in order to prevent frequent disconnections (optional).	**esr(config-bgp-af)# timers error-wait <TIME1> <TIME2>**	<TIME1> – minimum delay time in seconds, takes values of [1..65535]. <TIME2> – maximum delay time in seconds, takes values of [1..65535]. Default value: 60 and 300 seconds
25	Set the number of BGP neighbor stand alone system.	**esr(config-bgp-neighbor)# remote-as <AS>**	<AS> – stand alone system number, takes values of [1..4294967295].
26	Allow connections to neighbors that are located not in directly connected subnets (optional).	**esr(config-bgp-neighbor)# ebgp-multihop <NUM>**	<NUM> – maximum amount of hops when installing EBGP (used for TTL).
27	Specify BGP neighbor as a Route-Reflector client (optional).	**esr(config-bgp-neighbor)# route-reflector-client**	

Step	Description	Command	Keys
28	Set IP/IPv6 router address that will be used as source IP/IPv6 address in transmitted BGP route information updates (optional).	**esr(config-bgp-neighbor)# update-source { <ADDR> | <IPV6-ADDR> }**	<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]. <IF> – interface specified in form given in the Types and naming procedure of router interfaces section of CLI command reference guide. <TUN> – tunnel name specified in form given in the Types and naming procedure of router tunnels section.
29	Enable the mode in which the reception of routes in the BGP attribute, AS Path of which includes the numbers of process stand alone system, is allowed (optional).	**esr(config-bgp-neighbor)# allow-local-as <NUMBER>**	<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].
30	Enable the BFD protocol on the configured BGP neighbor (optional, used in conjunction with the update-source parameter).	**esr(config-bgp-neighbor)# bfd-enable**	
31	Specify neighbor authentication algorithm (optional).	**esr(config-bgp-neighbor)# authentication algorithm <ALGORITHM>**	<ALGORITHM> – encryption algorithm: md5 – password is encrypted by md5 algorithm.
32	Set the password for neighbor authentication (optional).	**esr(config-bgp-neighbor)# authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }**	<CLEAR-TEXT> – password, set by the string of 8 to 16 characters; <ENCRYPTED-TEXT> – encrypted password of 8 to 16 bytes (from 16 to 32 characters) in hexadecimal format (0xYYYY ...) or (YYYY ...).
33	Make neighborhood active.	**esr(config-bgp-neighbor)# enable**	

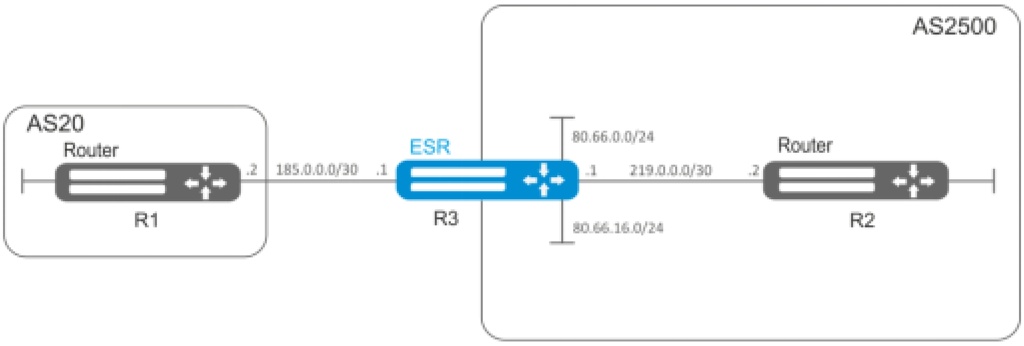
Step	Description	Command	Keys
34	Define the type of neighbor configured routing information and switch to this configuration mode.	**esr(config-bgp-neighbor)# address-family { ipv4 | ipv6 | vpnv4 } unicast**	**ipv 4** – IPv4 family; **ipv 6** – IPv6 family; **vpnv4** – VPNv4 family.
35	If prefix list filtering mode is selected, add subnet filtering in incoming or outgoing updates (Mandatory when configuring eBGP for subnet advertisement).	**esr(config-bgp-neighbor-af)# prefix-list <PREFIX-LIST-NAME> { in | out }**	<PREFIX-LIST-NAME> – name of a subnet list being configured, set by the string of up to 31 characters. **in** – incoming routes filtering; **out** – outgoing routes filtering.
36	Set the mode in which the default route is always sent to the BGP neighbor in the update along with other routes (optional, none for vpnv4).	**esr(config-bgp-neighbor-af)# default-originate**	
37	Set 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 (optional, none for vpnv4).	**esr(config-bgp-neighbor-af)# next-hop-self**	
38	Define the precedence of the routes received from a neighbor (optional).	**esr(config-bgp-neighbor-af)# preference <VALUE>**	<VALUE> – neighbor routes precedence, takes values in the range of [1..255]. Default value: 170.
39	Set 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) (optional, none for vpnv4).	**esr(config-bgp-neighbor-af)# remove-private-as [{ all | nearest | replace }]**	**all** – remove all private AS number from AS-path; **nearest** – replace the nearest private AS in the AS-path with a nearby public AS; **replace** – replace all private AS numbers with the number of the current BGP process. Default value: all.
40	Enable routing information exchange.	**esr(config-bgp-neighbor-af)# enable**	

When configuring iBGP, it is common to have multiple BGP neighbors with the same parameters within a single BGP process. To avoid configuration redundancy, it is recommended to use BGP peer-group in which common parameters can be described and it is easy to identify the BGP peer-group membership in the BGP neighbor configuration.

11.5.2 Configuration example

Objective:

Configure BGP on the R3 router with the following parameters:



- own subnets: 80.66.0.0/24, 80.66.16.0/24;
- advertising of directly connected subnets;
- proprietary AS 2500;
- first neighboring – subnet 219.0.0.0/30, proprietary IP address 219.0.0.1, neighbor IP address 219.0.0.2, AS2500;
- second neighboring – subnet 185.0.0.0/30, proprietary IP address 185.0.0.1, neighbor IP address 185.0.0.2, AS20.

Solution:

Configure required network interfaces:

```
esr-R3(config)# interface gigabitethernet 1/0/1
esr-R3(config-if-gi)# ip address 185.0.0.1/30
esr-R3(config-if-gi)# exit
esr-R3(config)# interface gigabitethernet 1/0/2
esr-R3(config-if-gi)# ip address 219.0.0.1/30
esr-R3(config-if-gi)# exit
esr-R3(config)# interface gigabitethernet 1/0/3
esr-R3(config-if-gi)# ip address 80.66.0.1/24
esr-R3(config-if-gi)# exit
esr-R3(config)# interface gigabitethernet 1/0/4
esr-R3(config-if-gi)# ip address 80.66.16.1/24
esr-R3(config-if-gi)# exit
```

Configure the firewall to receive BGP traffic from the WAN security zone:

```
esr-R3(config)# object-group service og_bgp
esr-R3(config-object-group-service)# port-range 179
esr-R3(config-object-group-service)# exit
esr-R3(config)# security zone wan
esr-R3(config-zone)# exit
esr-R3(config)# security zone-pair wan self
esr-R3(config-zone-pair)# rule 100
esr-R3(config-zone-pair-rule)# match protocol tcp
esr-R3(config-zone-pair-rule)# match destination-port og_bgp
esr-R3(config-zone-pair-rule)# action permit
esr-R3(config-zone-pair-rule)# enable
esr-R3(config-zone-pair-rule)# exit
esr-R3(config-zone-pair)# exit
```

Specify that the interfaces belong to the security zone:

```
esr-R3(config)# interface gigabitethernet 1/0/1
esr-R3(config-if-gi)# security-zone wan
esr-R3(config-if-gi)# exit
esr-R3(config)# interface gigabitethernet 1/0/2
esr-R3(config-if-gi)# security-zone wan
esr-R3(config-if-gi)# exit
```

Create a route-map, which will be used later when configuring enabling advertising to routers from another AS.

```
esr-R3(config)# route-map bgp-general
esr-R3(config-route-map)# rule 1
esr-R3(config-route-map-rule)# match ip address 80.66.0.0/24
esr-R3(config-route-map-rule)# action permit
esr-R3(config-route-map-rule)# exit
esr-R3(config-route-map)# rule 2
esr-R3(config-route-map-rule)# match ip address 80.66.16.0/24
esr-R3(config-route-map-rule)# action permit
esr-R3(config-route-map-rule)# exit
esr-R3(config-route-map)# exit
```

Create BGP process for AS 2500 and enter process parameters' configuration mode:

```
esr(config)# router bgp 2500
```

Configure advertising of directly connected subnets:

```
esr-R3(config-bgp)# address-family ipv4 unicast
esr-R3(config-bgp-af)# redistribute connected
esr-R3(config-bgp-af)# exit
```

Create a neighborhood with the R2 router via iBGP:

```
esr-R3(config-bgp)# neighbor 219.0.0.2
esr-R3(config-bgp-neighbor)# remote-as 2500
esr-R3(config-bgp-neighbor)# enable
```

Enable IPv4 route exchange:

```
esr-R3(config-bgp-neighbor)# address-family ipv4 unicast
esr-R3(config-bgp-neighbor-af)# enable
esr-R3(config-bgp-neighbor-af)# exit
esr-R3(config-bgp-neighbor)# exit
```

Create a neighborhood with the R1 router via eBGP:

```
esr-R3(config-bgp)# neighbor 185.0.0.2
esr-R3(config-bgp-neighbor)# remote-as 20
esr-R3(config-bgp-neighbor)# enable
```

Enable the exchange of ipv4 routes, permitting the necessary routes for advertising by means of a previously prepared route-map:

```
esr-R3(config-bgp-neighbor)# address-family ipv4 unicast
esr-R3(config-bgp-neighbor-af)# route-map bgp-general out
esr-R3(config-bgp-neighbor-af)# enable
esr-R3(config-bgp-neighbor-af)# exit
esr-R3(config-bgp-neighbor)# exit
```

Enable protocol operation:

```
esr-R3(config-bgp)# enable
esr-R3(config-bgp)# exit
```

To view BGP peers information, use the following command:

```
esr# show bgp neighbors
```

To view BGP routing table, use the following command:

```
esr# show bgp ipv4 unicast
```

11.5.3 BGP best route selection policy

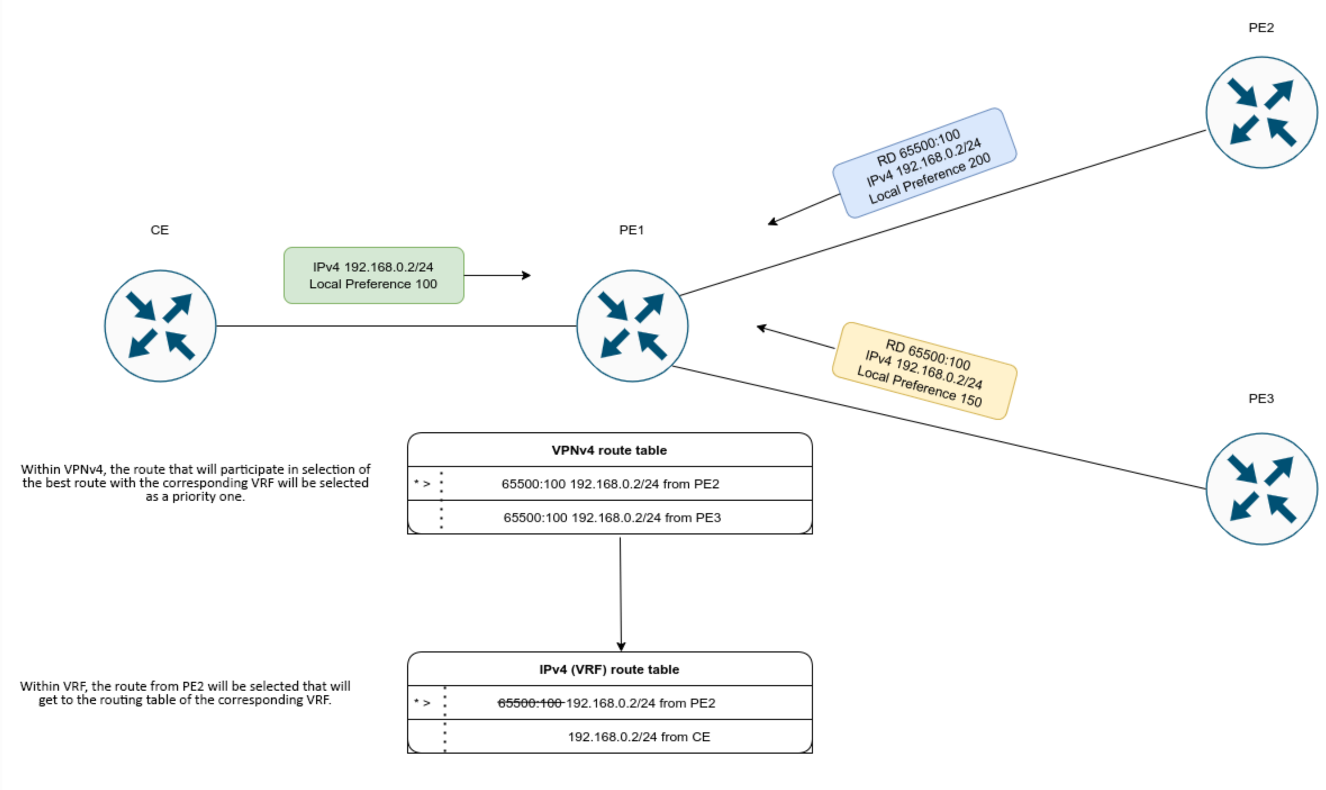
During operation, BGP calculates one best route to each received subnet. If there is no higher priority route learned by another routing protocol before this subnet, then the route is installed in the routing table.

If ECMP mechanism is enabled (router maximum-paths ..), then the routing table can get up to 16 active routes to the same subnet.
When advertising BGP to peers, the attributes of the best route will be used.

Algorithm of best route selection in BGP is given below.

The algorithm is used for the following address family: unicast IPv4, unicast IPv6, VPNv4 unicast, VPLS.

ⓘ **For VPNv4 routes, selection of the best route is as follows:**
First, the choice of the best route occurs within <u>its RD</u>. Further, within the framework of VRF, where it will get in accordance with its RT.



First of all, next-hop of the route is checked. If there is a connected route to the next-hop, it is considered available.

1. The route marked as 'stale' is less priority than a route without such a label. The route is marked as 'stale' in the process of LLGR technology operation (More detailed);
2. The value of the Weight attribute is compared – the route with the higher value becomes the best;
3. The value of the Local preferences attribute is compared – the route with the higher value becomes the best;
4. The length of the AS-path is compared –the route with the least number of hops becomes the best;
5. The value of the Origin attribute is compared – IGP is the highest priority. EGP takes precedence over Incomplete;
6. The value of the multiple exit discriminator (MED) attribute is compared – the lowest attribute value has the highest priority;
7. The route received from the EBGP peer takes precedence over route received from IBGP peers;
   7.1 If ECMP is enabled, then no further comparisons are made and the route (multipath) will be included in the routing table;
8. The value of the Router-Id parameter is compared – the route received from the BGP neighbor with the lowest Router-Id has the highest priority;
9. The number of addresses in the Cluster list is compared – the route with the least number of addresses becomes the best;
10. The addresses of the BGP peers are compared – the route received from the BGP peer with the least address has the highest priority;

In the output of routing information for a particular prefix, the best route will be marked as 'Best':

```
ESR# show bgp ipv4 unicast 192.0.2.0/24
192.0.2.0/24 via 100.64.28.1 on gi1/0/1.2800   [bgp65514 2022-05-22] (65041i)
    Administrative Distance: 170
    Type:                    unicast
    Origin:                  IGP
    AS PATH:                 65054 65055 65056 65077 65098 65059
    Next Hop:                100.64.28.1
    Local Preference:        100
    Community:               (3356:2) (3356:22) (3356:86) (3356:501) (3356:666) (3356:903)
(3356:2065)
                             (12389:6) (65000:64990)
    Weight:                  0
    Valid
192.0.2.0/24 via 101.7.0.1 on gi1/0/1.2800   [bgp65514 2022-05-22] (65041i)
    Administrative Distance: 170
    Type:                    unicast
    Origin:                  IGP
    AS PATH:                 65020 65030
    Next Hop:                101.7.0.1
    Local Preference:        200
    Community:               (3356:2) (3356:22) (3356:86) (3356:501) (3356:666) (3356:903)
(3356:2065)
                             (12389:6) (65000:64990)
    Weight:                  0
    Valid,Best
```

11.6 BFD configuration

BFD (Bidirectional Forwarding Detection) is a protocol operating over other protocols and allowing to reduce the problem detection time to 50 msec. BFD is two-party protocol, it requires the configuration of both routers (both routers generate BFD packets and respond to each other).

By default, the session is established in the following mode:

Protocol	Mode
iBGP	multi-hop
eBGP	single-hop
eBGP multi-hop	multi-hop
OSPF	single-hop
Static route	single-hop

To change the behavior (mode), manually override the session parameters by specifying the required mode. Example is given below.

Example of establishing iBGP neighborhood and enabling BFD for it:

```
ESR# show running-config routing bgp
router bgp 65516
  neighbor 10.100.0.2
    remote-as 65515
    update-source 10.100.0.1
    bfd-enable
    enable
  exit
  enable
exit

ESR# show bfd neighbors  10.100.0.2
Neighbor address:                    10.100.0.2
Local address:                       10.100.0.1
Interface:                           --
Remote discriminator:                3751534121
Local discriminator:                 1670865501
State:                               Up
Session type:                        Control
Session mode:                        Single-hop
Local diagnostic code:               No Diagnostic
Remote diagnostic code:              No Diagnostic
Minimal Tx Interval:                 300 ms
Minimal Rx Interval:                 300 ms
Multiplier:                          5
Actual Tx Interval:                  300 ms
Actual Detection Interval:           1500 ms
Number of transmitted packets:       1149
Number of received packets:          1153
Uptime:                              2m
Client:                              BGP
Last received packet:
    Desired Min Tx Interval:         300 ms
    Required Min Rx Interval:        300 ms
    Multiplier:                      5
```

By default BFD is set in single-hop mode. Switch to the multi-hop mode:

```
ESR(config)# ip bfd neighbor 10.100.0.2 local-address 10.100.0.1 multihop
ESR(config)# do commit
ESR(config)# do confirm
```

Both devices have to be configured. After session re-establishing, its mode will change to multi-hop:

```
esr-200# sh bfd neighbors  10.100.0.2
Neighbor address:                  10.100.0.2
Local address:                     10.100.0.1
Interface:                         --
Remote discriminator:              3751534121
Local discriminator:               1670865501
State:                             Up
Session type:                      Control
Session mode:                      Multi-hop
Local diagnostic code:             No Diagnostic
Remote diagnostic code:            No Diagnostic
Minimal Tx Interval:               300 ms
Minimal Rx Interval:               300 ms
Multiplier:                        5
Actual Tx Interval:                300 ms
Actual Detection Interval:         1500 ms
Number of transmitted packets:     9
Number of received packets:        11
Uptime:                            2m
Client:                            BGP
Last received packet:
    Desired Min Tx Interval:       300 ms
    Required Min Rx Interval:      300 ms
    Multiplier:                    5
```

11.6.1 Timers configuration

Timer value is individual for each network and largely depends on its parameters. In case of frequent BFD flapping, it is recommended to increase the value of the timers.

Timers, regardless of the protocol operation mode (single or multi-hop mode), can be configured in the context of the global configuration or on specific interfaces. The setting on the interfaces has the highest priority.

```
ESR(config)# ip bfd min-tx-interval 1000
ESR(config)# ip bfd min-rx-interval 1000
ESR(config)# do commmit

ESR# sh ip bfd
Minimum RX interval: 1000 ms
Minimum TX interval: 1000 ms
Idle TX interval:    1000 ms
Multiplier:          5 packets
Passive:             No
```

After BFD session is established each side individually calculates its own Tx Interval and Detection Interval. Tx Interval — the largest value is selected from the local Tx Interval and remote RX Interval. Detection Interval is calculated using the following formula: Detection Interval = remoteMultiplier * MAX(RxLocal || TxRemote), where remoteMultiplier is the value of the Multiplier of the remote side, RxLocal is the local Tx Interval, TxRemote is the Tx Interval of the remote side.

Locally configured timers, remote side timers, as well as calculated timers, can be viewed as follows:

```
esr-200# sh bfd neighbors 10.100.0.2
Neighbor address:                   10.100.0.2
Local address:                      10.100.0.1
Interface:                          --
Remote discriminator:               3751534121
Local discriminator:                1670865501
State:                              Up
Session type:                       Control
Session mode:                       Multi-hop
Local diagnostic code:              No Diagnostic
Remote diagnostic code:             No Diagnostic
Minimal Tx Interval:                300 ms       <----  Local  Tx Interval
Minimal Rx Interval:                300 ms       <----  Local  Rx Interval
Multiplier:                         5            <----  Local  Multiplier
Actual Tx Interval:                 300 ms       <----  Calculated Tx Interval
Actual Detection Interval:          1500 ms      <----  Calculated Detection Interval
Number of transmitted packets:      21781
Number of received packets:         21804
Uptime:                             1d21h54m
Client:                             BGP
Last received packet:
    Desired Min Tx Interval:        300 ms       <----
    Required Min Rx Interval:       300 ms       <----    Remote side timers
    Multiplier:                     5            <----
```

11.6.2 Configuration algorithm

Step	Description	Command	Keys
1	Enable BFD for OSFP on the interface.	**esr(config-if-gi)# ip ospf bfd-enable**	
2	Enable BFD for BGP neighbor on the interface.	**esr(config-bgp-neighbor)# bfd-enable**	
3	Set the interval after which the BFD message is sent to the neighbor. Globally (optional).	**esr(config)# ip bfd idle-tx-interval <TIMEOUT>**	<TIMEOUT> – interval after which the BFD packet should be sent, takes values in milliseconds in the range of [200..65535] for ESR-1000/1200/1500/1511/1700/3100/3200 and [300..65535] for ESR-10/12V(F)/14VF/15/20/21/30/100/200 Default value: 1 second.
4	Enable the logging of BFD protocol state changes (optional).	**esr(config)# ip bfd log-adjacency-changes**	

Step	Description	Command	Keys
5	Set the minimum interval after which the neighbor should generate BFD message. Globally (optional).	**esr(config)# ip bfd min-rx-interval <TIMEOUT>**	<TIMEOUT> – interval after which the BFD message should be sent by the neighbor, takes values in milliseconds in the range of [200..65535] for ESR-1000/1200/1500/1511/1700/3100/3200 and [300..65535] for ESR-10/12V(F)/14V(F)/20/21/30/100/200 By default: • 300 ms on ESR-10/12V(F)/14VF/15/20/21/30/100/200; • 200 ms on ESR-1000/1200/1500/1511/1700/3100/3200.
6	Set the minimum interval after which the BFD message is sent to the neighbor. Globally (optional).	**esr(config)# ip bfd min-tx-interval <TIMEOUT>**	<TIMEOUT> – interval after which the BFD message should be sent by the neighbor, takes values in milliseconds in the range of [200..65535] for ESR-1000/1200/1500/1511/1700/3100/3200 and [300..65535] for ESR-10/12V(F)/14VF/15/20/21/30/100/200. By default: • 300 ms on ESR-10/12V(F)/14VF/20/21/100/200 • 200 ms on ESR-1000/1200/1500/1511/1700/3100
7	Set the amount of dropped packets, at which the BFD neighbor is considered to be unavailable. Globally.	**esr(config)# ip bfd multiplier <COUNT>**	<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.

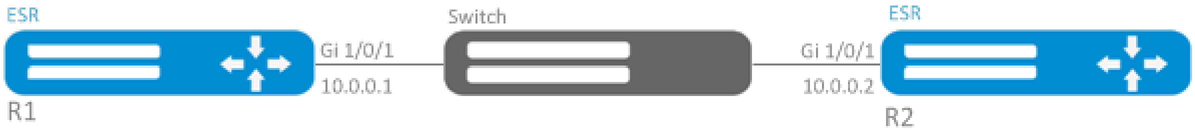
Step	Description	Command	Keys
8	Enable BFD operation with the specified IP address.	**esr(config)# ip bfd neighbor <ADDR> [{ interface <IF> | tunnel <TUN> }] [local-address <ADDR> [multihop]] [vrf <VRF>]**	<ADDR> – gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]; <IF> – interface or interface group; <TUN> – tunnel type and number. <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.
9	Switch BFD session to the passive mode, so that BFD messages will not be sent until the messages from BFD neighbor are received. Globally (optional).	**esr(config)# ip bfd passive**	
10	Set the interval after which the BFD message is sent to the neighbor. On the interface (optional).	**esr(config-if-gi)# ip bfd idle-tx-interval <TIMEOUT>**	<TIMEOUT> – interval after which the BFD packet should be sent, takes values in milliseconds in the range of [200..65535] for ESR-1000/1200/1500/1511/1700/3100/3200 and [300..65535] for ESR-10/12V(F)/14VF/15/20/21/30/100/200. Default value: 1 second.

Step	Description	Command	Keys
11	Set the minimum interval after which the neighbor should generate BFD message. On the interface (optional).	**esr(config-if-gi)# ip bfd min-rx-interval <TIMEOUT>**	<TIMEOUT> – interval after which the BFD message should be sent by the neighbor, takes values in milliseconds in the range of [200..65535] for ESR-1000/1200/1500/1511/1700/3100/3200 and [300..65535] for ESR-10/12V(F)/14V(F)/15/20/21/30/100/200. By default: • 300 ms on ESR-10/12V(F)/14VF/15/20/21/30/100/200; • 200 ms on ESR-1000/1200/1500/1511/1700/3100/3200.
12	Set the minimum interval after which the BFD message is sent to the neighbor. On the interface (optional).	**esr(config-if-gi)# ip bfd min-tx-interval <TIMEOUT>**	<TIMEOUT> – interval after which the BFD message should be sent by the neighbor, takes values in milliseconds in the range of [200..65535] for ESR-1000/1200/1500/1511/1700/3100/3200 and [300..65535] for ESR-10/12V(F)/14V(F)/15/20/21/30/100/200. By default: • 300 ms on ESR-10/12V(F)/14VF/15/20/21/30/100/200; • 200 ms on ESR-1000/1200/1500/1511/1700/3100/3200.
13	Set the amount of dropped packets, at which the BFD neighbor is considered to be unavailable. On the interface (optional).	**esr(config-if-gi)# ip bfd multiplier <COUNT>**	<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.
14	Switch BFD session to the passive mode, so that BFD messages will not be sent until the messages from BFD neighbor are received. On the interface (optional).	**esr(config-if-gi)# ip bfd passive**	

11.6.3 BFD with BGP configuration example

Objective:

Configure eBGP between ESR R1 and R2 and enable BFD.



Solution:

1. R1 configuration
   Preconfigure Gi1/0/1 interface:

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# ip firewall disable
esr(config-if-gi)# ip address 10.0.0.1/24
```

   Configure eBGP with BFD:

```
esr(config)# router bgp 100
esr(config-bgp)# neighbor 10.0.0.2
esr(config-bgp-neighbor)# remote-as 200
esr(config-bgp-neighbor)# update-source 10.0.0.1
esr(config-bgp-neighbor)# bfd-enable
esr(config-bgp-neighbor)# enable
esr(config-bgp-neighbor)# exit
esr(config-bgp)# enable
esr(config-bgp)# exit
```

2. R2 configuration
   Preconfigure Gi1/0/1 interface:

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# ip firewall disable
esr(config-if-gi)# ip address 10.0.0.2/24
```

   Configure eBGP with BFD:

```
esr(config)# router bgp 200
esr(config-bgp)# neighbor 10.0.0.1
esr(config-bgp-neighbor)# remote-as 100
esr(config-bgp-neighbor)# update-source 10.0.0.2
esr(config-bgp-neighbor)# bfd-enable
esr(config-bgp-neighbor)# enable
esr(config-bgp-neighbor)# exit
esr(config-bgp)# enable
esr(config-bgp)# exit
```

11.7 PBR routing policy configuration

11.7.1 Configuration algorithm of Route-map for BGP

Route-maps may serve as filters processing routing information when it is received from or sent to the neighboring device. Processing may include filtering based on various route criteria and setting attributes (MED, AS-PATH, community, LocalPreference, etc.) for the respective routes.

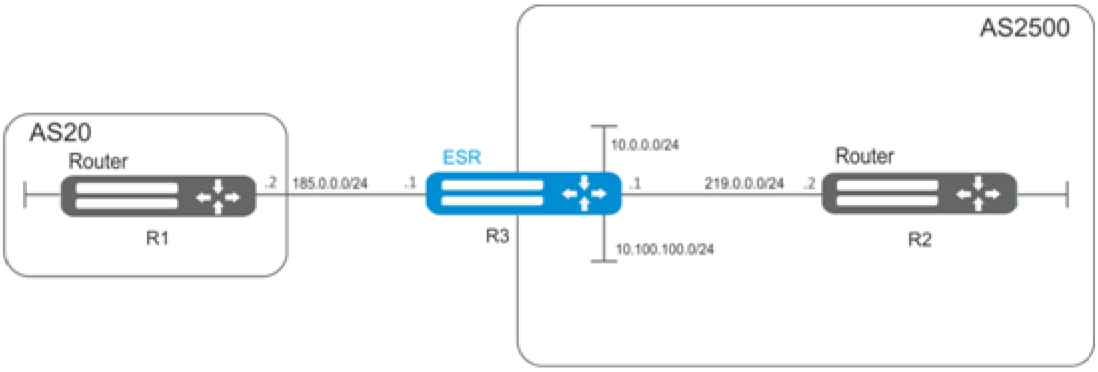
Step	Description	Command	Keys
1	Create a route map for IP routes filtering and modification.	**esr(config)# route-map <NAME>**	<NAME> – router map name, set by the string of up to 31 characters.
2	Create a route map rule.	**esr(config-route-map)# rule <ORDER>**	<ORDER> – rule number, takes values of [1..10000].
3	Specify the action that should be applied for routing information.	**esr(config-route-map-rule)# action <ACT>**	<ACT> – allocated action: • **permit** – routing information reception or advertising is permitted; • **deny** – denied.
4	Set BGPAS-Path attribute value in the route for which the rule should work (optional).	**esr(config-route-map-rule)# match as-path [begin | end | contain] <AS-PATH>**	<AS-PATH> – list of stand alone system numbers, defined as AS,AS,AS, takes values of [1..4294967295]. Optional parameters: • **begin** – attribute value begins with the specified AS numbers; • **end** – attribute value ends with the specified AS numbers; • **contain** – attribute value includes the specified AS numbers list.
5	Set BGPCommunity attribute value for which the rule should work (optional).	**esr(config-route-map-rule)# match community <COMMUNITY-LIST>**	<COMMUNITY-LIST> – community list, defined as AS:N,AS:N, takes values of [1..4294967295]. Up to 64 communities can be specified.
6	BGPExtendedCommunity attribute value for which the rule should work (optional).	**esr(config-route-map-rule)# match extcommunity <EXTCOMMUNITY-LIST>**	<EXTCOMMUNITY-LIST> – extcommunity list, defined as KIND:AS:N, KIND:AS:N, where KIND – extcommunity type: • **rt** (Route Target); • **ro** (Route Origin); N – extcommunity number, takes values of [1..65535].
7	Set IP addresses profile including destination subnet values in the route (optional).	**esr(config-route-map-rule)# match ip address object-group <OBJ-GROUP-NETWORK-NAME>**	<OBJ-GROUP-NETWORK-NAME> – name of the IP addresses profile that includes destination subnets prefixes, set by the string of up to 31 characters.

Step	Description	Command	Keys
		esr(config-route-map-rule)# match ipv6 address object-group <OBJ-GROUP-NETWORK-NAME>	
8	Set IP addresses profile that includes BGPNext-Hop attribute value in the route for which the rule should work (optional).	**esr(config-route-map-rule)# match ip next-hop object-group <OBJ-GROUP-NETWORK-NAME>**	<OBJ-GROUP-NETWORK-NAME> – name of the IP addresses profile that includes destination subnets prefixes, set by the string of up to 31 characters.
		esr(config-route-map-rule)# match ipv6 next-hop object-group <OBJ-GROUP-NETWORK-NAME>	
9	Set the profile that includes IP addresses of the router having advertised the route for which the rule should work (optional).	**esr(config-route-map-rule)# match ip route-source object-group <OBJ-GROUP-NETWORK-NAME>**	<OBJ-GROUP-NETWORK-NAME> – name of the IP addresses profile that includes destination subnets prefixes, set by the string of up to 31 characters.
		esr(config-route-map-rule)# match ipv6 route-source object-group <OBJ-GROUP-NETWORK-NAME>	
10	Specify ACL group for which the rule should work (optional).	**esr(config-route-map-rule)# match access-group <NAME>**	<NAME> – access control list name, set by the string of up to 31 characters.
11	Set BGP MED attribute value in the route for which the rule should work (optional).	**esr(config-route-map-rule)# match metric bgp <METRIC>**	<METRIC> – BGP MED attribute value, takes values in the range of [0..4294967295].
12	Set OSPF Metric attribute value in the route for which the rule should work.	**esr(config-route-map-rule)# match metric ospf <TYPE> <METRIC>**	<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].
13	Set RIP Metric attribute value in the route for which the rule should work.	**esr(config-route-map-rule)# match metric rip <METRIC>**	<METRIC> – RIP Metric attribute value, takes values in the range of [0..16].
14	Set OSPF Tag attribute value in the route for which the rule should work.	**esr(config-route-map-rule)# match tag ospf <TAG>**	<TAG> – OSPF Tag attribute value, takes values in the range of [0..4294967295].

Step	Description	Command	Keys
15	Set RIP Tag attribute value in the route for which the rule should work.	**esr(config-route-map-rule)# match tag rip <TAG>**	<RIP> – RIP Tag attribute value, takes values in the range of [0..65535].
16	Set BGP AS-Path attribute value that will be added to the beginning of AS-Path list (optional).	**esr(config-route-map-rule)# action set as-path prepend <AS-PATH> {track <TRACK-ID>}**	<AS-PATH> – stand alone systems number list that will be added to the current value in the route. Set as AS, AS, AS, takes values of [1..4294967295]. <TCACK-ID> – vrrp-tracking identifier that provides the specified action execution. Changes in the range of [1..60].
17	Set BGP Community attribute value that will be specified in the route (optional).	**esr(config-route-map-rule)# action set community {COMMUNITY-LIST> | no-advertise | no-export }**	<COMMUNITY-LIST> – community list, defined as AS:N,AS:N, where each part takes values of [1..65535]. • **no-advertise** – routes transmitted with the given community should not be advertised to other BGP neighbors; • **no-export** – routes transmitted with the given community should not be advertised to eBGP neighbors but can be advertised to external neighbors in the confederation.
18	Set BGP ExtCommunity attribute value that will be specified in the route (optional).	**esr(config-route-map-rule)# action set extcommunity <EXTCOMMUNITY-LIST>**	<EXTCOMMUNITY-LIST> – extcommunity list, defined as KIND:AS:N, KIND:AS:N, where KIND – extcommunity type: • **rt** (Route Target); • **ro** (Route Origin); N – extcommunity number, takes values of [1..65535].
19	Specify BGP Next-Hop attribute that will be set in the route when advertising (optional).	**esr(config-route-map-rule)# action set ip bgp-next-hop <ADDR>**	<ADDR> – gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
		esr(config-route-map-rule)# action set ipv6 bgp-next-hop <IPV6-ADDR>	<IPV6-ADDR> – gateway IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Step	Description	Command	Keys
20	Specify Next-Hop value that will be set in the route received by BGP (optional).	**esr(config-route-map-rule)# action set ip next-hop {NEXTHOP> | blackhole | unreachable | prohibit}**	<NEXTHOP> – gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]; • **blackhole** – packets to this subnet will be removed without sending notifications to a sender; • **unreachable** – packets to this subnet will be removed, 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);
		esr(config-route-map-rule)# action set ipv6 next-hop <IPV6-NEXTHOP>	<IPV6-ADDR> – gateway IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].
21	Specify BGP Local Preference attribute value that will be set in the route (optional).	**esr(config-route-map-rule)# action set local-preference <PREFERENCE>**	<PREFERENCE> – BGP Local Preference attribute value, takes values in the range of [0..255].
22	Specify BGP Origin attribute value that will be set in the route (optional).	**esr(config-route-map-rule)# action set origin <ORIGIN>**	<ORIGIN> – BGP Origin attribute value: • **egp** – route is learnt by EGP; • **igp** – route is received inside the initial AS; • **incomplete** – route is learnt in another way.
23	Specify BGP MED value that will be set in the route (optional).	**esr(config-route-map-rule)# action set metric bgp <METRIC>**	<METRIC> – BGP MED attribute value, takes values in the range of [0..4294967295].
24	Add filtration and modification of routes in incoming or outgoing directions.	**esr(config-bgp-neighbor)# route-map <NAME><DIRECTION>**	<NAME> – name of the route map having been configured; <DIRECTION> – direction: • **in** – filtration and modification of received routes; • **out** – filtration and modification of advertised routes.
		esr(config-ipv6-bgp-neighbor)# route-map <NAME><DIRECTION>	

11.7.2 Configuration example 1. Route-map for BGP



Objective:

Assign community for routing information coming from AS 20:

First, do the following:

- Configure BGP with AS2500 on ESR router;
- Establish neighboring with AS20.

Solution:

Create a policy:

```
esr# configure
esr(config)# route-map from-as20
```

Create rule 1:

```
esr(config-route-map)# rule 1
```

If AS PATH contains AS20, assign community 20:2020 to it and exit:

```
esr(config-route-map-rule)# match as-path contain 20
esr(config-route-map-rule)# action set community 20:2020
esr(config-route-map-rule)# exit
esr(config-route-map)# exit
```

In AS2500 BGP process, enter neighbor parameter configuration:

```
esr(config)# router bgp 2500
esr(config-bgp)# neighbor 185.0.0.2
esr(config-bgp-neighbor)# address-family ipv4 unicast
```

Map the policy to routing information:

```
esr(config-bgp-neighbor-af)# route-map from-as20 in
```

11.7.3 Configuration example 2. Route-map for BGP

Objective:

For the whole transmitted routing information (from community 2500:25), assign MED equal to 240 and define EGP routing information source:

First:

Configure BGP with AS2500 on ESR.

Solution:

Create a policy:

```
esr(config)# route-map to-as20
```

Create rule:

```
esr(config-route-map)# rule 1
```

If community contains 2500:25, assign MED 240 and Origin EGP to it:

```
esr(config-route-map-rule)# match community 2500:25
esr(config-route-map-rule)# action set metric bgp 240
esr(config-route-map-rule)# action set origin egp
esr(config-route-map-rule)# exit
esr(config-route-map)# exit
```

In AS2500 BGP process, enter neighbor parameter configuration:

```
esr(config)# router bgp 2500
esr(config-bgp)# neighbor 185.0.0.2
esr(config-bgp-neighbor-af)# address-family ipv4 unicast
```

Map the policy to routing information being advertised:

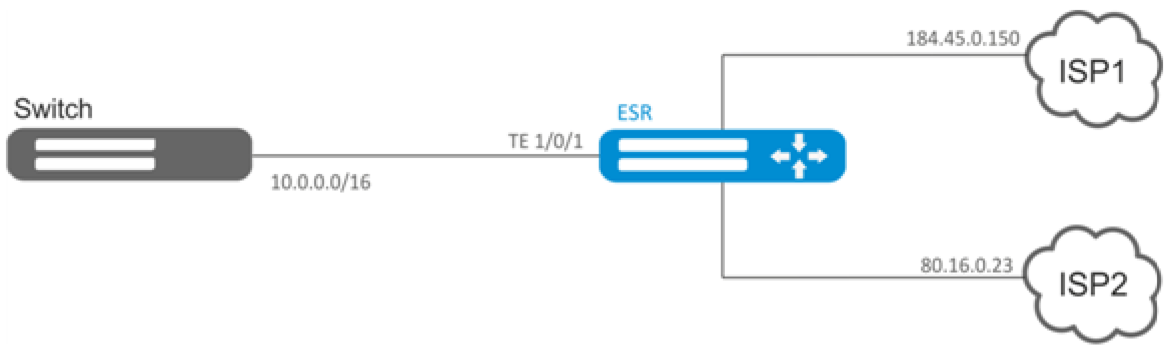
```
esr(config-bgp-neighbor-af)# route-map to-as20 out
esr(config-bgp-neighbor-af)# exit
esr(config-bgp-neighbor)# exit
esr(config-bgp)# exit
```

11.7.4 Route-map based on access control lists (Policy-based routing) configuration algorithm

Step	Description	Command	Keys
1	Create a route map for IP routes filtration and modification.	**esr(config)# route-map <NAME>**	<NAME> – router map name, set by the string of up to 31 characters.

Step	Description	Command	Keys
2	Create a route map rule.	**esr(c onfig-route-map)# rule <ORDER>**	<ORDER> – rule number, takes values of [1..10000].
3	Specify the action that should be applied for routing information.	**esr(config-route-map-rule)# action <ACT>**	<ACT> – allocated action: • **permit** – routing information reception or advertising is permitted; • **deny** – denied.
4	Set ACL for which the rule should work (optional).	**esr(config-route-map-rule)# match ip access-group <NAME>**	<NAME> – access control list name, set by the string of up to 31 characters.
5	Set Next-Hop for the packets that meet the requirements of the specified ACL (optional).	**esr(config-route-map-rule)# action set ip next-hop verify-availability <NEXTHOP><METRIC>**	<NEXTHOP> – gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]; <METRIC> – route metric, takes values of [0..255].
6	Specify ACL-based routing policy.	**esr(config-if-gi)# ip policy route-map <NAME>**	<NAME> – configured routing policy name, set by the string of up to 31 characters.
7	Allow filtration and modification of local traffic based on the routing policy.	**esr(config)# ip local policy [vrf <VRF>] route-map <NAME>**	<NAME> – configured routing policy name, set by the string of up to 31 characters.

11.7.5 Route-map based on access control lists (Policy-based routing) configuration example



Objective:

Distribute traffic between Internet service providers based on user subnets.

First, assign IP address to interfaces.

Route traffic from addresses 10.0.20.0/24 through ISP1 (184.45.0.150), and traffic from addresses 10.0.30.0/24 – through ISP2 (80.16.0.23). Monitor availability of ISP addresses (ISP connection operational capability), and if one the connections goes down, redirect all the traffic from malfunctioning connection to the operational one.

Solution:

Create ACL:

```
esr# configure
esr(config)# ip access-list extended sub20
esr(config-acl)# rule 1
esr(config-acl-rule)# match source-address 10.0.20.0 255.255.255.0
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# match protocol any
esr(config-acl-rule)# action permit
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# exit
esr(config)# ip access-list extended sub30
esr(config-acl)# rule 1
esr(config-acl-rule)# match source-address 10.0.30.0 255.255.255.0
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# match protocol any
esr(config-acl-rule)# action permit
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# exit
```

Create a policy:

```
esr(config)# route-map PBR
```

Create rule 1:

```
esr(config-route-map)# rule 1
```

Specify ACL as a filter:

```
esr(config-route-map-rule)# match ip access-group sub20
```

Specify next-hop for sub20:

```
esr(config-route-map-rule)# action set ip next-hop verify-availability 184.45.0.150 10
esr(config-route-map-rule)# action set ip next-hop verify-availability 80.16.0.23 30
esr(config-route-map-rule)# exit
esr(config-route-map)# exit
```

Rule 1 should provide traffic routing from the network 10.0.20.0/24 to address 184.45.0.150, and in case of its failure, to address 80.16.0.23. Gateway precedence is defined by metrics values – 10 and 30.

Create rule 2:

```
esr(config-route-map)# rule 2
```

Specify ACL as a filter:

```
esr(config-route-map-rule)# match ip access-group sub30
```

Specify nexthop for sub30 and exit:

```
esr(config-route-map-rule)# action set ip next-hop verify-availability 80.16.0.23 10
esr(config-route-map-rule)# action set ip next-hop verify-availability 184.45.0.150 30
esr(config-route-map-rule)# exit
esr(config-route-map)# exit
```

Rule 2 should provide traffic routing from the network 10.0.30.0/24 to address 80.16.0.23, and in case of its failure, to address 184.45.0.150. Precedence is defined by metrics values.

Switch to TE 1/0/1 interface:

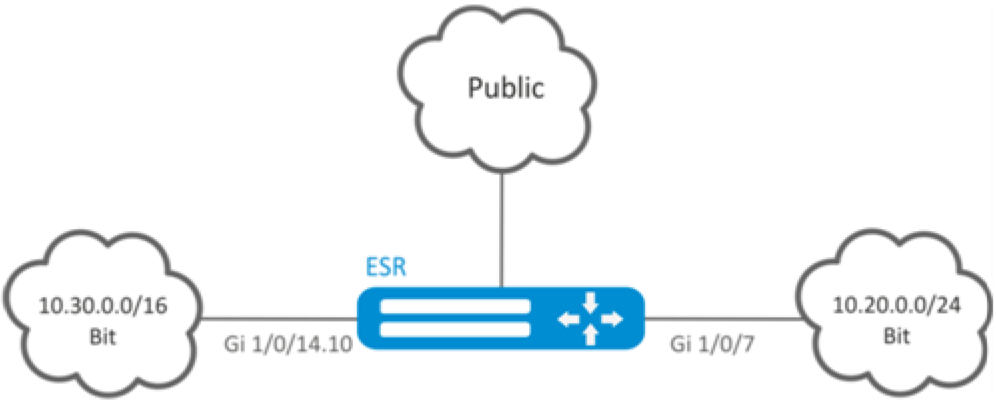
```
esr(config)# interface tengigabitethernet 1/0/1
```

Map the policy the respective interface:

```
esr(config-if-te)# ip policy route-map PBR
```

11.8 VRF configuration

VRF (Virtual Routing and Forwarding) is a technology designed for isolation of routing information that belongs to different classes (e.g., routes of a specific client).



11.8.1 Configuration algorithm

Step	Description	Command	Keys
1	Create VRF instance and switch to the VRF instance parameters configuration mode.	**esr(config)# ip vrf <VRF>**	<VRF> – VRF instance name, set by the string of up to 31 characters.
2	Assign the description of the configured VRF instance.	**esr(config-vrf)# description <DESCRIPTION>**	<DESCRIPTION> – VRF instance description, set by the string of up to 255 characters.

Step	Description	Command	Keys
3	Set the capacity of routing tables in configured VRF for IPv4/IPv6 (optional).	**esr(config-vrf)# ip protocols <PROTOCOL> max-routes <VALUE>** **esr(config-vrf)#ipv6 protocols <PROTOCOL> max-routes <VALUE>**	<PROTOCOL> – protocol type, takes the following values: ospf, bgp; <VALUE> – amount of routes in the routing table, takes values in the range of: • OSPF ESR-1000/1200/1500/1511/1700/3100/3200 – [1..500000], ESR-20/21/30/100/200 – [1..300000], ESR-10/12V(F)/14VF/15 – [1..30000] • BGP ESR-1000/1200/1500/1511/1700/3100/3200 – [1..5000000], ESR-20/21/30/100/200 – [1..2500000], ESR-10/12V(F)/14VF/15 – [1..1000000]. Default value: 0.
4	Enable and configure dynamic traffic routing protocols (Static/OSPF/BGP/IS-IS) in VRF instance (optional). See the related sections: Static routes configuration, OSPF configuration, and BGP configuration.		
5	In the configuration mode of physical/logical interface, tunnel, DNAT/SNAT rule, DAS server or SNMPv3 user, specify the name of VRF instance for which the mode will be used (optional).	**esr(config-snat-ruleset)# ip vrf forwarding <VRF>**	<VRF> – VRF instance name, set by the string of up to 31 characters.
6	Configure LT tunnel to transmit traffic to global mode or to other VRFs (if required).		See LT tunnel configuration section.

11.8.2 Configuration example

Objective:

ESR series router features 2 connected networks that should be isolated from other networks.

Solution:

Create VRF:

```
esr(config)# ip vrf bit
esr(config-vrf)# exit
```

Create a security zone:

```
esr(config)# security zone vrf-sec
esr(config-zone)# ip vrf forwarding bit
esr(config-zone)# exit
```

Create rule for a pair of zones and allow all TCP/UDP traffic:

```
esr(config)# security zone-pair vrf-sec vrf-sec
esr(config-zone-pair)# rule 1
esr(config-zone-rule)# match source-address any
esr(config-zone-rule)# match destination-address any
esr(config-zone-rule)# match protocol udp
esr(config-zone-rule)# match source-port any
esr(config-zone-rule)# match destination-port any
esr(config-zone-rule)# action permit
esr(config-zone-rule)# enable
esr(config-zone-rule)# exit
esr(config-zone-pair)# rule 2
esr(config-zone-rule)# match source-address any
esr(config-zone-rule)# match destination-address any
esr(config-zone-rule)# match protocol tcp
esr(config-zone-rule)# match source-port any
esr(config-zone-rule)# match destination-port any
esr(config-zone-rule)# action permit
esr(config-zone-rule)# enable
esr(config-zone-rule)# exit
```

Create interface mapping, assign IP addresses, specify an inherence to a security zone:

```
esr(config)# interface gigabitethernet 1/0/7
esr(config-if-gi)# ip vrf forwarding bit
esr(config-if-gi)# ip address 10.20.0.1/24
esr(config-if-gi)# security-zone vrf-sec
esr(config-if-gi)# exit
esr(config)# interface gigabitethernet 1/0/14.10
esr(config-subif)# ip vrf forwarding bit
esr(config-subif)# ip address 10.30.0.1/16
esr(config-subif)# security-zone vrf-sec
esr(config-subif)# exit
esr(config)# exit
```

To view information on interfaces mapped to VRF, use the following command:

```
esr# show ip vrf
```

To view VRF routing table, use the following command:

```
esr# show ip route vrf bit
```

11.9 MultiWAN configuration

MultiWAN technology establishes a fail-safe connection with redundancy of links from multiple providers and solves the problem involving traffic balancing between redundant links.

11.9.1 Configuration algorithm

Step	Description	Command	Keys
1	Configure interfaces through which MultiWAN will operate: set IP addresses and specify security zone.		
2	Write static routes through WAN (if required).	**esr(config)# ip route <SUBNET> wan load-balance rule <ID> [<METRIC>]**	<ID> – identifier of the rule being created (see item 2). <METRIC> – route metric, takes values of [0..255].
3	Create WAN rule and switch to the rule parameters configuration mode.	**esr(config)# wan load-balance rule <ID>**	<ID> – identifier of the rule being created, takes values in the range of [1..50].

Step	Description	Command	Keys
4	Specify interfaces or tunnels which are gateways in the route created by MultiWAN service.	**esr(config-wan-rule)# outbound { interface <IF> | tunnel <TUN> } [WEIGHT]**	<IF>– interface name; <TUN> – tunnel name; [WEIGHT] – tunnel or interface weight, defined in the range of [1..255]. If the value is equal 2, than 2 times more traffic will be transmit via the given interface than via the interface with the default value. A route with the highest weight will be active in the redundancy mode. Default value: 1
5	Describe the rules (optional).	**esr(config-wan-rule)# description <DESCRIPTION>**	<DESCRIPTION> – wan rule description, set by the string of up to 255 characters.
6	This command is used to switch from the balancing mode to the redundancy mode (if necessary).	**esr(config-wan-rule)# failover**	
7	Enable wan rule.	**esr(config-wan-rule)# enable**	
8	Create a list of IP addresses to check the connection integrity and perform the switching to the list parameters configuration mode.	**esr(config)# wan load-balance target-list <NAME>**	<NAME> – list name, set by the string of up to 31 characters.
9	Specify the check target and switch to the target parameters configuration mode.	**esr(config-target-list)# target <ID>**	<ID> – target identifier, set in the range of [1..50]. If the 'all' parameter value is used when removing, all targets for the configured target list will be removed.
10	Describe target (optional).	**esr(config-wan-target)# description <DESCRIPTION>**	<DESCRIPTION> – target description, set by the string of up to 255 characters.
11	Specify the standby time via ICMP (optional).	**esr(config-wan-target)# resp-time <TIME>**	<TIME> – timeout, takes value in seconds [1..30].
12	Specify IP address of the check.	**esr(config-wan-target)# ip address <ADDR>**	<ADDR> – destination IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Step	Description	Command	Keys
		esr(config-wan-target)# ipv6 address <IPV6-ADDR>	<IPV6-ADDR> – destination IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].
13	Enable target check.	**esr(config-wan-target)# enable**	
Commands for 14–17 items should be applied on interfaces/tunnels in MultiWAN.			
14	Enable WAN mode on the interface for IPv4/IPv6 stack.	**esr(config-if-gi)# wan load-balance enable** **esr(config-if-gi)# ipv6 wan load-balance enable**	
15	Set the amount of ineffective attempts to check the connection, after which, if there is not response from the opposing side, the connection is considered to be inactive (optional).	**esr(config-if-gi)# wan load-balance failure-count <VALUE>** **esr(config-if-gi)# ipv6 wan load-balance failure-count <VALUE>**	<VALUE> – number of attempts, takes values in the range of [1..10]. Default value: 1.
16	Set the amount of successful attempts to check the connection, after which, if successful, the connection is considered to be active again (optional).	**esr(config-if-gi)# wan load-balance success-count <VALUE>** **esr(config-if-gi)# ipv6 wan load-balance success-count <VALUE>**	<VALUE> – number of attempts, takes values in the range of [1..10]. Default value: 1.
17	Set a neighbor's IP address that will be indicated as one of the gateways in a static route created by MultiWAN service.	**esr(config-if-gi)# wan load-balance nexthop { <IP> | dhcp enable | tunnel enable }**	<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 as nexthop – destination p-t-p address. Applicable for the interfaced being connected that operate via ppp.

Step	Description	Command	Keys
		esr(config-if-gi)# ipv6 wan load-balance nexthop { <IPV6> }	<IPV6> – destination IPv6 address (gateway), defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].
18	This command will be checking the IP addresses from the integrity check list. If all (default)/at least one (using the check-all key) of the tested hosts is unavailable, the gateway is considered unavailable.	**esr(config-if-gi)# wan load-balance target-list { check-all | <NAME> }** **esr(config-if-gi)# ipv6 wan load-balance target-list { check-all | <NAME> }**	<NAME> – run check on the basis of a certain target list (specified in item 7). check-all – run check on the basis of all targets in the list.
19	Write static routes through WAN.	**esr(config)# ip route <SUBNET> wan load-balance rule <ID> [<METRIC>]** **esr(config)# ipv6 route <SUBNET> wan load-balance rule <ID [<METRIC>]**	<ID> – identifier of the rule being created (see item 2). <METRIC> – route metric, takes values of [0..255].

11.9.2 Configuration example

Objective:

Configure route to the server (108.16.0.1/28) with the load balancing option.

Solution:

First, perform the following:

- Configure zones for te1/0/1 and te1/0/2 interfaces;
- Specify IP addresses for te1/0/1 and te1/0/2 interfaces.

Main configuration step:

Configure routing:

```
esr(config)# ip route 108.16.0.0/28 wan load-balance rule 1
```

Create WAN rule:

```
esr(config)# wan load-balance rule 1
```

Specify affected interfaces:

```
esr(config-wan-rule)# outbound interface tengigabitethernet 1/0/2
esr(config-wan-rule)# outbound interface tengigabitethernet 1/0/1
```

Enable the created balancing rule and exit the rule configuration mode:

```
esr(config-wan-rule)# enable
esr(config-wan-rule)# exit
```

Create a list for the connection integrity check:

```
esr(config)# wan load-balance target-list google
```

Create integrity check target:

```
esr(config-target-list)# target 1
```

Specify address to be checked, enable check for the specified address and exit:

```
esr(config-wan-target)# ip address 8.8.8.8
esr(config-wan-target)# enable
esr(config-wan-target)# exit
```

Configure interfaces. In te1/0/1 interface configuration mode, specify nexthop:

```
esr(config)# interface tengigabitethernet 1/0/1
esr(config-if)# wan load-balance nexthop 203.0.0.1
```

In te1/0/1 interface configuration mode, specify a list of targets for connection check:

```
esr(config-if)# wan load-balance target-list google
```

In te1/0/1 interface configuration mode, enable WAN mode and exit:

```
esr(config-if)# wan load-balance enable
esr(config-if)# exit
```

In te1/0/2 interface configuration mode, specify nexthop:

```
esr(config)# interface tengigabitethernet 1/0/2
esr(config-if)# wan load-balance nexthop 65.6.0.1
```

In te1/0/2 interface configuration mode, specify a list of targets for connection check:

```
esr(config-if)# wan load-balance target-list google
```

In te1/0/2 interface configuration mode, enable WAN mode and exit:

```
esr(config-if)# wan load-balance enable
esr(config-if)# exit
```

To switch into redundancy mode, configure the following:

Proceed to WAN rule configuration mode:

```
esr(config)# wan load-balance rule 1
```

MultiWAN function may also work in redundancy mode when traffic is directed to the active interface with the highest weight. To enable this mode, use the following command:

```
esr(config-wan-rule)# failover
```

11.10 IS-IS configuration

IS-IS — ISO standardized dynamic routing protocol based on link-state. It provides fast convergence and excellent scalability, makes economical use of network bandwidth, and uses the Dijkstra Algorithm to calculate the best routes. A distinctive feature of the IS-IS protocol is to work on top of the data link layer of the OSI model, so it is not bound to a specific network layer protocol.

11.10.1 Configuration algorithm

Step	Description	Command	Keys
1	Create an IS-IS process and switch to the parameters configuration mode of this process.	**esr(config)# router isis <ID> [vrf <VRF>]**	<ID> – process number, takes values of [1..65535]; <VRF> – VRF instance name, set by the string of up to 31 characters.
2	Set NET address.	**esr(config-isis)# net {<NET>}**	<NET> – NET address, format: ff[.ffff.ffff.ffff.ffff.ffff.ffff].ffff.ffff.ffff.00.
3	Enable IS-IS process.	**esr(config-isis)# enable**	
4	Set the authentication algorithm for the L2 layer (optional).	**esr(config-isis)# authentication domain algorithm <ALGORITHM>**	<ALGORITHM> – authentication algorithm: • cleartext – unencrypted password; • md5 – password is hashed by md5 algorithm.
5	Set the authentication password for the L2 layer (optional).	**esr(config-isis)# authentication domain key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }**	<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...).

Step	Description	Command	Keys
6	Set a list of keys for authentication (optional).	**esr(config-isis)# authentication domain key chain <KEYCHAIN>**	<KEYCHAIN> – key list identifier, set by the string of up to 16 characters.
7	Select the authentication algorithm for the L1 layer (optional).	**esr(config-isis)# authentication area algorithm <ALGORITHM>**	<ALGORITHM> – authentication algorithm: • cleartext – unencrypted password; • md5 – password is hashed by md5 algorithm.
8	Set the authentication password for the L1 layer (optional).	**esr(config-isis)# authentication area key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }**	<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...).
9	Set a list of keys for authentication (optional).	**esr(config-isis)# authentication area key chain <KEYCHAIN>**	<KEYCHAIN> – key list identifier, set by the string of up to 16 characters.
10	Enable transmission of router name to the LSP (optional).	**esr(config-isis)# hostname dynamic**	
11	Set the IS-IS process operating level (optional).	**esr(config-isis)# is-type {<LEVEL>}**	<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.
12	Set the type of metric to be used in the IS-IS process (optional).	**esr(config-isis)# metric-style { narrow | wide | transition } [<LEVEL>]**	narrow — accepts and generates TLVs (on network reachability) of the old type; wide — accepts and generates TLVs (on network reachability) of the new type; transition — accepts and generates TLVs (on network reachability) of the new and old type; <LEVEL> – IS-IS protocol operation level: • level-1 – operate only on level 1; • level-2-only – operate only on level 2.
13	Set the route priority for the specified IS-IS process (optional).	**esr(config-isis)# preference {<VALUE>}**	<VALUE> – may take values [1..255].

Step	Description	Command	Keys
14	Enable IS-IS operation with IPv4 and/or IPv6 addresses (optional).	**esr(config-isis)# address-family { ipv4 | ipv6 }**	ipv4 – IPv4 family; ipv6 – IPv6 family.
15	Set the update interval for own LSP (optional).	**esr(config-isis)# lsp-refresh-interval { min | max } <TIME> [<LEVEL>]**	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.
16	Set the lifetime of own LSP (optional).	**esr(config-isis)# max-lsp-lifetime <TIME> [<LEVEL>]**	<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.
17	Set a timeout before the next SPF calculation (optional).	**esr(config-isis)# spf-timeout <TIME> [<LEVEL>]**	<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.
18	Enable advertising of routes received in an alternative way (optional).	**esr(config-isis)# redistribute bgp <AS> [route-map <NAME>] [is-type <LEVEL>]** **esr(config-isis)# redistribute ipv6 bgp <AS> [route-map <NAME>] [is-type <LEVEL>]**	<AS> – stand alone system number, takes values of [1..4294967295]. <NAME> – name of the route map that will be used for advertised routes filtration and modification, set by the string of up to 31 characters; <LEVEL> – IS-IS protocol operation level: • level-1 – operate only on level 1; • level-2-only – operate only on level 2.

Step	Description	Command	Keys
		esr(config-isis)# redistribute ospf <ID> <ROUTE-TYPE> [route-map <NAME>] [is-type <LEVEL>]	<ID> – process number, takes values of [1..65535]. <ROUTE-TYPE> – route type: • 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 operation level: • level-1 – operate only on level 1; • level-2-only – operate only on level 2.
		esr(config-isis)# redistribute ipv6 ospf <ID> <ROUTE-TYPE> [route-map <NAME>] [is-type <LEVEL>]	
		esr(config-isis)# redistribute isis <ID> <ROUTE-TYPE> [route-map <NAME>] [is-type <LEVEL>]	<ID> – process number, takes values of [1..65535]. <ROUTE-TYPE> – route type: • level-1 – level 1 routes advertising; • level-2 – level 2 routes advertising; • inter-area – IS-IS process routes advertising between zones; <NAME> – name of the route map that will be used for advertised IS-IS routes filtration and modification, set by the string of up to 31 characters; <LEVEL> – IS-IS protocol operation level: • level-1 – operate only on level 1; • level-2-only – operate only on level 2.
		esr(config-isis)# redistribute rip [route-map <NAME>] [is-type <LEVEL>]	<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 operation level: • level-1 – operate only on level 1; • level-2-only – operate only on level 2.

Step	Description	Command	Keys
		esr(config-isis)# redistribute static [route-map <NAME>] [is-type <LEVEL>]	<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 operation level: • level-1 – operate only on level 1; • level-2-only – operate only on level 2.
		esr(config-isis)# redistribute connected [route-map <NAME>] [is-type <LEVEL>]	<NAME> – name of the route map that will be used for advertised connected routes filtration and modification, set by the string of up to 31 characters; <LEVEL> – IS-IS protocol operation level: • level-1 – operate only on level 1; • level-2-only – operate only on level 2.
19	Add subnets filtration in incoming or outgoing updates (optional).	**esr(config-isis)# prefix-list { ipv6 <LIST_NAME> | <LIST_NAME> } {in|out}**	<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.
20	Add subnets filtration in incoming or outgoing updates (optional).	**esr(config-isis)# route-map <NAME> {in|out}**	<NAME> – name of the route map that will be used for advertised routes filtration and modification, set by the string of up to 31 characters.
21	Set a matching of interface to a specified IS-IS process.	**esr(config-if-gi)# isis instance <ID>**	<ID> – process number, takes values of [1..65535].
22	Enable the IS-IS protocol on the interface.	**esr(config-if-gi)# isis enable**	
23	Enable the use of TLV#8 in hello packets (optional).	**esr(config-if-gi)# isis hello-padding**	
24	Set the priority when selecting DIS (optional).	**esr(config-if-gi)# isis priority <VALUE> [<LEVEL>]**	<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.

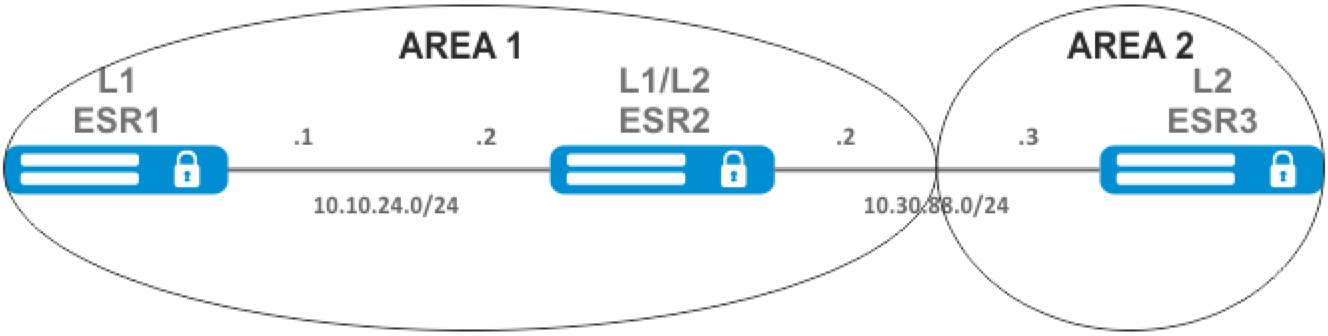
Step	Description	Command	Keys
25	Set the metric value for the interface (optional).	**esr(config-if-gi)# isis metric <VALUE> [<LEVEL>]**	<VALUE> – number, may take values [1..16777215]; <LEVEL> – IS-IS protocol operation level: • level-1 – operate only on level 1; • level-2-only – operate only on level 2.
26	Set defines which routing layer on the interface the current IS-IS process will run on (optional).	**esr(config-if-gi)# isis circuit-type {<LEVEL>}**	<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.
27	Set the interval for sending hello packets (optional).	**esr(config-if-gi)# isis hello-interval <TIME> [<LEVEL>]**	<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.
28	Set the multiplier for calculating and sending Hold Time (optional).	**esr(config-if-gi)# isis hello-multiplier <VALUE> [<LEVEL>]**	<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.
29	Set the interface to point-to-point IS-IS protocol mode (optional).	**esr(config-if-gi)# isis network point-to-point**	
30	Set the interval for generating and sending CSNP (optional).	**esr(config-if-gi)# isis csnp-interval <TIME> [<LEVEL>]**	<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.
31	Set the interval for generating and sending PSNP (optional).	**esr(config-if-gi)# isis psnp-interval <TIME> [<LEVEL>]**	<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.

Step	Description	Command	Keys
32	Set the interval between LSP transmissions on the Broadcast network (optional).	**esr(config-if-gi)# isis lsp-interval <TIME> [<LEVEL>]**	<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.
33	Set the LSP re-distribution interval in the PtP network (optional).	**esr(config-if-gi)# isis lsp-retransmit-interval <TIME> [<LEVEL>]**	<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.
34	Set the authentication algorithm for the hello packets (optional).	**esr(config-if-gi)# isis authentication algorithm <ALGORITHM> [<LEVEL>]**	<ALGORITHM> – authentication algorithm: • cleartext – unencrypted password; • 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.
35	Set the password for hello packet authentication (optional).	**esr(config-if-gi)# isis authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> } [<LEVEL>]**	<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 ...); <LEVEL> – IS-IS protocol operation level: • level-1 – operate only on level 1; • level-2-only – operate only on level 2.
36	Set the key list for hello packet authentication (optional).	**esr(config-if-gi)# isis authentication key chain <KEYCHAIN> [<LEVEL>]**	<KEYCHAIN> – key list identifier, set by the string of up to 16 characters; <LEVEL> – IS-IS protocol operation level: • level-1 – operate only on level 1; • level-2-only – operate only on level 2.

11.10.2 Configuration example

Objective:

Configure the IS-IS protocol on routers to exchange routing information with neighbors. Router ESR1 will be L1-only, ESR2 will be L1/L2, ESR3 will be L2-only, which will also be in another area.



Solution:

Pre-configure IP addresses on interfaces according to the network structure shown in figure above.

Proceed to the ESR1 router configuration. Create IS-IS process with identifier 1 and proceed to the protocol configuration mode:

```
ESR1(config)# router isis 1
```

Set the number of the zone in which the router will operate and its system ID:

```
ESR1(config-isis)# net 49.0001.1111.1111.1111.00
```

Configure the router to operate only on the first layer of the IS-IS protocol:

```
ESR1(config-isis)# is-type level-1
```

Set the operation of the router with a narrow metric on the first level:

```
ESR1(config-isis)# metric-style narrow level-1
```

Enable the IS-IS process on the router:

```
ESR1(config-isis)# enable
```

Proceed to the interface configuration. It is necessary to set the number of the IS-IS process which will run on the interface and to enable the protocol itself to run on it:

```
ESR1(config-if-gi)# isis instance 1
ESR1(config-if-gi)# isis enable
```

Proceed to the ESR2 router configuration.

```
ESR2(config)# router isis 2
```

Set the zone number, the same as on ESR1, as well as a unique system identifier:

```
ESR2(config-isis)# net 49.0001.2222.2222.2222.00
```

Set the router to operate with a narrow metric on the first layer and with a wide metric on the second layer, and enable this IS-IS process:

```
ESR2(config-isis)# metric-style narrow level-1
ESR2(config-isis)# metric-style wide level-2
ESR2(config-isis)# enable
```

Configure the interfaces on the router. The configuration will be the same on both interfaces.

```
ESR2(config-if-gi)# isis instance 2
ESR2(config-if-gi)# isis enable
```

Proceed to the ESR3 router configuration.

```
ESR3(config)# router isis 3
ESR3(config-isis)# net 49.0002.3333.3333.3333.00
ESR3(config-isis)# is-type level-2
ESR3(config-isis)# metric-style wide level-2
ESR3(config-isis)# enable
ESR3(config-if-gi)# isis instance 3
ESR3(config-if-gi)# isis enable
```

The neighborhood establishment can be viewed with the show isis neighbors command. Execute it on ESR2:

```
ESR2# show isis neighbors
IS-IS 2
IS-IS Level 1 Neighbors
System ID                Hostname          Interface         State         Holdtime  SNPA
1111.1111.1111  ESR1                      gi1/0/2           Up               25
a8f9.4baa.1d42
IS-IS Level 2 Neighbors
System ID               Hostname          Interface         State         Holdtime  SNPA
3333.3333.3333  ESR3                     gi1/0/1           Up               8
a8f9.4bab.813a
```

12 MPLS technology management

12.1 LDP configuration

LDP is a tag distribution protocol. To find the neighbors hello messages are sent to the multicast address 224.0.0.2. When exchanging hello messages, routers learn each other's transport addresses. A router with a bigger address initializes the TCP session. After checking the parameters, the LDP session is considered established.

ESR routers support the following LDP operation modes:

- Tag information exchange mode — Downstream Unsolicited;
- Mechanism for controlling the distribution of tags — Independent Label Distribution Control;
- Label retention mode — Liberal Label Retention;

On interfaces where LDP and MPLS switching are enabled, the firewall must be disabled.

The current version LDP only works with IPv4 addresses.

12.1.1 Configuration algorithm

Step	Description	Command	Keys
1	In the context of MPLS parameters configuration, specify the interfaces involved in the MPLS switching process	**esr(config-mpls)# forwarding interface { <IF> | <TUN> }**	<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;
2	Specify the router-id for LDP (not necessary if transport-address is specified).	**esr(config-ldp)# router-id <ID>**	<ID> – router identifier, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
3	In the context of the address family ipv4 settings, specify transport-address (not necessary if router-id is specified).	**esr(config-ldp-af-ipv4)# transport-address <ADDR>**	<ADDR> – defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
4	In the context of the address family ipv4 settings, specify interfaces for enabling LDP process.	**esr(config-ldp-af-ipv4)# interface { <IF> | <TUN> }**	<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.

Step	Description	Command	Keys
5	Enable LDP process.	**esr(config-ldp)# enable**	
6	Enable explicit-null functionality (optional).	**esr(config-ldp)# egress-label-type explicit-null**	
7	In the LDP neighbor configuration mode, set the password with the password command (optional).	**esr(config-ldp-neig)# password {<TEXT> | ENCRYPTED-TEXT>}**	<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...).

The following functionality is also available as part of the LDP configuration:

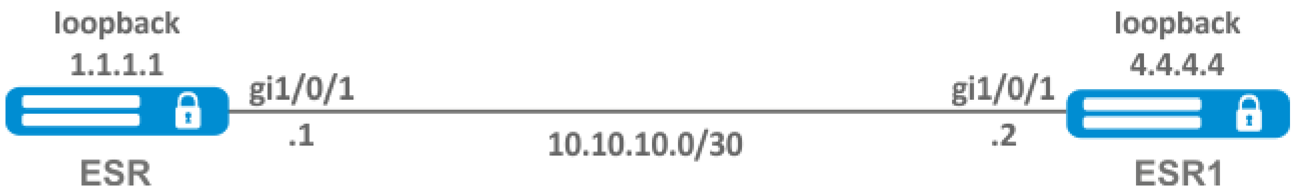
- LDP tag filtering configuration (see section LDP tag filtering configuration);
- LDP session parameters configuration (see section Configuring session parameters in LDP);
- t-LDP session parameters configuration (see section Configuring session parameters in targeted-LDP).

If router-id value is changed, then the new value will be applied only after restarting this protocol. To restart mpls ldp, use the clear mpls ldp command.

12.1.2 Configuration example

Objective:

Configure LDP communication between peers.



Solution:

ESR pre-configuration:

First, IP addresses must be assigned to the interfaces, the firewall must be disabled and one of the internal routing protocols must be configured.

ESR pre-configuration:

```
hostname ESR
router ospf 1
  area 0.0.0.0
    enable
  exit
  enable
exit


interface gigabitethernet 1/0/1
  ip firewall disable
  ip address 10.10.10.1/30
  ip ospf instance 1
  ip ospf
exit


interface loopback 1
  ip address 1.1.1.1/32
  ip ospf instance 1
  ip ospf
exit
```

ESR1 pre-configuration:

```
hostname ESR1
router ospf 1
  area 0.0.0.0
    enable
  exit
  enable
exit


interface gigabitethernet 1/0/1
  ip firewall disable
  ip address 10.10.10.2/30
  ip ospf instance 1
  ip ospf
exit


interface loopback 1
  ip address 4.4.4.4/32
  ip ospf instance 1
  ip ospf
exit
```

Configuration on ESR:

ESR

```
ESR# config
ESR(config)# mpls
ESR(config-mpls)# forwarding interface gigabitethernet 1/0/1
ESR(config-mpls)# ldp
ESR(config-ldp)# router-id 1.1.1.1
ESR(config-ldp)# enable
ESR(config-ldp)# address-family ipv4
ESR(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
ESR(config-ldp-af-ipv4-if)# end
ESR#
```

Configuration on ESR1:

ESR1

```
ESR1# configure
ESR1(config)# mpls
ESR1(config-mpls)# forwarding interface gigabitethernet 1/0/1
ESR1(config-mpls)# ldp
ESR1(config-ldp)# router-id 4.4.4.4
ESR1(config-ldp)# enable
ESR1(config-ldp)# address-family ipv4
ESR1(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
ESR1(config-ldp-af-ipv4-if)# end
ESR1#
```

Check:

Enter the following commands at one of the peers:

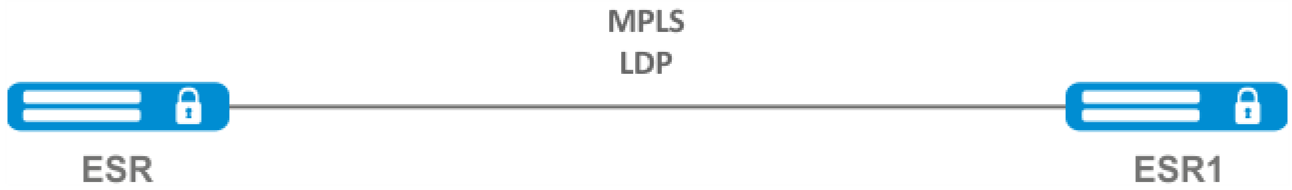
```
ESR# show mpls ldp discovery detailed
Local LDP ID: 1.1.1.1
Discovery sources:
    Interfaces:
        gigabitethernet 1/0/1:
                 Hello interval: 5 seconds
        Transport IP address: 1.1.1.1
        LDP ID: 4.4.4.4
            Source IP address: 10.10.10.2
            Transport IP address: 4.4.4.4
                          Hold time: 15 seconds
                          Proposed hold time: 90/15 (local/peer) seconds
```

The output will show the parameters of the neighboring peer obtained from the multicast hello messages.

The LDP session should be in the 'Operational' state.

```
ESR1# show mpls ldp neighbor
Peer LDP ID: 4.4.4.4; Local LDP ID 1.1.1.1
    State: Operational
    TCP connection: 4.4.4.4:40245 - 1.1.1.1:646
    Messages sent/received: 10/11
    Uptime: 00:00:58
    LDP discovery sources:
        gigabitethernet 1/0/1
```

12.2 Configuring session parameters in LDP



By default, hello messages sent out are set to the following values:

Parameter	LDP
Hello interval	5 seconds
Hold timer	15 seconds
Keepalive holdtime	180 seconds

Hold timer is a matching parameter — the smallest is chosen. This example shows that the ESR after matching the Hold timer is 10 seconds.

```
ESR# sh mpls ldp discovery detailed
Local LDP ID: 4.4.4.4
Discovery sources:
    Interfaces:
        gigabitethernet 1/0/4:
            Hello interval: 5 seconds
            Transport IP address: 4.4.4.4
            LDP ID: 1.1.1.1
                Source IP address: 10.10.10.1
                Transport IP address: 1.1.1.1
                                   Hold time: 10 seconds
                                   Proposed hold time: 15/10 (local/peer) seconds
```

If after matching, the Hello interval is greater than the Hold timer, then the Hello interval will be equal to Hold timer/3.

ESR routers have the ability to flexibly configure Hello holdtime, Hello interval and Keepalive holdtime settings. Example of configuring Hello holdtime for an LDP session:

```
ESR# show run mpls
mpls
  ldp
  router-id 4.4.4.4
     discovery hello holdtime 40
    address-family ipv4
      interface gigabitethernet 1/0/4
             discovery hello holdtime 60
    exit
  exit
  enable
exit
```

If the Hello Holdtime and Hello Interval parameters are not specified, the default values are used. If parameters are specified, the priority of values for address-family will be higher than for globally configured values.

```
ESR# show mpls ldp discovery detailed
Local LDP ID: 4.4.4.4
Discovery sources:
    Interfaces:
        gigabitethernet 1/0/4:
            Hello interval: 5 seconds
            Transport IP address: 4.4.4.4
            LDP ID: 1.1.1.1
                Source IP address: 10.10.10.1
                Transport IP address: 1.1.1.1
                Hold time: 15 seconds
                                   Proposed hold time: 60 /15 (local/peer) seconds
```

The parameters configured in address-family can be configured for each individual interface participating in the LDP process.

```
ESR# show running-config mpls
mpls
  ldp
    router-id 4.4.4.4
    discovery hello holdtime 50
    discovery hello interval 10
    address-family ipv4
             interface gigabitethernet 1/0/1
                 discovery hello holdtime 60
                 discovery hello interval 20
      exit
             interface gigabitethernet 1/0/4
                 discovery hello holdtime 30
                 discovery hello interval 10
      exit
    exit
  enable
exit
```

For a TCP session, Keepalive holdtime is also a matching parameter similar to Hold timer. Keepalive interval is calculated automatically and equals Keepalive holdtime/3. Keepalive holdtime can be set globally as well as for each neighbor. The timer set for a particular neighbor is a higher priority.

```
ESR# show running-config mpls
mpls
  ldp
  router-id 4.4.4.4
     keepalive 30 // set in the global LDP configuration
  neighbor 1.1.1.1
     keepalive 55// set to neighbor with the 1.1.1.1 address
  exit
exit
```

```
ESR# sh mpls ldp neighbor 1.1.1.1
Peer LDP ID: 1.1.1.1; Local LDP ID 4.4.4.4
    State: Operational
    TCP connection: 1.1.1.1:646 - 4.4.4.4:56668
    Messages sent/received: 401/401
    Uptime: 02:00:24
    Peer holdtime: 55
    Keepalive interval: 18
    LDP discovery sources:
```

12.2.1 Algorithm for setting Hello holdtime and Hello interval in the global LDP configuration

Step	Description	Command	Keys
1	Configure the LDP (see section LDP configuration).		
2	In the LDP configuration mode, set Hello holdtime.	**esr(config-ldp)# discovery hello holdtime <TIME>**	<TIME> — Time in seconds in the range of [3..65535]. Default value: 15.
3	In the LDP configuration mode, set Hello interval.	**esr(config-ldp)# discovery hello interval <TIME>**	<TIME> — Time in seconds in the range of [3..65535]. Default value: 5.

12.2.2 Algorithm for setting Hello holdtime and Hello interval for address family

Step	Description	Command	Keys
1	Configure the LDP (see section LDP configuration).		
2	In the LDP address family configuration mode, set Hello holdtime on the specified interface.	**esr(config-ldp-af-ipv4-if)# discovery hello holdtime <TIME>**	<TIME> — time in the range of [3..65535] seconds. Default value: 15.

Step	Description	Command	Keys
3	In the LDP address family configuration mode, set Hello interval on the specified interface.	**esr(config-ldp-af-ipv4-if)# discovery hello interval <TIME>**	<TIME> — time in the range of [3..65535] seconds. Default value: 5.

12.2.3 Algorithm for setting Keepalive holdtime parameter in the global LDP configuration

Step	Description	Command	Keys
1	Configure the LDP (see section LDP configuration).		
2	In the LDP configuration mode, set the Keepalive parameter.	**esr(config-ldp)# keepalive <TIME>**	<TIME> — time in the range of [3..65535] seconds. Default value: 180.

12.2.4 Algorithm for setting Keepalive holdtime parameter for the specific neighbor

Step	Description	Command	Keys
1	Configure the LDP (see section LDP configuration).		
2	In the neighbor configuration mode, set the Keepalive holdtime parameter.	**esr(config-ldp-neig)# keepalive <TIME>**	<TIME> — time in the range of [3..65535] seconds. Default value: 180.

12.2.5 Configuration example

Objective:

Override Hello holdtime (40 seconds) and Hello interval (10 seconds) parameters for the entire LDP process. For the neighbor with address 1.1.1.1 set the Keepalive holdtime to 150 seconds.

Solution:

ESR

```
ESR(config)# mpls
ESR(config-mpls)# ldp
ESR(config-ldp)# discovery hello holdtime 40
ESR(config-ldp)# discovery hello interval 10
ESR(config-ldp)# neighbor 1.1.1.1
ESR(config-ldp-neig)# keepalive 150
```

Check:

To view hello parameters:

ESR

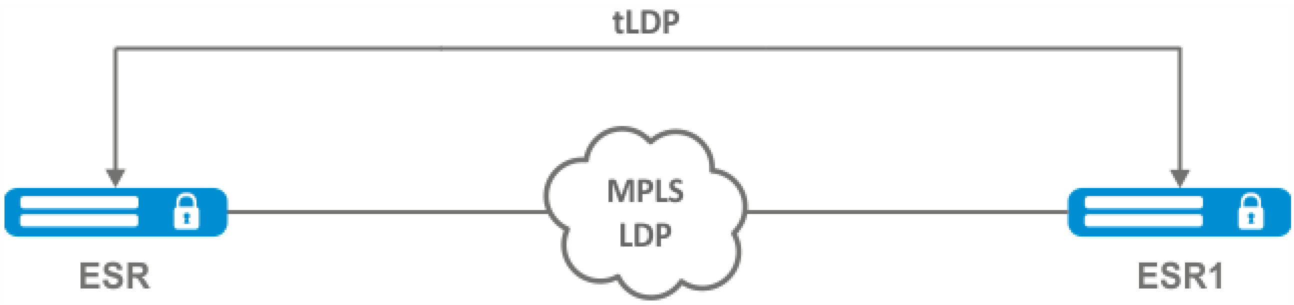
```
ESR# sh mpls ldp discovery detailed
Local LDP ID: 4.4.4.4
Discovery sources:
    Interfaces:
        gigabitethernet 1/0/4:
            Hello interval:       10 seconds
            Transport IP address: 4.4.4.4
            LDP ID:               1.1.1.1
                Source IP address:    10.10.10.1
                Transport IP address: 1.1.1.1
                Hold time:            15 seconds
                Proposed hold time:   40/15 (local/peer) seconds
```

To view parameter of the established TCP session:

ESR

```
ESR# sh mpls ldp neighbor 1.1.1.1
Peer LDP ID: 1.1.1.1; Local LDP ID 4.4.4.4
    State:                   Operational
    TCP connection:          1.1.1.1:646 - 4.4.4.4:45414
    Messages sent/received: 15/15
    Uptime:                  00:06:31
    Peer holdtime:           150
    Keepalive interval:      50
    LDP discovery sources:
```

12.3 Configuring session parameters in targeted-LDP



By default, the targeted LDP session is set to the following values:

Parameter	**targeted-LDP**
Hello interval	5 seconds
Hold timer	45 seconds
Keepalive holdtime	180 seconds

Hold timer is a matching parameter — the smallest is chosen. This example shows that the ESR after matching set 30 seconds:

```
ESR1# sh mpls ldp discovery detailed

...
Targeted hellos:
1.1.1.1 -> 4.4.4.4:
Hello interval: 2 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: 30 seconds
Proposed hold time: 30/45 (local/peer) seconds
```

If after matching, the Hello interval is greater than the Hold timer, then the Hello interval will be equal to Hold timer/3.

ESR routers have the possibility to flexibly configure Hello holdtime, Hello interval and Keepalive holdtime parameters: the parameters can be set for the entire LDP process, as well as for the corresponding neighbor.

Example output for the LDP process:

```
ESR# sh running-config mpls
mpls
  ldp
    router-id 1.1.1.1
         keepalive 160
         discovery targeted-hello holdtime 30
         discovery targeted-hello interval 10
  exit
exit
```

Example output for a targeted-LDP session for a particular neighbor:

```
ESR# sh running-config mpls
mpls
  ldp
    router-id 1.1.1.1
    neighbor 4.4.4.4
      keepalive 160
      targeted
      discovery targeted-hello holdtime 30
      discovery targeted-hello interval 45
    exit
  exit
exit
```

If parameters are set for both the LDP process and a specific neighbor, the priority will be the settings set for the neighbor.

```
ESR# sh running-config mpls
mpls
  ldp
    router-id 1.1.1.1
    keepalive  160
    discovery hello holdtime 90
    discovery targeted-hello interval 30
    neighbor 4.4.4.4
      keepalive 140
      targeted
      discovery targeted-hello holdtime 45
      discovery targeted-hello interval 15
    exit
  exit
exit
```

```
ESR# show mpls ldp discovery detailed
...
Targeted hellos:
1.1.1.1 -> 4.4.4.4:
Hello interval: 15 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


ESR# show mpls ldp neighbor 4.4.4.4
Peer LDP ID: 4.4.4.4; Local LDP ID 1.1.1.1
     State: Operational
      TCP connection: 4.4.4.4:51861 - 1.1.1.1:646
      Messages sent/received: 10/10
      Uptime: 00:00:09
      Peer holdtime: 140
      Keepalive interval: 46
      LDP discovery sources:
            1.1.1.1 -> 4.4.4.4:
```

12.3.1 Algorithm for setting Hello holdtime, Hello interval and Keepalive holdtime for the LDP process

1	Configure the LDP (see section LDP configuration).		
2	In the LDP configuration mode, set Hello holdtime.	**esr(config-ldp)# discovery targeted-hello holdtime <TIME>**	<TIME> — time in the range of [3..65535] seconds. Default value: 45.
3	In the LDP configuration mode, set Hello interval.	**esr(config-ldp)# discovery targeted- hello interval <TIME>**	<TIME> — time in the range of [1..65535] seconds. Default value: 5.
4	In the LDP configuration mode, set Keepalive holdtime.	**esr(config-ldp)# keepalive <TIME>**	<TIME> — time in the range of [3..65535] seconds. Default value: 180.

12.3.2 Algorithm for setting Hello holdtime, Hello interval and Keepalive holdtime for the specific neighbor

1	Configure the LDP (see section LDP configuration).		
2	In the LDP neighbor configuration mode, set Hello holdtime.	**esr(config-ldp-neig)# discovery targeted-hello holdtime <TIME>**	<TIME> — time in the range of [3..65535] seconds. Default value: 45.

3	In the LDP neighbor configuration mode, set Hello interval.	**esr(config-ldp-neig)# discovery targeted- hello interval <TIME>**	<TIME> — time in the range of [1..65535] seconds. Default value: 5.
4	In the LDP neighbor configuration mode, set Keepalive holdtime.	**esr(config-ldp-neig)# keepalive <TIME>**	<TIME> — time in the range of [3..65535] seconds. Default value: 180.

12.3.3 Configuration example

Objective:

Override Hello holdtime (120 seconds) and Hello interval (30 seconds) parameters for the entire targeted-LDP process. For the neighbor with address 4.4.4.4 set the Keepalive holdtime to 150 seconds.

Solution:

ESR

```
ESR(config)# mpls
ESR(config-mpls)# ldp
ESR(config-ldp)# discovery targeted-hello holdtime 40
ESR(config-ldp)# discovery targeted-hello interval 10
ESR(config-ldp)# neighbor 4.4.4.4
ESR(config-ldp-neig)# keepalive 150
```

Check:

To view hello parameters of the targeted LDP session:

ESR

```
ESR1# sh mpls ldp discovery detailed
...
    Targeted hellos:
         1.1.1.1 -> 4.4.4.4:
            Hello interval:       10 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:            40 seconds
                Proposed hold time:   40/45 (local/peer) seconds
```

To view parameter of the established TCP session:

ESR

```
ESR# sh mpls ldp neighbor 4.4.4.4
Peer LDP ID: 4.4.4.4; Local LDP ID 1.1.1.1
    State:                 Operational
    TCP connection:        4.4.4.4:34879 - 1.1.1.1:646
    Messages sent/received: 11/11
    Uptime:                00:01:05
    Peer holdtime:         150
    Keepalive interval:    50
    LDP discovery sources:
         1.1.1.1 -> 4.4.4.4:
            Hello interval: 10 seconds
            Holdtime:       40 seconds
  ...
```

12.4 LDP tag filtering configuration

By default, routers allocate a separate label to each FEC. There are scenarios when it is necessary to allocate MPLS tags only for certain FECs.

12.4.1 Configuration algorithm

Step	Description	Command	Keys
1	Configure the LDP (see section LDP configuration).		
2	Create network type object-group.	**esr(config)# object-group network <NAME>**	<NAME> – name of a subnet list being configured, set by the string of up to 31 characters.
3	Describe the subnets for which labels will be assigned.	**esr(config-object-group-network)# ip prefix <ADDR/LEN>**	<ADDR/LEN> – IP address and subnet mask, defined as AAA.BBB.CCC.DDD/EE where each part AAA-DDD takes values of [0..255] and EE takes values of [1..32].
4	In the context of the LDP configuration, apply the created object-group.	**esr(config-ldp)# advertise-labels <NAME>**	<NAME> – name of a subnet list being configured, set by the string of up to 31 characters.

⚠ **Tags will be allocated ONLY to the subnets described in the object-group, regardless of how they were learned (connected, local, IGP, etc.).**

⚠ **Prefixes must be described in the object-group.**
The prefix must have an exact match with the route from the FIB.

This functionality is supported for IPv4.

12.4.2 Configuration example

Objective:

Assign MPLS tags only to FEC 10.10.0.2/32 and 10.10.0.1/32.

Solution:

On ESR_A and ESR_B create an object-group ADV_LABELS type network and add to it the prefixes 10.10.0.1/32 and 10.10.0.2/32 respectively.

ESR_A

```
esr(config)# object-group network ADV_LABELS
esr(config-object-group-network)# ip prefix 10.10.0.1/32
esr(config-object-group-network)# ip prefix 10.10.0.2/32
```

ESR_B

```
esr(config)# object-group network ADV_LABELS
esr(config-object-group-network)# ip prefix 10.10.0.1/32
esr(config-object-group-network)# ip prefix 10.10.0.2/32
```

Apply the created object-group on both routers:

ESR_A и ESR_B

```
esr(config)# mpls
esr(config-ldp)# ldp
esr(config-ldp)# advertise-labels ADV_LABELS
```

Check:

On ESR_B make sure that the tag is assigned to the appropriate prefixes:

```
esr# sh mpls ldp bindings 10.10.0.1/32
10.10.0.1/32
local label: exp-null
remote label: 75 lsr: 172.16.0.1
```

And not assigned to 192.168.2.0/24:

```
esr# sh mpls ldp bindings 192.168.2.0/24
esr#
```

12.5 L2VPN Martini mode configuration

L2VPN allows organizing ethernet frames transmission through the MPLS domain. Allocation and distribution of tunnel labels, in this mode, is carried out by means of the LDP. In the implementation of L2VPN can be divided into two cases:

1. P2P — 'point-to-point' tunnel;
2. VPLS — 'point-to-multipoint' tunnel.

In both cases, a virtual channel (pseudo-wire) is created to transmit ethernet frames between routers. To negotiate pseudo-wire parameters, as well as to allocate and transfer tunnel labels between routers, an LDP session is established in the targeted mode.

12.5.1 L2VPN VPWS configuration algorithm

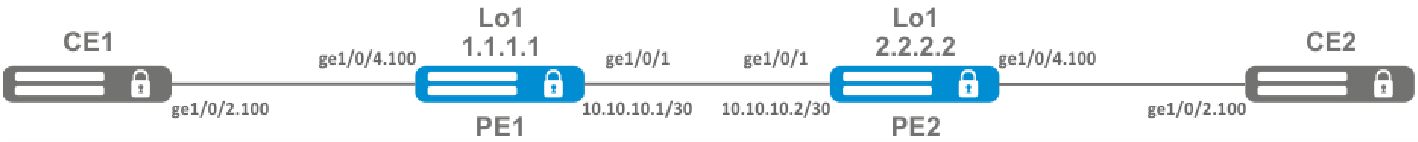
Step	Description	Command	Keys
1	Configure the LDP (see section LDP configuration).		
2	Create pw-class in the system and switch to the pw-class configuration mode.	**esr(config-l2vpn)# pw-class <WORD>**	<WORD> — pw-class name [1..31] characters long.
3	Add a description for pw-class (optional).	**esr(config-l2vpn-pw-class)# description <LINE>**	<LINE> – description. Set by the string [1..255] characters long.
4	Set the MTU value for the pseudo-wire included in the pw-class (optional).	**esr(config-l2vpn-pw-class)# encapsulation mpls mtu <MTU>**	<MTU> — MTU value, takes values in the range of [552..10000] Default value: 1500.
5	Disable status-tlv messaging (optional).	**esr(config-l2vpn-pw-class)# encapsulation mpls status-tlv disable**	Default value: status-tlv enable
6	Create p2p-class in the system and switch to the p2p-class configuration mode.	**esr(config-l2vpn)# p2p <NAME>**	<NAME> — name of the p2p service, set by the string of up to 31 characters.

Step	Description	Command	Keys
7	Specify Attached Circuit interface.	**esr(config-l2vpn-p2p)# interface { <IF> | <TUN> }**	<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.
8	Enable p2p tunnel.	**esr(config-l2vpn-p2p)# enable**	
9	Specify transport mode (optional).	**esr(config-l2vpn-p2p)# transport-mode { ethenet | vlan }**	<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.
10	Create a pseudo-wire and switch to its parameters configuration mode.	**esr(config-l2vpn-p2p)# pw <PW_ID> <LSR_ID>**	<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].
11	Add a description for pseudo-wire (optional).	**esr(config-l2vpn-pw)# description <LINE>**	<LINE> – description. Set by the string [1..255] characters long.
12	Set pw-class for pseudo-wire.	**esr(config-l2vpn-pw)# pw-class <WORD>**	<WORD> — pw-class name [1..31] characters long.
13	Set the LSR address to which the pseudo-wire is set (optional if the neighbor address is the same as the LSR_ID).	**esr(config-l2vpn-pw)# neighbor-address <ADDR>**	<ADDR> – router IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
14	Enable pseudo-wire.	**esr(config-l2vpn-pw)# enable**	
	If it is necessary to change the default settings for a targeted LDP session, see section Configuring session parameters in targeted-LDP.		

12.5.2 L2VPN VPWS configuration example

Objective:

Configure l2vpn so that ge1/0/2.100 interface of the CE1 router and ge1/0/2.100 interface of the CE2 router operate within the same broadcast domain.



Solution:

Pre-requisite:

- Enable Jumbo frames support with the 'system jumbo-frames' command (the device must be rebooted for the changes to take effect);
- Configure IP addresses on interfaces according to the network structure shown in the figure above;
- Organize the exchange of routes between PE1 and PE2 using IGP (OSPF, IS-IS, RIP).

On the PE1 router create a sub-interface from which traffic from CE1 will be received:

```
PE1# configure
PE1(config)# interface gigabitethernet 1/0/4.100
PE1(config-subif)# exit
```

Set the MTU value on the interface towards PE2 to 9600 to avoid MTU overrun after encapsulating the MPLS header and disable the firewall:

```
PE1#(config)# interface gigabitethernet 1/0/1
PE1(config-if-gi)# mtu 9600
PE1(config-if-gi)# ip firewall disable
PE1(config-if-gi)# exit
```

Allow packets with an MPLS header to be received on the interface towards the MPLS network (in this example, the interface towards PE2):

```
PE1(config)# mpls
PE1(config-mpls)# forwarding interface gigabitethernet 1/0/1
```

Configure the LDP protocol and enable neighbor detection on the interface towards PE2:

```
PE1(config-mpls)# ldp
PE1(config-ldp)# router-id 1.1.1.1
PE1(config-ldp)# address-family ipv4
PE1(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
PE1(config-ldp-af-ipv4-if)# exit
PE1(config-ldp-af-ipv4)# transport-address 1.1.1.1
PE1(config-ldp-af-ipv4)# exit
PE1(config-ldp)# enable
PE1(config-ldp)# exit
```

Create a pw-class on the basis of which the virtual channel (pw) will be created later. Since, in this example, the default parameters will be applied to pw, it will be sufficient to specify the class name:

```
PE1(config-mpls)# l2vpn
PE1(config-l2vpn)# pw-class for_p2p_VLAN100
PE1(config-l2vpn-pw-class)# exit
```

Create a new l2vpn of type p2p and add pw to router PE3, take the pw identifier as VID for convenience (in this case equal to 100):

```
PE1(config-l2vpn)# p2p to_PE2_VLAN100
PE1(config-l2vpn-p2p)# interface gigabitethernet 1/0/4.100
PE1(config-l2vpn-p2p)# pw 100 3.3.3.3
PE1(config-l2vpn-pw)# pw-class for_p2p_VLAN100
PE1(config-l2vpn-pw)# enable
PE1(config-l2vpn-pw)# exit
PE1(config-l2vpn-p2p)# enable
PE1(config-l2vpn-p2p)# end
```

Apply the configuration:

```
PE1# commit
PE1# confirm
```

Configure the PE2 router in the same way as PE1:

```
PE2# configure
PE2(config)# interface gigabitethernet 1/0/4.100
PE2(config-subif)# exit
PE2#(config)# interface gigabitethernet 1/0/1
PE2(config-if-gi)# mtu 9600
PE1(config-if-gi)# ip firewall disable
PE1(config-if-gi)# exit
PE2(config)# mpls
PE2(config-mpls)# forwarding interface gigabitethernet 1/0/1
PE2(config-mpls)# ldp
PE2(config-ldp)# router-id 2.2.2.2
PE2(config-ldp)# address-family ipv4
PE2(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
PE2(config-ldp-af-ipv4-if)# exit
PE2(config-ldp-af-ipv4)# transport-address 2.2.2.2
PE2(config-ldp-af-ipv4)# exit
PE2(config-ldp)# enable
PE2(config-ldp)# exit
PE2(config-mpls)# l2vpn
PE2(config-l2vpn)# pw-class for_p2p_VLAN100
PE2(config-l2vpn-pw-class)# exit
PE2(config-l2vpn)# p2p to_PE1_VLAN100
PE2(config-l2vpn-p2p)# interface gigabitethernet 1/0/4.100
PE2(config-l2vpn-p2p)# pw 100 1.1.1.1
PE2(config-l2vpn-pw)# pw-class for_p2p_VLAN100
PE2(config-l2vpn-pw)# enable
PE2(config-l2vpn-pw)# exit
PE2(config-l2vpn-p2p)# enable
PE2(config-l2vpn-p2p)# end
PE2# commit
PE2# confirm
```

Make sure that the LDP neighborhood is established and display the virtual channel status (pseudowire) between PE1 and PE2:

```
PE2# show mpls ldp neighbor
Peer LDP ID: 1.1.1.1; Local LDP ID 2.2.2.2
    State: Operational
    TCP connection: 1.1.1.1:646 - 2.2.2.2:34625
    Messages sent/received: 12/12
    Uptime: 00:03:50
    LDP discovery sources:
        2.2.2.2 -> 1.1.1.1
```

```
PE2# show mpls l2vpn pseudowire
Neighbor                                  PW ID         Type         Status
---------------------------------------- ----------    ----------    ------
1.1.1.1                                    100         Ethernet       Up
```

The LDP neighborhood is established, pseudowire has moved to 'UP' status. The l2vpn p2p type configuration is now complete.

12.5.3 L2VPN VPLS configuration algorithm

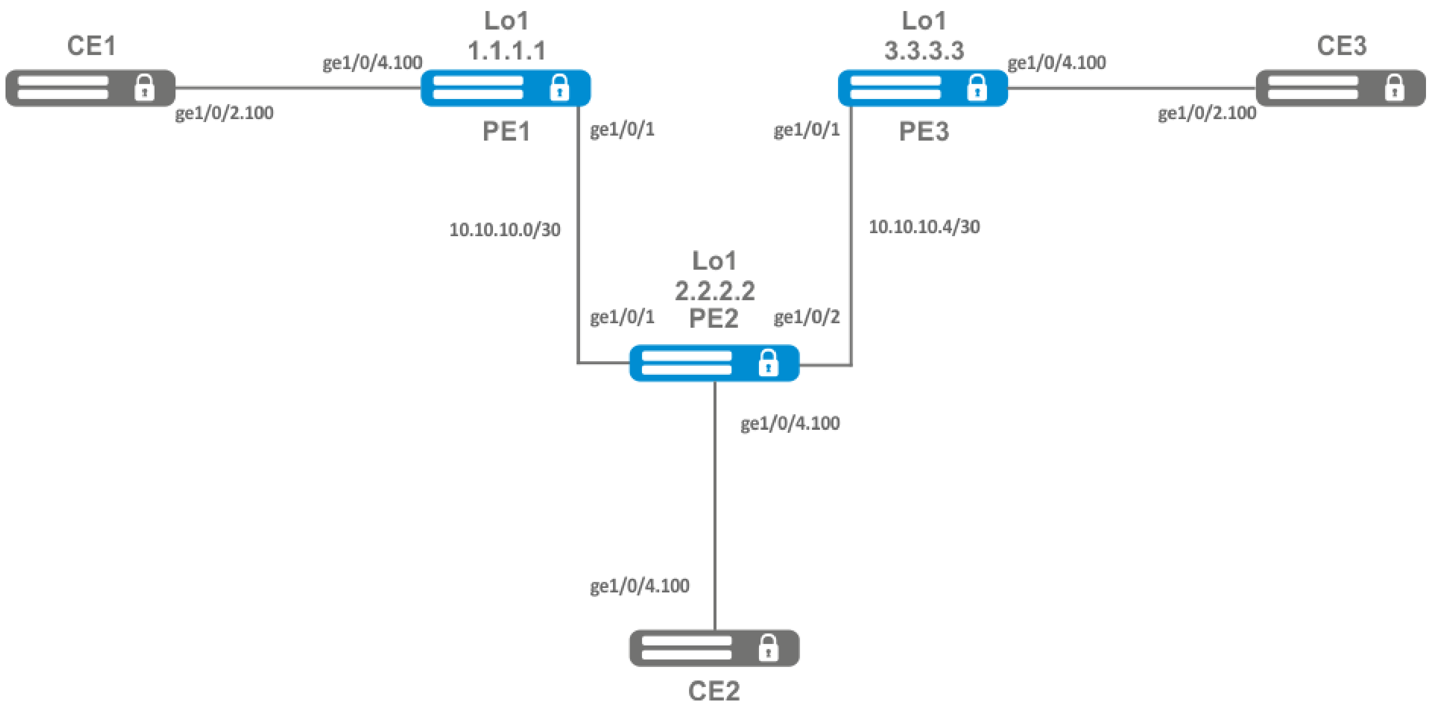
Step	Description	Command	Keys
1	Configure the LDP (see section LDP configuration).		
2	Create a network bridge in the system without specifying an IP address (see section Bridge configuration).		
3	Create pw-class in the system and switch to the pw-class configuration mode.	**esr(config-l2vpn)# pw-class <WORD>**	<WORD> — pw-class name [1..31] characters long.
4	Add a description for pw-class (optional).	**esr(config-l2vpn-pw-class)# description <LINE>**	<LINE> — description. Set by the string [1..255] characters long.
5	Set the MTU value for the pseudo-wire included in the pw-class (optional).	**esr(config-l2vpn-pw-class)# encapsulation mpls mtu <MTU>**	<MTU> — MTU value, takes values in the range of [552..10000]. Default value: 1500.
6	Disable status-tlv messaging (optional).	**esr(config-l2vpn-pw-class)# encapsulation mpls status-tlv disable**	Default value: status-tlv enable.
7	Create VPLS domain in the system and switch to the VPLS domain configuration mode.	**esr(config-l2vpn)# vpls <NAME>**	<NAME> — name of the p2p service, set by the string of up to 31 characters.
8	Enable VPLS tunnel.	**esr(config-l2vpn-vpls)# enable**	
9	Add bridge domain.	**esr (config-l2vpn-vpls)# bridge-group <ID>**	<ID> — bridge domain identifier, specified in the range [1..250].
10	Specify transport mode (optional).	**esr(config-l2vpn-vpls)# transport-mode { ethenet | vlan }**	<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.

Step	Description	Command	Keys
11	Create a pseudo-wire and switch to its parameters configuration mode	**esr(config-l2vpn-vpls)# pw <PW_ID> <LSR_ID>**	<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].
12	Add a description for pseudo-wire (optional).	**esr(config-l2vpn-pw)# description <LINE>**	<LINE> — description. Set by the string [1..255] characters long.
13	Set pw-class for pseudo-wire	**esr(config-l2vpn-pw)# pw-class <WORD>**	<WORD> — pw-class name [1..31] characters long.
14	Set the LSR address to which the pseudo-wire is set (Optional if the neighbor address is the same as the LSR_ID).	**esr(config-l2vpn-pw)# neighbor-address <ADDR>**	<ADDR> – router IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
15	Enable pseudo-wire.	**esr(config-l2vpn-pw)# enable**	
16	If the topology of the VPLS domain to be created requires more than one pseudo-wire, repeat steps 10 to 14.		
17	If it is necessary to change the default settings for a targeted LDP session, see section Configuring session parameters in targeted-LDP.		

12.5.4 L2VPN VPLS configuration example

Objective:

Configure l2vpn so that CE1, CE2, CE3 routers have L2 connectivity through the gi1/0/2.100 and gi1/0/4 (CE2) interfaces.



Solution:

Pre-requisite:

- Enable Jumbo frames support with the 'system jumbo-frames' command (the device must be rebooted for the changes to take effect);
- Configure IP addresses on interfaces according to the network structure shown in the figure above;
- Organize the exchange of routes between PE1, PE2 and PE3 using IGP (OSPF, IS-IS).

On router PE1, create a bridge group and enable it:

```
PE1# configure
PE1(config)# bridge 10
PE1(config-bridge)# enable
PE1(config-bridge)# exit
```

On the Interface to the CE1 side, include it in the created bridge group:

```
PE1(config)# interface gigabitethernet 1/0/4.100
PE1(config-subif)# bridge-group 10
PE1(config-subif)# exit
```

Set the MTU value on the interface towards PE2 to 9600 to avoid MTU overrun after encapsulating the MPLS header and disable the firewall:

```
PE1#(config)# interface gigabitethernet 1/0/1
PE1(config-if-gi)# mtu 9600
PE1(config-if-gi)# ip firewall disable
PE1(config-if-gi)# exit
```

Allow packets with an MPLS header to be received on the interface towards the MPLS network (in this example, the interface towards PE2):

```
PE1(config)# mpls
PE1(config-mpls)# forwarding interface gigabitethernet 1/0/1
```

Configure the LDP protocol and enable neighbor detection on the interface towards PE2:

```
PE1(config-mpls)# ldp
PE1(config-ldp)# router-id 1.1.1.1
PE1(config-ldp)# address-family ipv4
PE1(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
PE1(config-ldp-af-ipv4-if)# exit
PE1(config-ldp-af-ipv4)# transport-address 1.1.1.1
PE1(config-ldp-af-ipv4)# exit
PE1(config-ldp)# enable
PE1(config-ldp)# exit
```

Create a pw-class on the basis of which the virtual channels (pw) will be created later. Since, in this example, the default parameters will be applied to pw, it will be sufficient to specify the class name:

```
PE1(config-mpls)# l2vpn
PE1(config-l2vpn)# pw-class for_vpls1
PE1(config-l2vpn-pw-class)# exit
```

Create a new l2vpn of vpls type and add pw to routers PE2 and PE3, take the pw identifier as VID for convenience (in this case equal to 100):

```
PE1(config-l2vpn)# vpls vpls1
PE1(config-l2vpn-vpls)# bridge-group 10
PE1(config-l2vpn-vpls)# pw 100 2.2.2.2
PE1(config-l2vpn-pw)# pw-class for_vpls1
PE1(config-l2vpn-pw)# enable
PE1(config-l2vpn-pw)# exit
PE1(config-l2vpn-vpls)# pw 100 3.3.3.3
PE1(config-l2vpn-pw)# pw-class for_vpls1
PE1(config-l2vpn-pw)# enable
PE1(config-l2vpn-pw)# exit
PE1(config-l2vpn-vpls)# enable
PE1(config-l2vpn-vpls)# end
```

Apply the created configuration:

```
PE1# commit
PE1# confirm
```

Configure PE2 and PE3 routers in the same way as PE1:

```
PE2# configure
PE2(config)# bridge 10
PE2(config-bridge)# enable
PE2(config-bridge)# exit
PE2(config)# interface gigabitethernet 1/0/4.100
PE2(config-subif)# bridge-group 10
PE2(config-subif)# exit
PE2(config)# interface gigabitethernet 1/0/2
PE2(config-if-gi)# mtu 9600
PE2(config-if-gi)# ip firewall disable
PE2(config-if-gi)# exit
PE2(config)# mpls
PE2(config-mpls)# forwarding interface gigabitethernet 1/0/1
PE2(config-mpls)# forwarding interface gigabitethernet 1/0/2
PE2(config-mpls)# ldp
PE2(config-ldp)# enable
PE2(config-ldp)# router-id 2.2.2.2
PE2(config-ldp)# address-family ipv4
PE2(config-ldp-af-ipv4)# transport-address 2.2.2.2
PE2(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
PE2(config-ldp-af-ipv4-if)# exit
PE2(config-ldp-af-ipv4)# interface gigabitethernet 1/0/2
PE2(config-ldp-af-ipv4-if)# exit
PE2(config-ldp-af-ipv4)# exit
PE2(config-ldp)# exit
PE2(config-mpls)# l2vpn
PE2(config-l2vpn)# pw-class for_vpls1
PE2(config-l2vpn-pw-class)# exit
PE2(config-l2vpn)# vpls vpls1
PE2(config-l2vpn-vpls)# enable
PE2(config-l2vpn-vpls)# bridge-group 10
PE2(config-l2vpn-vpls)# pw 100 1.1.1.1
PE2(config-l2vpn-pw)# pw-class for_vpls1
PE2(config-l2vpn-pw)# enable
PE2(config-l2vpn-pw)# exit
PE2(config-l2vpn-vpls)# pw 100 3.3.3.3
PE2(config-l2vpn-pw)# pw-class for_vpls1
PE2(config-l2vpn-pw)# enable
PE2(config-l2vpn-pw)# end
PE2# commit
PE2# confirm
PE3(config)# bridge 10
PE3(config-bridge)# enable
PE3(config-bridge)# exit
PE3(config)# interface gigabitethernet 1/0/4.100
PE3(config-subif)# bridge-group 10
PE3(config-subif)# exit
PE3(config)# interface gigabitethernet 1/0/1
PE3(config-if-gi)# mtu 9600
PE3(config-if-gi)# ip firewall disable
PE3(config-if-gi)# exit
PE3(config)# mpls
PE3(config-mpls)# forwarding interface gigabitethernet 1/0/1
PE3(config-mpls)# exit
PE3(config)# mpls
PE3(config-mpls)# ldp
```

```
PE3(config-ldp)# enable
PE3(config-ldp)# router-id 3.3.3.3
PE3(config-ldp)# address-family ipv4
PE3(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
PE3(config-ldp-af-ipv4-if)# exit
PE3(config-ldp-af-ipv4)# transport-address 3.3.3.3
PE3(config-ldp-af-ipv4)# exit
PE3(config-ldp)# exit
PE3(config-mpls)# l2vpn
PE3(config-l2vpn)# pw-class for_vpls
PE3(config-l2vpn-pw-class)# exit
PE3(config-l2vpn)# vpls vpls1
PE3(config-l2vpn-vpls)# enable
PE3(config-l2vpn-vpls)# bridge-group 10
PE3(config-l2vpn-vpls)# pw 100 2.2.2.2
PE3(config-l2vpn-pw)# pw-class for_vpls
PE3(config-l2vpn-pw)# enable
PE3(config-l2vpn-pw)# exit
PE3(config-l2vpn-vpls)# pw 100 1.1.1.1
PE3(config-l2vpn-pw)# pw-class for_vpls
PE3(config-l2vpn-pw)# enable
PE3(config-l2vpn-pw)# end
PE3# commit
PE3# confirm
```

Make sure that the LDP neighborhood is established and display the virtual channel status (pseudowire) between PE1, PE2 and PE3:

```
PE3# show mpls ldp neighbor
Peer LDP ID: 1.1.1.1; Local LDP ID 3.3.3.3
    State: Operational
    TCP connection: 1.1.1.1:646 - 3.3.3.3:45979
    Messages sent/received: 22/22
    Uptime: 00:13:16
    LDP discovery sources:
        3.3.3.3 -> 1.1.1.1
Peer LDP ID: 2.2.2.2; Local LDP ID 3.3.3.3
    State: Operational
    TCP connection: 2.2.2.2:646 - 3.3.3.3:59627
    Messages sent/received: 22/22
    Uptime: 00:13:20
    LDP discovery sources:
        3.3.3.3 -> 2.2.2.2
        gigabitethernet 1/0/1
```

```
PE3# show mpls l2vpn pseudowire
Neighbor                                   PW ID     Type          Status
---------------------------------------- ----------     ----------      ------
1.1.1.1                                      100        Ethernet     Up
2.2.2.2                                      100        Ethernet     Up
```

The LDP neighborhood is established, pseudowire has moved to 'UP' status. The l2vpn configuration is now complete.

12.6 L2VPN Kompella mode configuration

Unlike Martini mode, where all operation is done by the LDP, in this mode the LDP does only operate with transport labels. Autodetection (not typical of LDP signaling), and the construction of a pseudowire connection is entrusted to BGP.

12.6.1 L2VPN VPLS configuration algorithm

Step	Description	Command	Keys
1	Configure the LDP (see section LDP configuration).		
2	Create a network bridge in the system without specifying an IP address (see section Bridge configuration).		
3	Create VPLS domain in the system and switch to the VPLS domain configuration mode.	**esr(config-l2vpn)# vpls <NAME>**	<NAME> — name of the p2p service, set by the string of up to 31 characters.
4	Enable VPLS tunnel.	**esr(config-l2vpn-vpls)# enable**	
5	Add bridge domain.	**esr(config-l2vpn-vpls)# bridge-group <ID>**	<ID> — bridge domain identifier, specified in the range [1..250].
6	Switch to the autodiscovery bgp configuration context.	**esr(config-l2vpn-vpls)# autodiscovery bgp**	
7	Specify route distinguisher for the given VPLS instance.	**esr(config-bgp)# rd <RD>**	<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];

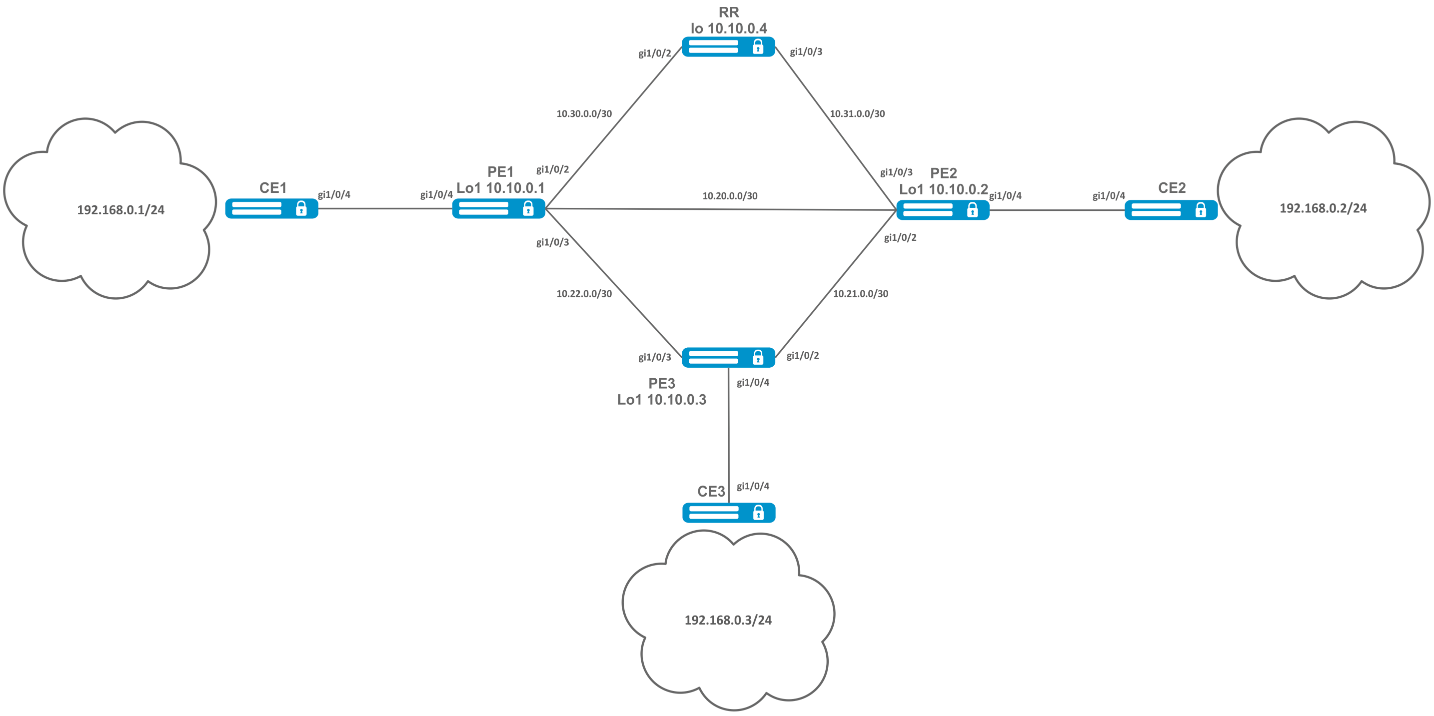
Step	Description	Command	Keys
8	Specify route target import for the given VPLS instance.	**esr(config-bgp)# route-target import <RT>**	<RT> – Route-target value, specified in one of the following forms: • <ASN>:<nn> — where <ASN> may take values [1..65535], nn may take values [1..65535]; • <ADDR>:<nn> — 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];
9	Specify route target export for the given VPLS instance.	**esr(config-bgp)# route-target export <RT>**	<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];
10	Specify ve id.	**esr(config-bgp)# ve id <ID>**	<ID> — VPLS instance identifier, specified in the range [1..16384].
11	Specify vpn id.	**esr (config-bgp)# vpn id <ID>**	<ID> — VPN identifier, specified in the range [1..4294967295]
12	Specify ve range (optional).	**esr (config-bgp)# ve range <RANGE>**	<RANGE> — range of VPLS border device identifiers [8..100].
13	Specify mtu (optional).	**esr (config-bgp)# mtu <VALUE>**	<VALUE> — MTU value [552..10000].

Step	Description	Command	Keys
14	Enable ignoring encapsulation type (optional).	**esr(config-bgp)# ignore encapsulation-mismatch**	
15	Enable ignoring MTU values (optional).	**esr(config-bgp)# ignore mtu-mismatch**	
16	In the context of address-family l2vpn vpls BGP configuration, enable extended attribute transfer.	**esr(config-bgp-neighbor-af)# send-community extended**	

12.6.2 L2VPN VPLS configuration example

Objective:

Configure L2VPN service: all CE devices must work within the same broadcast domain.



Solution:

First perform the following steps:

- Enable Jumbo frames support with the 'system jumbo-frames' command (the device must be rebooted for the changes to take effect);
- Configure IP addresses on interfaces according to the network structure shown in the figure above;
- Organize the exchange of routes between PE1, PE2, PE3 and RR using IGP (OSPF, IS-IS).

Configure the RR router:

```
hostname RR

system jumbo-frames

router ospf 1
area 0.0.0.0
enable
exit
enable
exit

interface gigabitethernet 1/0/2
mtu 9500
ip firewall disable
ip address 10.30.0.2/30
ip ospf instance 1
ip ospf
exit
interface gigabitethernet 1/0/3
mtu 9500
ip firewall disable
ip address 10.31.0.2/30
ip ospf instance 1
ip ospf
exit
interface loopback 1
ip address 10.10.0.4/32
ip ospf instance 1
ip ospf
exit
mpls
ldp
router-id 10.10.0.4
address-family ipv4
interface gigabitethernet 1/0/2
exit
interface gigabitethernet 1/0/3
exit
exit
enable
exit
forwarding interface gigabitethernet 1/0/2
forwarding interface gigabitethernet 1/0/3
exit
```

Configure the BGP Route Reflector for the address family l2vpn:

```
RR(config)# router bgp 65500
RR(config-bgp)#   router-id 10.10.0.4
RR(config-bgp)#   neighbor 10.10.0.1
RR(config-bgp-neighbor)#     remote-as 65500
RR(config-bgp-neighbor)#     route-reflector-client
RR(config-bgp-neighbor)#     update-source 10.10.0.4
RR(config-bgp-neighbor)#     address-family l2vpn vpls
RR(config-bgp-neighbor-af)#       send-community extended
RR(config-bgp-neighbor-af)#       enable
RR(config-bgp-neighbor-af)#     exit
RR(config-bgp-neighbor)#     enable
RR(config-bgp-neighbor)#   exit
RR(config-bgp)#   neighbor 10.10.0.2
RR(config-bgp-neighbor)#     remote-as 65500
RR(config-bgp-neighbor)#     route-reflector-client
RR(config-bgp-neighbor)#     update-source 10.10.0.4
RR(config-bgp-neighbor)#     address-family l2vpn vpls
RR(config-bgp-neighbor-af)#       send-community extended
RR(config-bgp-neighbor-af)#       enable
RR(config-bgp-neighbor-af)#     exit
RR(config-bgp-neighbor)#     enable
RR(config-bgp-neighbor)#   exit
RR(config-bgp)#   neighbor 10.10.0.3
RR(config-bgp-neighbor)#     remote-as 65500
RR(config-bgp-neighbor)#     route-reflector-client
RR(config-bgp-neighbor)#     update-source 10.10.0.4
RR(config-bgp-neighbor)#     address-family l2vpn vpls
RR(config-bgp-neighbor-af)#       send-community extended
RR(config-bgp-neighbor-af)#       enable
RR(config-bgp-neighbor-af)#     exit
RR(config-bgp-neighbor)#     enable
RR(config-bgp-neighbor)#   exit
RR(config-bgp)#   enable
```

Configure BGP on the PE routers:

Pre-configuration

```
hostname PE1

system jumbo-frames

router ospf 1
area 0.0.0.0
enable
exit
enable
exit

interface gigabitethernet 1/0/1
mtu 9500
```

Pre-configuration

```
ip firewall disable
ip address 10.20.0.1/30
ip ospf instance 1
ip ospfexit
interface gigabitethernet 1/0/2
mtu 9500
ip firewall disable
ip address 10.30.0.1/30
ip ospf instance 1
ip ospf
exitinterface gigabitethernet 1/0/3
mtu 9500
ip firewall disable
ip address 10.22.0.1/30
ip ospf instance 1
ip ospf
exit
interface loopback 1
ip address 10.10.0.1/32
ip ospf instance 1
ip ospf
exit
mpls
ldp
router-id 10.10.0.1
address-family ipv4
interface gigabitethernet 1/0/1
exit
interface gigabitethernet 1/0/2
exit
interface gigabitethernet 1/0/3
exit

exit

enable

exit
forwarding interface gigabitethernet 1/0/1
forwarding interface gigabitethernet 1/0/2
forwarding interface gigabitethernet 1/0/3
exit
```

BGP configuration:

```
PE1(config)# router bgp 65500
PE1(config-bgp)#   neighbor 10.10.0.4
PE2(config-bgp)#   router-id 10.10.0.1
PE1(config-bgp-neighbor)#     remote-as 65500
PE1(config-bgp-neighbor)#     update-source 10.10.0.1
PE1(config-bgp-neighbor)#     address-family l2vpn vpls
PE1(config-bgp-neighbor-af)#       send-community extended
PE1(config-bgp-neighbor-af)#       enable
PE1(config-bgp-neighbor-af)#     exit
PE1(config-bgp-neighbor)#     enable
PE1(config-bgp-neighbor)#   exit
PE1(config-bgp)#   enable
PE1(config-bgp)# exit
```

Check that the BGP session with RR is successfully established:

```
PE1# show bgp neighbors
BGP neighbor is 10.10.0.4
BGP state: Established
Neighbor address: 10.10.0.4
Neighbor AS: 65500
Neighbor ID: 10.10.0.4
Neighbor caps: refresh enhanced-refresh restart-aware AS4
Session: internal multihop AS4
Source address: 10.10.0.1
Weight: 0
Hold timer: 110/180
Keepalive timer: 21/60
Uptime: 7375 s
```

Configuration of BGP on PE2:

Pre-configuration

```
hostname PE2

system jumbo-frames

router ospf 1
area 0.0.0.0
enable
exit
enable
exit
```

Pre-configuration

```
interface gigabitethernet 1/0/1
mtu 9500
ip firewall disable
ip address 10.20.0.2/30
ip ospf instance 1
ip ospf
exit
interface gigabitethernet 1/0/2
mtu 9500
ip firewall disable
ip address 10.21.0.1/30
ip ospf instance 1
ip ospf
exit
interface gigabitethernet 1/0/3
mtu 9500
ip firewall disable
ip address 10.31.0.1/30
ip ospf instance 1
ip ospf
exit
interface loopback 1
ip address 10.10.0.2/32
ip ospf instance 1
ip ospf
exit
mpls
ldp
router-id 10.10.0.2
address-family ipv4
interface gigabitethernet 1/0/1
exit
interface gigabitethernet 1/0/2
exit
interface gigabitethernet 1/0/3
exit
exit

enable

exit
forwarding interface gigabitethernet 1/0/1
forwarding interface gigabitethernet 1/0/2
forwarding interface gigabitethernet 1/0/3
exit
```

```
PE2(config)# router bgp 65500
PE2(config-bgp)#   router-id 10.10.0.2
PE2(config-bgp)#   neighbor 10.10.0.4
PE2(config-bgp-neighbor)#     remote-as 65500
PE2(config-bgp-neighbor)#     update-source 10.10.0.2
PE2(config-bgp-neighbor)#     address-family l2vpn vpls
PE2(config-bgp-neighbor-af)#       send-community extended
PE2(config-bgp-neighbor-af)#       enable
PE2(config-bgp-neighbor-af)#     exit
PE2(config-bgp-neighbor)#     enable
PE2(config-bgp-neighbor)#   exit
PE2(config-bgp)#   enable
PE2(config-bgp)# exit
```

Check that the session with RR is successfully established:

```
PE2# show bgp neighbors
BGP neighbor is 10.10.0.4
BGP state: Established
Neighbor address: 10.10.0.4
Neighbor AS: 65500
Neighbor ID: 10.10.0.4
Neighbor caps: refresh enhanced-refresh restart-aware AS4
Session: internal multihop AS4
Source address: 10.10.0.2
Weight: 0
Hold timer: 113/180
Keepalive timer: 56/60
Uptime: 47 s
```

Configuration of BGP on PE3:

Pre-configuration

```
hostname PE3

system jumbo-frames

router ospf 1
area 0.0.0.0
enable
exit
enable
exit

interface gigabitethernet 1/0/2
mtu 9500
ip firewall disable
ip address 10.21.0.2/30
ip ospf instance 1
ip ospf
exit
interface gigabitethernet 1/0/3
mtu 9500
ip firewall disable
ip address 10.22.0.2/30
ip ospf instance 1
ip ospf
exit
interface loopback 1
ip address 10.10.0.3/24
ip ospf instance 1
ip ospf
exit
mpls
ldp
router-id 10.10.0.3
address-family ipv4
interface gigabitethernet 1/0/2
exit
interface gigabitethernet 1/0/3
exit
exit
enable
exit
forwarding interface gigabitethernet 1/0/2
forwarding interface gigabitethernet 1/0/3
exit
```

```
PE3(config)# router bgp 65500
PE3(config-bgp)#   router-id 10.10.0.3
PE3(config-bgp)#   neighbor 10.10.0.4
PE3(config-bgp-neighbor)#     remote-as 65500
PE3(config-bgp-neighbor)#     update-source 10.10.0.3
PE3(config-bgp-neighbor)#     address-family l2vpn vpls
PE3(config-bgp-neighbor-af)#       send-community extended
PE3(config-bgp-neighbor-af)#       enable
PE3(config-bgp-neighbor-af)#     exit
PE3(config-bgp-neighbor)#     enable
PE3(config-bgp-neighbor)#   exit
PE3(config-bgp)#   enable
PE3(config-bgp)# exit
```

Check that the BGP session is successfully established:

```
PE3# show bgp neighbors
BGP neighbor is 10.10.0.4
BGP state: Established
Neighbor address: 10.10.0.4
Neighbor AS: 65500
Neighbor ID: 10.10.0.4
Neighbor caps: refresh enhanced-refresh restart-aware AS4
Session: internal multihop AS4
Source address: 10.10.0.3
Weight: 0
Hold timer: 141/180
Keepalive timer: 27/60
Uptime: 77 s
```

The next step is to create a bridge domain on each PE router, and include an interface (Attachment circuit, AC) that looks towards CE:

PE1:

```
PE1(config)# bridge 1
PE1(config-bridge)# enable
PE1(config-bridge)# exit
PE1(config)# interface gigabitethernet 1/0/4
PE1(config-if-gi)# mode switchport
PE1(config-if-gi)# bridge-group 1
```

Check that the interface is included into the bridge domain:

```
PE1# show interfaces bridge
Bridges      Interfaces
----------   --------------------------------------------------------------
bridge 1     gi1/0/4

PE1# sh interfaces status bridge 1
Interface 'bridge 1' status information:
 Description:           --
 Operational state:     Up
 Administrative state: Up
 Supports broadcast:    Yes
 Supports multicast:    Yes
 MTU:                   1500
 MAC address:           a8:f9:4b:ac:4d:15
 Last change:           4 minutes and 22 seconds
 Mode:                  Routerport
```

PE2:

```
PE2(config)# bridge 1
PE2(config-bridge)# enable
PE2(config-bridge)# exit
PE2(config)# interface gigabitethernet 1/0/4
PE2(config-if-gi)# mode switchport
PE2(config-if-gi)# bridge-group 1
```

```
PE2# show interfaces bridge 1
Bridges      Interfaces
----------   --------------------------------------------------------------
bridge 1     gi1/0/4

PE2# sh interfaces status bridge 1
Interface 'bridge 1' status information:
 Description:           --
 Operational state:     Up
 Administrative state: Up
 Supports broadcast:    Yes
 Supports multicast:    Yes
 MTU:                   1500
 MAC address:           a8:f9:4b:ad:f2:45
 Last change:           10 seconds
 Mode:                  routerport
```

PE3:

```
PE3(config)# bridge 1
PE3(config-bridge)# enable
PE3(config-bridge)# exit
PE3(config)# interface gigabitethernet 1/0/4
PE3(config-if-gi)# mode switchport
PE3(config-if-gi)# bridge-group 1
```

```
PE3# show interfaces bridge
Bridges      Interfaces
----------   --------------------------------------------------------------
bridge 1     gi1/0/4
PE3# sh interfaces status bridge
Interface            Admin    Link     MTU      MAC address          Last change
Mode
                     state    state
------------------   -----    -----    ------   ------------------   -------------------------
----------
bridge 1             Up       Up       1500     a8:f9:4b:ac:df:f0    1 minute and 21 seconds
Routerport

PE3# sh interfaces status bridge 1
Interface 'bridge 1' status information:
 Description:           --
 Operational state:     Up
 Administrative state: Up
 Supports broadcast:    Yes
 Supports multicast:    Yes
 MTU:                   1500
 MAC address:           a8:f9:4b:ac:df:f0
 Last change:           1 minute and 24 seconds
 Mode:                  Routerport
```

Next, perform the VPLS configuration:

PE1:

Switch to the L2VPN configuration context and include the previously created bridge domain.

```
PE1(config)# mpls
PE1(config-mpls)# l2vpn
PE1(config-l2vpn)# vpls l2vpn
PE1(config-l2vpn-vpls)# bridge-group 1
```

Specify RD, RT, VE-ID, VPN-ID according to the network scheme and activate the service:

> **In some cases entering such parameters as RD and RT can be skipped: if only VPN ID is specified, they will be formed as follows: <AS number> : <vpn-id>.**
> **For example, we have an AS 65550 autonomous system number, vpn-id is 10, then the following parameters will be generated:**
> **RD - 65550: 10.**
> **RT import/export - 65550:10.**

```
PE1(config-l2vpn-vpls)# autodiscovery bgp
PE1(config-bgp)# rd 65500:100
PE1(config-bgp)# route-target import 65500:100
PE1(config-bgp)# route-target export 65500:100
PE1(config-bgp)# ve id 1
PE1(config-bgp)# vpn id 1
PE1(config-bgp)# exit
PE1(config-l2vpn-vpls)# enable
```

After activating the service, check that route information appeared in the l2vpn table, and it is advertised on RR:

```
PE1# 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
----- --------------------- ----- ----- ----- ---------- ---------- ---------- ------ ------
*>    65500:100              1     1     10    --         --         --         --


PE1# show ip bgp l2vpn vpls all neighbor 10.10.0.4 advertise-routes
Origin codes: i - IGP, e - EGP, ? - incomplete

Route Distinguisher    VID   VBO   VBS   Next hop        Metric     LocPrf     Path
--------------------- ----- ----- ----- --------------- ---------- ---------- ------
65500:100              1     1     10    10.10.0.1       --         100        i


* Detailed output of the advertised route *

PE1# show ip bgp l2vpn vpls all neighbor 10.10.0.4 advertise-routes  ve-id 1 block
-offset 1
BGP routing table entry for 65500:100 VE ID 1 VE Block Offset 1
    VE Block Size:       10
    Label Base:          86
    Next hop:            10.10.0.1
    AS path:             --
    Origin:              IGP
    Local preference:    100
    Extended Community: RT:65500:100
    Layer2-info:         encaps (VPLS), control flags(0x00), MTU (1500)
```

Proceed to the PE2 configuration:

```
PE2(config-mpls)# l2vpn
PE2(config-l2vpn)# vpls l2vpn
PE2(config-l2vpn-vpls)# bridge-group 1
PE2(config-l2vpn-vpls)# autodiscovery bgp
PE2(config-bgp)# rd 65500:100
```

```
PE2(config-bgp)# route-target export 65500:100
PE2(config-bgp)# route-target import  65500:100
PE2(config-bgp)# vpn id 2
PE2(config-bgp)# ve id 2
PE2(config-bgp)# exit
PE2(config-l2vpn-vpls)# enable
```

Check that PE2 is advertising the route information on RR:

```
PE2# show ip bgp l2vpn vpls all  neighbor 10.10.0.4 advertise-routes
Origin codes: i - IGP, e - EGP, ? - incomplete

Route Distinguisher    VID   VBO   VBS   Next hop      Metric     LocPrf     Path
--------------------- ----- ----- ----- ------------- ---------- ---------- ------
65500:100              2     1     10    10.10.0.2     --         100        i
```

In the l2vpn table you can see its routes as well as routes from PE1:

```
PE2# 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
----- --------------------- ----- ----- ----- ----------- ---------- ---------- ------ -----
*>    65500:100               2     1     10    --          --         --         --
*>i   65500:100               1     1     10    10.10.0.1   --         100        0      i
```

The calculated service marks can be viewed as follows:

1)

```
PE2# show mpls l2vpn bindings
Neighbor: 10.10.0.1, PW ID: 2, VE ID: 1
Local label: 45
Encasulation Type: VPLS
Control flags: 0x00
MTU: 1500
Remote label: 87
Encasulation Type: VPLS
Control flags: 0x00
MTU: 1500
```

2)

```
PE2# show mpls forwarding-table
Local       Outgoing         Prefix             Outgoing            Next Hop
label       label            or tunnel ID       Interface
--------    --------         -----------------  --------------      ----------
45          87               PW ID 2                 --             10.10.0.1
```

Check the service state:

```
PE2# show mpls l2vpn vpls l2vpn
VPLS: l2vpn
    bridge 1:
        MTU:     1500
        Status: Up
    ACs:
      gigabitethernet 1/0/4:
        MTU:     1500
        Status: Up
    PWs:
      PW ID 2, Neighbor 10.10.0.1:
        MTU:          1500
        Last change: 00:21:33
        Status:       Up
```

Proceed to the PE3 configuration:

```
PE3#    config
PE3(config)# mpls
PE3(config-mpls)# l2vpn
PE3(config-l2vpn)# vpls l2vpn
PE3(config-l2vpn-vpls)# bridge-group 1
PE3(config-l2vpn-vpls)# autodiscovery bgp
PE3(config-bgp)# rd 65500:100
PE3(config-bgp)# route-target export 65500:100
PE3(config-bgp)# route-target import  65500:100
PE3(config-bgp)# ve id 3
PE3(config-bgp)# vpn id 3
PE3(config-bgp)# exit
PE3(config-l2vpn-vpls)# enable
```

Check the routing information in PE3:

```
PE3# 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
----- --------------------- ----- ----- ----- ----------- ---------- ---------- ------ -----
*>    65500:100             3     1     10    --          --         --         --
*>i   65500:100             2     1     10    10.10.0.2   --         100        0      i
*>i   65500:100             1     1     10    10.10.0.1   --         100        0      i
```

Check that PE3 is advertising the route information on RR:

```
PE3# show ip bgp l2vpn vpls all neighbor 10.10.0.4 advertise-routes
Origin codes: i - IGP, e - EGP, ? - incomplete

Route Distinguisher   VID   VBO   VBS   Next hop    Metric  LocPrf   Path
--------------------- ----- ----- ----- ----------- ------- -------- -----
65500:100             3     1     10    10.10.0.3   --      100      i
```

Check that the pseudowire is built before both PEs and is in the 'UP' status:

```
PE3# show mpls l2vpn vpls l2vpn
VPLS: l2vpn
    bridge 1:
        MTU:     1500
        Status: Up
    ACs:
      gigabitethernet 1/0/4:
        MTU:     1500
        Status: Up
    PWs:
      PW ID 3, Neighbor 10.10.0.2:
        MTU:          1500
        Last change: 00:06:08
        Status:       Up
      PW ID 3, Neighbor 10.10.0.1:
        MTU:          1500
        Last change: 00:06:08
        Status:       Up
```

Check the network availability of client equipment (CE):

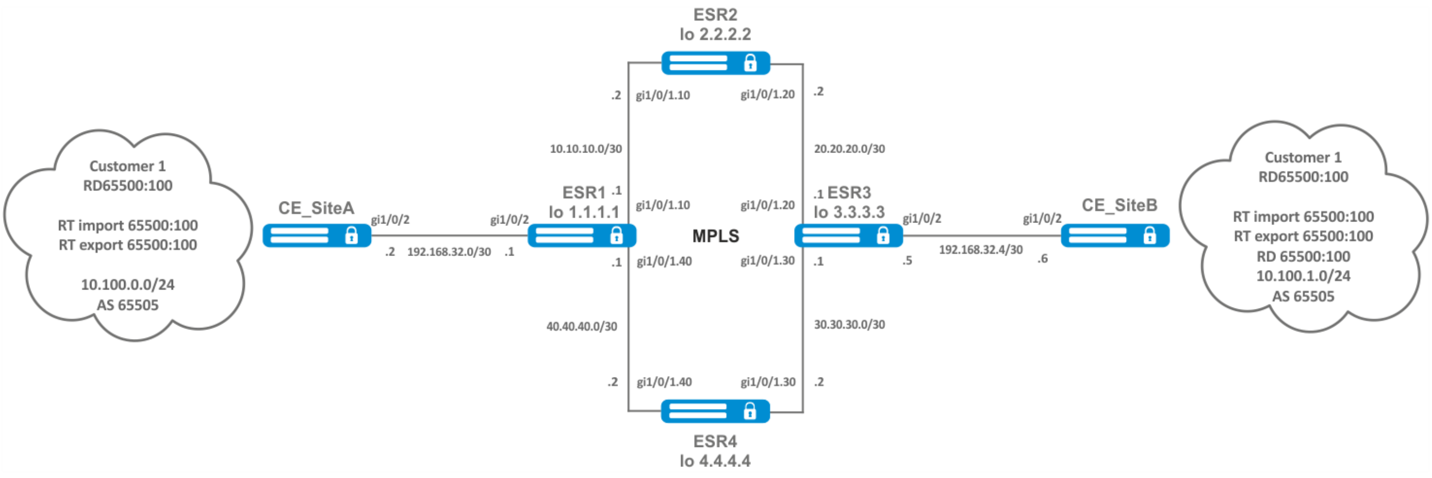
```
CE3# ping 192.168.0.1
PING 192.168.0.1 (192.168.0.1) 56(84) bytes of data.
!!!!!
--- 192.168.0.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 0.173/0.208/0.290/0.045 ms
CE3# ping 192.168.0.2
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.
!!!!!
--- 192.168.0.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 0.158/0.204/0.255/0.032 ms

PE3# sh mac address-table bridge 1
VID     MAC Address         Interface                        Type
-----   -----------------   ------------------------------   -------
--      a8:f9:4b:aa:11:08   gigabitethernet 1/0/4            Dynamic
--      a8:f9:4b:aa:11:06   dypseudowire 3_10.10.0.1         Dynamic
--      a8:f9:4b:aa:11:07   dypseudowire 3_10.10.0.2         Dynamic
3 valid mac entries
```

L2VPN service configuration is now complete.

12.7 L3VPN configuration

L3VPN service allows to combine distributed client IP networks, and ensure the transfer of traffic between them within a single VRF.

⚠ **The current implementation of MP-BGP only supports VPN-IPv4 routes (AF I= 1, SAFI = 128).**



12.7.1 Configuration algorithm

Step	Description	Command	Keys
1	Configure addressing and one of IGP on all P and PE routers.		
2	Configure LDP transport tag distribution.		
3	Create VRF.	**esr(config)# ip vrf <VRF>**	<VRF> – VRF instance name, set by the string of up to 31 characters.
4	Specify route distinguisher for the given VRF.	**esr(config-vrf)# rd <RD>**	<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];

Step	Description	Command	Keys
5	Specify route target import for the given VRF.	**esr(config-vrf)# route-target import <RT>**	<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];
6	Specify route target export for the given VRF.	**esr(config-vrf)# route-target export <RT>**	<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];

Step	Description	Command	Keys

7	Specify the allowed number of routes for this VRF.	**esr(config-vrf)# ip protocols <PROTOCOLS> max-routes <VALUE>**	<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-1000/1200/1500/1511/1700/3100/3200 – [1..5000000];
			  • ESR-20/21/30/100/200 – [1..2500000],
			  • ESR-10/12V/12VF/14VF/15 – [1..1000000].
			• OSPF and IS-IS
			  • ESR-1000/1200/1500/1511/1700/3100/3200 – [1..500000];
			  • ESR-20/21/30/100/200 – [1..300000];
			  • ESR-10/12V/12VF/14VF/15 – [1..30000].

8	In the context of address-family VPNv4 BGP configuration, enable extended attribute transfer.	**esr(config-bgp-neighbor-af)# send-community extended**	

12.7.2 Configuration example

Objective:

Configure L3VPN based on MPLS technology between ESR1 and ESR3. The final result of the configuration is the appearance of connectivity between nodes connected to the VRF on different routers in the network (i.e. the union of VRFs on different routers via MPLS transport). In this case, transfer of MPLS service tags for L3VPN service via MP-BGP and transfer of transport tags to reach nexthop addresses of received BGP routes must be provided.

Solution:

Configuring addressing and enabling IGP and P/PE on routers:

ESR1

```
ESR1(config)# router ospf log-adjacency-changes
ESR1(config)# router ospf 1
ESR1(config-ospf)# router-id 1.1.1.1
ESR1(config-ospf)# area 0.0.0.0
ESR1(config-ospf-area)# enable
ESR1(config-ospf-area)# exit
ESR1(config-ospf)# enable
ESR1(config-ospf)# exit
ESR1(config)#
ESR1(config)# interface loopback 1
ESR1(config-loopback)# ip address 1.1.1.1/32
ESR1(config-loopback)# ip ospf instance 1
ESR1(config-loopback)# ip ospf
ESR1(config-loopback)# exit
ESR1(config)#
ESR1(config)# interface gigabitethernet 1/0/1.10
ESR1(config-subif)# ip firewall disable
ESR1(config-subif)# ip address 10.10.10.1/30
ESR1(config-subif)# ip ospf instance 1
ESR1(config-subif)# ip ospf
ESR1(config-subif)# exit
ESR1(config)#
ESR1(config)# interface gigabitethernet 1/0/1.40
ESR1(config-subif)# ip firewall disable
ESR1(config-subif)# ip address 40.40.40.1/30
ESR1(config-subif)# ip ospf instance 1
ESR1(config-subif)# ip ospf
ESR1(config-subif)# exit
ESR1(config)#
ESR1(config)# system jumbo-frames
ESR1(config)# do commit
ESR1(config)# do confirm
```

ESR2

```
ESR2(config)# router ospf log-adjacency-changes
ESR2(config)# router ospf 1
ESR2(config-ospf)# router-id 2.2.2.2
ESR2(config-ospf)# area 0.0.0.0
ESR2(config-ospf-area)# enable
ESR2(config-ospf-area)# exit
ESR2(config-ospf)# enable
ESR2(config-ospf)# exit
ESR2(config)#
ESR2(config)# interface loopback 1
ESR2(config-loopback)# ip address 2.2.2.2/32
ESR2(config-loopback)# ip ospf instance 1
ESR2(config-loopback)# ip ospf
ESR2(config-loopback)# exit
ESR2(config)#
ESR2(config)# interface gigabitethernet 1/0/1.10
ESR2(config-subif)# ip firewall disable
ESR2(config-subif)# ip address 10.10.10.2/30
ESR2(config-subif)# ip ospf instance 1
ESR2(config-subif)# ip ospf
ESR2(config-subif)# exit
ESR2(config)#
ESR2(config)# interface gigabitethernet 1/0/1.20
ESR2(config-subif)# ip firewall disable
ESR2(config-subif)# ip address 20.20.20.2/30
ESR2(config-subif)# ip ospf instance 1
ESR2(config-subif)# ip ospf
ESR2(config-subif)# exit
ESR2(config)#
ESR2(config)# system jumbo-frames
ESR2(config)# do commit
ESR2(config)# do confirm
```

ESR3

```
ESR3(config)# router ospf log-adjacency-changes
ESR3(config)# router ospf 1
ESR3(config-ospf)# router-id 3.3.3.3
ESR3(config-ospf)# area 0.0.0.0
ESR3(config-ospf-area)# enable
ESR3(config-ospf-area)# exit
ESR3(config-ospf)# enable
ESR3(config-ospf)# exit
ESR3(config)#
ESR3(config)# interface loopback 1
ESR3(config-loopback)# ip address 3.3.3.3/32
ESR3(config-loopback)# ip ospf instance 1
ESR3(config-loopback)# ip ospf
ESR3(config-loopback)# exit
ESR3(config)#
ESR3(config)# interface gigabitethernet 1/0/1.20
ESR3(config-subif)# ip firewall disable
ESR3(config-subif)# ip address 20.20.20.1/30
ESR3(config-subif)# ip ospf instance 1
ESR3(config-subif)# ip ospf
ESR3(config-subif)# exit
ESR3(config)#
ESR3(config)# interface gigabitethernet 1/0/1.30
ESR3(config-subif)# ip firewall disable
ESR3(config-subif)# ip address 30.30.30.1/30
ESR3(config-subif)# ip ospf instance 1
ESR3(config-subif)# ip ospf
ESR3(config-subif)# exit
ESR3(config)#
ESR3(config)# system jumbo-frames
ESR3(config)# do commit
ESR3(config)# do confirm
```

ESR4

```
ESR4(config)# router ospf log-adjacency-changes
ESR4(config)# router ospf 1
ESR4(config-ospf)# router-id 4.4.4.4
ESR4(config-ospf)# area 0.0.0.0
ESR4(config-ospf-area)# enable
ESR4(config-ospf-area)# exit
ESR4(config-ospf)# enable
ESR4(config-ospf)# exit
ESR4(config)#
ESR4(config)# interface loopback 1
ESR4(config-loopback)# ip address 4.4.4.4/32
ESR4(config-loopback)# ip ospf instance 1
ESR4(config-loopback)# ip ospf
ESR4(config-loopback)# exit
ESR4(config)#
ESR4(config)# interface gigabitethernet 1/0/1.40
ESR4(config-subif)# ip firewall disable
ESR4(config-subif)# ip address 40.40.40.2/30
ESR4(config-subif)# ip ospf instance 1
ESR4(config-subif)# ip ospf
ESR4(config-subif)# exit
ESR4(config)#
ESR4(config)# interface gigabitethernet 1/0/1.30
ESR4(config-subif)# ip firewall disable
ESR4(config-subif)# ip address 30.30.30.2/30
ESR4(config-subif)# ip ospf instance 1
ESR4(config-subif)# ip ospf
ESR4(config-subif)# exit
ESR4(config)#
ESR4(config)# system jumbo-frames
ESR4(config)# do commit
ESR4(config)# do confirm
```

It is necessary to make sure that the OSPF protocol is running on each router:

```
ESR1# show ip ospf neighbors


Router ID        Pri  State          DTime  Interface      Router IP
---------        ---  -----          -----  -------------  ---------
2.2.2.2          128  Full/BDR       00:39  gi1/0/1.10     10.10.10.2
4.4.4.4          128  Full/BDR       00:32  gi1/0/1.40     40.40.40.2


ESR1# show ip ospf
O       40.40.40.0/30       [150/10]          dev gi1/0/1.40                    [ospf1 1970-01-0
8]  (1.1.1.1)
O     * 30.30.30.0/30       [150/20]          via 40.40.40.2 on gi1/0/1.40      [ospf1 1970-01-0
8]  (3.3.3.3)
O       1.1.1.1/32          [150/0]           dev lo1                           [ospf1 1970-01-0
8]  (1.1.1.1)
O     * 4.4.4.4/32          [150/10]          via 40.40.40.2 on gi1/0/1.40      [ospf1 1970-01-0
8]  (4.4.4.4)
O     * 20.20.20.0/30       [150/20]          via 10.10.10.2 on gi1/0/1.10      [ospf1 22:05:45]
    (3.3.3.3)
O       10.10.10.0/30       [150/10]          dev gi1/0/1.10                    [ospf1 22:05:33]
    (1.1.1.1)
O     * 3.3.3.3/32          [150/20]          multipath                         [ospf1 22:05:45]
    (3.3.3.3)
                                              via 40.40.40.2 on gi1/0/1.40 weight 1
O     * 2.2.2.2/32          [150/10]          via 10.10.10.2 on gi1/0/1.10      [ospf1 22:05:45]
    (2.2.2.2)
```

LDP configuration on the P/PE routers :

ESR1

```
ESR1# config
ESR1(config)# mpls
ESR1(config-mpls)# ldp
ESR1(config-ldp)# address-family ipv4
ESR1(config-ldp-af-ipv4)# transport-address 1.1.1.1
ESR1(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1.10
ESR1(config-ldp-af-ipv4-if)# exit
ESR1(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1.40
ESR1(config-ldp-af-ipv4-if)# exit
ESR1(config-ldp-af-ipv4)# exit
ESR1(config-ldp)# enable
ESR1(config-ldp)# exit
ESR1(config-mpls)# forwarding interface gigabitethernet 1/0/1.10
ESR1(config-mpls)# forwarding interface gigabitethernet 1/0/1.40
ESR1(config-mpls)# exit
ESR1(config)# do commit
ESR1(config)# do confirm
```

ESR2

```
ESR2# config
ESR2(config)# mpls
ESR2(config-mpls)# ldp
ESR2(config-ldp)# address-family ipv4
ESR2(config-ldp-af-ipv4)# transport-address 2.2.2.2
ESR2(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1.10
ESR2(config-ldp-af-ipv4-if)# exit
ESR2(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1.20
ESR2(config-ldp-af-ipv4-if)# exit
ESR2(config-ldp-af-ipv4)# exit
ESR2(config-ldp)# enable
ESR2(config-ldp)# exit
ESR2(config-mpls)# forwarding interface gigabitethernet 1/0/1.10
ESR2(config-mpls)# forwarding interface gigabitethernet 1/0/1.20
ESR2(config-mpls)# exit
ESR2(config)# do commit
ESR2(config)# do confirm
```

ESR3

```
ESR3# config
ESR3(config)# mpls
ESR3(config-mpls)# ldp
ESR3(config-ldp)# address-family ipv4
ESR3(config-ldp-af-ipv4)# transport-address 3.3.3.3
ESR3(config-ldp-af-ipv4)#  interface gigabitethernet 1/0/1.20
ESR3(config-ldp-af-ipv4-if)# exit
ESR3(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1.30
ESR3(config-ldp-af-ipv4-if)# exit
ESR3(config-ldp-af-ipv4)# exit
ESR3(config-ldp)# enable
ESR3(config-ldp)# exit
ESR3(config-mpls)# forwarding interface gigabitethernet 1/0/1.20
ESR3(config-mpls)# forwarding interface gigabitethernet 1/0/1.30
ESR3(config-mpls)# exit
ESR3(config)# do commit
ESR3(config)# do confirm
```

ESR4

```
ESR4# config
ESR4(config)# mpls
ESR4(config-mpls)# ldp
ESR4(config-ldp)# address-family ipv4
ESR4(config-ldp-af-ipv4)# transport-address 4.4.4.4
ESR4(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1.30
ESR4(config-ldp-af-ipv4-if)# exit
ESR4(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1.40
ESR4(config-ldp-af-ipv4-if)# exit
ESR4(config-ldp-af-ipv4)# exit
ESR4(config-ldp)# enable
ESR4(config-ldp)# exit
ESR4(config-mpls)# forwarding interface gigabitethernet 1/0/1.30
ESR4(config-mpls)# forwarding interface gigabitethernet 1/0/1.40
ESR4(config-mpls)# exit
ESR4(config)# do commit
ESR4(config)# do confirm
```

One of the following commands can be used to check the LDP convergence:

```
ESR1# show mpls ldp neighbor
Peer LDP ID: 2.2.2.2; Local LDP ID 1.1.1.1
    State: Operational
    TCP connection: 2.2.2.2:33933 - 1.1.1.1:646
    Messages sent/received: 1059/1070
    Uptime: 17:32:07
    LDP discovery sources:
        gigabitethernet 1/0/1.10
Peer LDP ID: 4.4.4.4; Local LDP ID 1.1.1.1
    State: Operational
    TCP connection: 4.4.4.4:40894 - 1.1.1.1:646
    Messages sent/received: 1376/1386
    Uptime: 22:38:38
    LDP discovery sources:
        gigabitethernet 1/0/1.40
```

MP-BGP configuration

Create VRF on ESR1 and ESR3, respectively. Specify RD, rt-export/import in accordance with the scheme. Configure interface for interaction with CE (CE-SiteA и CE-SiteB). Additionally, create a route-map to allow BGP route announcements:

Without specifying RD and RT attributes the route information will not get into the VPNv4 table.

ESR1

```
ESR1(config)# ip vrf Customer1
ESR1(config-vrf)# ip protocols bgp max-routes 1000
ESR1(config-vrf)# rd 65500:100
ESR1(config-vrf)# route-target import 65500:100
ESR1(config-vrf)# route-target export 65500:100
ESR1(config-vrf)# exit
ESR1(config)# interface gigabitethernet 1/0/2
ESR1(config-if-gi)# ip vrf forwarding Customer1
ESR1(config-if-gi)# description "Customer1"
ESR1(config-if-gi)# ip firewall disable
ESR1(config-if-gi)# ip address 192.168.32.1/30
ESR1(config-if-gi)# exit
ESR1(config)# route-map OUTPUT
ESR1(config-route-map)# rule 1
ESR1(config-route-map-rule)# action permit
ESR1(config-route-map-rule)# exit
ESR1(config-route-map)# exit
ESR1(config)# do commit
ESR1(config)# do confirm
```

ESR3

```
ESR3(config)# ip vrf Customer1
ESR3(config-vrf)# ip protocols bgp max-routes 1000
ESR3(config-vrf)# rd 65500:100
ESR3(config-vrf)# route-target export 65500:100
ESR3(config-vrf)# route-target import 65500:100
ESR3(config-vrf)# exit
ESR3(config)# interface gigabitethernet 1/0/2
ESR3(config-if-gi)# ip vrf forwarding Customer1
ESR3(config-if-gi)# description "Customer1"
ESR3(config-if-gi)# ip firewall disable
ESR3(config-if-gi)# ip address 192.168.32.5/30
ESR3(config-if-gi)# exit
ESR3(config)# route-map OUTPUT
ESR3(config-route-map)# rule 1
ESR3(config-route-map-rule)# action permit
ESR3(config-route-map-rule)# exit
ESR3(config-route-map)# exit
ESR3(config)# do commit
ESR3(config)# do confirm
```

Configure iBGP between ESR1 and ESR3. Enable extended community sending on both devices.

ESR1

```
ESR1(config)# router bgp log-neighbor-changes
ESR1(config)# router bgp 65500
ESR1(config-bgp)# router-id 1.1.1.1
ESR1(config-bgp)# enable
ESR1(config-bgp)# neighbor 3.3.3.3
ESR1(config-bgp-neighbor)# remote-as 65500
ESR1(config-bgp-neighbor)# update-source 1.1.1.1
ESR1(config-bgp-neighbor)# enable
ESR1(config-bgp-neighbor)# address-family vpnv4 unicast
ESR1(config-bgp-neighbor-af)# send-community extended
ESR1(config-bgp-neighbor-af)# enable
ESR1(config-bgp-neighbor-af)# exit
ESR1(config-bgp-neighbor)# exit
ESR1(config-bgp)# exit
ESR1(config)# do commit
ESR1(config)# do confirm
```

ESR3

```
ESR3(config)# router bgp log-neighbor-changes
ESR3(config)# router bgp 65500
ESR3(config-bgp)# router-id 3.3.3.3
ESR3(config-bgp)# enable
ESR3(config-bgp)# neighbor 1.1.1.1
ESR3(config-bgp-neighbor)# remote-as 65500
ESR3(config-bgp-neighbor)# update-source 3.3.3.3
ESR3(config-bgp-neighbor)# enable
ESR3(config-bgp-neighbor)# address-family vpnv4 unicast
ESR3(config-bgp-neighbor-af)# send-community extended
ESR3(config-bgp-neighbor-af)# enable
ESR3(config-bgp-neighbor-af)# exit
ESR3(config-bgp-neighbor)# exit
ESR3(config-bgp)# exit
ESR3(config)# do commit
ESR3(config)# do confirm
```

It is necessary to make sure that BGP session is successfully established.

```
ESR1# show ip bgp neighbors
BGP neighbor is 3.3.3.3
    BGP state: Established
    Neighbor address: 3.3.3.3
    Neighbor AS: 65500
    Neighbor ID: 3.3.3.3
    Neighbor caps: refresh enhanced-refresh restart-aware AS4
    Session: internal multihop AS4
    Source address: 1.1.1.1
    Weight: 0
    Hold timer: 126/180
    Keepalive timer: 40/60
    Address family ipv4 unicast:
    Default originate: No
    Default information originate: No
    Uptime: 88495 s
```

PE-CE routing configuration

According to topology, Customer1 advertises a BGP(AS65505) subnet 10.100.0.0/24. Configure eBGP session between CE_SiteA and PE. Configure the corresponding interfaces, eBGP between ESR1 and CE_SiteA. It is also necessary to allow the announcement of routes towards the PE.

By default the route advertising is prohibited for EBGP, an allow rule must be configured. For IBGP route advertising is allowed.

Configuration on the CE-SiteA router:

CE _SiteA

```
CE-SiteA(config)# interface gigabitethernet 1/0/2
CE-SiteA(config-if-gi)# ip firewall disable
CE-SiteA(config-if-gi)# ip address 192.168.32.2/30
CE-SiteA(config-if-gi)# exit
CE-SiteA(config)# interface loopback 1
CE-SiteA(config-loopback)# ip address 10.100.0.1/24
CE-SiteA(config-loopback)# exit
CE-SiteA(config)# route-map OUTPUT
CE-SiteA(config-route-map)# rule 1
CE-SiteA(config-route-map-rule)# match ip address 10.100.0.0/24
CE-SiteA(config-route-map-rule)# action permit
CE-SiteA(config-route-map-rule)# exit
CE-SiteA(config-route-map)# exit
CE-SiteA(config)# router bgp log-neighbor-changes
CE-SiteA(config)# router bgp 65505
CE-SiteA(config-bgp)# router-id 192.168.32.1
CE-SiteA(config-bgp)# neighbor 192.168.32.1
CE-SiteA(config-bgp-neighbor)# remote-as 65500
CE-SiteA(config-bgp-neighbor)# allow-local-as 1
CE-SiteA(config-bgp-neighbor)# update-source 192.168.32.2
CE-SiteA(config-bgp-neighbor)# address-family ipv4 unicast
CE-SiteA(config-bgp-neighbor-af)# route-map OUTPUT out
CE-SiteA(config-bgp-neighbor-af)# enable
CE-SiteA(config-bgp-neighbor-af)# exit
CE-SiteA(config-bgp-neighbor)# enable
CE-SiteA(config-bgp-neighbor)# exit
CE-SiteA(config-bgp)# address-family ipv4 unicast
CE-SiteA(config-bgp-af)# network 10.100.0.0/24
CE-SiteA(config-bgp-af)# exit
CE-SiteA(config-bgp)# enable
CE-SiteA(config-bgp)# exit
CE-SiteA(config)# do commit
CE-SiteA(config)# do confirm
```

Configure eBGP on the ESR1 router.

Create eBGP session with CE_SiteA and allow routes transmission to the BGP peer:

ESR1

```
ESR1(config)# router bgp 65500
ESR1(config-bgp)# vrf Customer1
ESR1(config-bgp-vrf)# router-id 192.168.32.1
ESR1(config-bgp-vrf)# neighbor 192.168.32.2
ESR1(config-bgp-vrf-neighbor)# remote-as 65505
ESR1(config-bgp-vrf-neighbor)# update-source 192.168.32.1
ESR1(config-bgp-vrf-neighbor)# address-family ipv4 unicast
ESR1(config-bgp-neighbor-af-vrf)# route-map OUTPUT out
ESR1(config-bgp-neighbor-af-vrf)# enable
ESR1(config-bgp-neighbor-af-vrf)# exit
ESR1(config-bgp-vrf-neighbor)# enable
ESR1(config-bgp-vrf-neighbor)# exit
ESR1(config-bgp-vrf)# address-family ipv4 unicast
ESR1(config-bgp-vrf-af)# redistribute connected
ESR1(config-bgp-vrf-af)# redistribute bgp 65500
ESR1(config-bgp-vrf-af)# exit
ESR1(config-bgp-vrf)# enable
ESR1(config-bgp-vrf)# exit
ESR1(config-bgp)# exit
ESR1(config)# do commit
ESR1(config)# do confirm
```

When passing routes from VRF to the VPNv4 table of only connected and/or static networks, you do not need to specify the enable command. Enabling is required only if there are BGP peers in the VRF.

```
ESR1(config)# router bgp 65500
ESR1(config-bgp)# router-id 1.1.1.1
ESR1(config-bgp)# neighbor 3.3.3.3
ESR1(config-bgp-neighbor)# remote-as 65500
ESR1(config-bgp-neighbor)# update-source 1.1.1.1
ESR1(config-bgp-neighbor)# enable
ESR1(config-bgp-neighbor)# address-family vpnv4 unicast
ESR1(config-bgp-neighbor-af)# send-community extended
ESR1(config-bgp-neighbor-af)# enable
ESR1(config-bgp-neighbor-af)# exit
ESR1(config-bgp-neighbor)# exit
ESR1(config-bgp)# enable
EESR1(config-bgp)# vrf Customer1
ESR1(config-bgp-vrf)# address-family ipv4 unicast
ESR1(config-bgp-vrf-af)# redistribute connected
ESR1(config-bgp-vrf-af)# redistribute static
ESR1(config-bgp-vrf-af)# exit
ESR1(config-bgp-vrf)# exit
ESR1(config-bgp)# exit
ESR1(config)# do commit
ESR1(config)# do confirm
```

The following commands can be used to check the accepted and advertised routes:

```
ESR1# show bgp vpnv4 unicast vrf Customer1 neighbors 192.168.32.2 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      Weight Path
*> u 10.100.1.0/24        192.168.32.1                 100                65500 i
*> u 192.168.32.4/30      192.168.32.1                 100                65500 i
```

Display the advertised routes for a specific peer. The route information is displayed after the filtering is applied:

```
ESR1# show bgp vpnv4 unicast vrf Customer1 neighbors 192.168.32.2 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      Weight Path
*> u 10.100.0.0/24        192.168.32.2                 100         0      65505
```

Output received routing information from a specific peer. Route information is displayed after filtering is applied.

CE_SiteB

Perform similar steps between ESR3 and CE_SiteB routers.

Configure the corresponding interfaces and create eBGP session between ESR3 and CE_SiteB:

CE_SiteB

```
CE-SiteB(config)# interface gigabitethernet 1/0/2
CE-SiteB(config-if-gi)# ip firewall disable
CE-SiteB(config-if-gi)# ip address 192.168.32.6/30
CE-SiteB(config-if-gi)# exit
CE-SiteB(config)#
CE-SiteB(config)# interface loopback 1
CE-SiteB(config-loopback)# ip address 10.100.1.1/24
CE-SiteB(config-loopback)# exit
CE-SiteB(config)#
CE-SiteB(config)# route-map OUTPUT
CE-SiteB(config-route-map)# rule 1
CE-SiteB(config-route-map-rule)# match ip address 10.100.1.0/24
CE-SiteB(config-route-map-rule)# action permit
CE-SiteB(config-route-map-rule)# exit
CE-SiteB(config-route-map)# exit
CE-SiteB(config)#
CE-SiteB(config)# router bgp 65505
CE-SiteB(config-bgp)# router-id 192.168.32.6
CE-SiteB(config-bgp)# neighbor 192.168.32.5
CE-SiteB(config-bgp-neighbor)# remote-as 65500
CE-SiteB(config-bgp-neighbor)# allow-local-as 1
CE-SiteB(config-bgp-neighbor)# update-source 192.168.32.6
CE-SiteB(config-bgp-neighbor)# address-family ipv4 unicast
CE-SiteB(config-bgp-neighbor-af)# route-map OUTPUT out
CE-SiteB(config-bgp-neighbor-af)# enable
CE-SiteB(config-bgp-neighbor-af)# exit
CE-SiteB(config-bgp-neighbor)# enable
CE-SiteB(config-bgp-neighbor)# exit
CE-SiteB(config-bgp)# address-family ipv4 unicast
CE-SiteB(config-bgp-af)# network 10.100.1.0/24
CE-SiteB(config-bgp-af)# exit
CE-SiteB(config-bgp)# enable
CE-SiteB(config-bgp)# exit
CE-SiteB(config)# do commit
CE-SiteB(config)# do confirm
```

Configure eBGP for ESR3 and allow transmission of routing information from VRF ro VPNv4 table:

ESR3

```
router bgp 65500
ESR3(config)# router bgp 65500
ESR3(config-bgp)# vrf Customer1
ESR3(config-bgp-vrf)# router-id 192.168.32.5
ESR3(config-bgp-vrf)# neighbor 192.168.32.6
ESR3(config-bgp-vrf-neighbor)# remote-as 65505
ESR3(config-bgp-vrf-neighbor)# update-source 192.168.32.5
ESR3(config-bgp-vrf-neighbor)# address-family ipv4 unicast
ESR3(config-bgp-neighbor-af-vrf)# route-map OUTPUT out
ESR3(config-bgp-neighbor-af-vrf)# enable
ESR3(config-bgp-neighbor-af-vrf)# exit
ESR3(config-bgp-vrf-neighbor)# enable
ESR3(config-bgp-vrf-neighbor)# exit
ESR3(config-bgp-vrf)# address-family ipv4 unicast
ESR3(config-bgp-vrf-af)# redistribute connected
ESR3(config-bgp-vrf-af)# redistribute bgp 65500
ESR3(config-bgp-vrf-af)# exit
ESR3(config-bgp-vrf)# enable
ESR3(config-bgp-vrf)# exit
ESR3(config-bgp)# exit
ESR3(config)# do commit
ESR3(config)# do confirm
```

The following commands can be used to view the VPNv4 table:

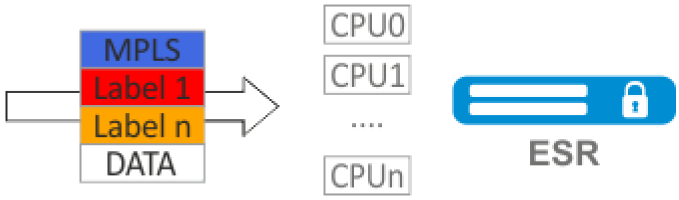
```
ESR1# show ip bgp vpnv4 unicast all
Status codes: * - valid, > - best, i - internal, S - stale
Origin codes: i - IGP, e - EGP, ? - incomplete


Codes Route Distinguisher   IP Prefix          Next hop     Metric     Label   LocPrf     Weight
Path
----- --------------------- ----------------  ------------ ---------- ------- ---------- ------
-----
*>    65500:100             10.100.0.0/24      --           --         23      --         --
    ?
*>i   65500:100             192.168.32.4/30    3.3.3.3      --         84      100        0
      i
*>i   65500:100             10.100.1.0/24      3.3.3.3      --         84      100        0
      i
```

This command outputs all received VPNv4 routes after filtering is applied.

12.8 MPLS traffic balancing

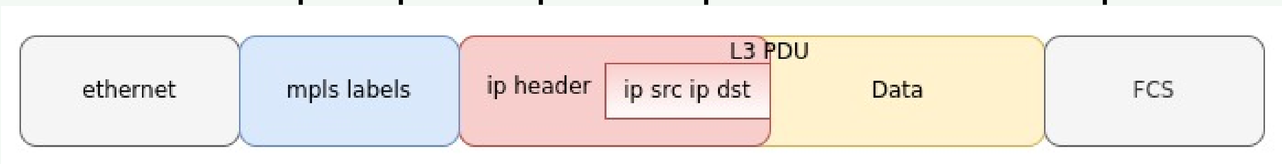
ESR routers have a multi-core architecture. One of the first links in processing incoming traffic is the load balancer daemon (lbd), which performs two main functions:

1. Distributes the load evenly among all router CPUs.
2. Detects abnormal situations with high load on some CPUs, and redistributes processing from these CPUs to less loaded ones.

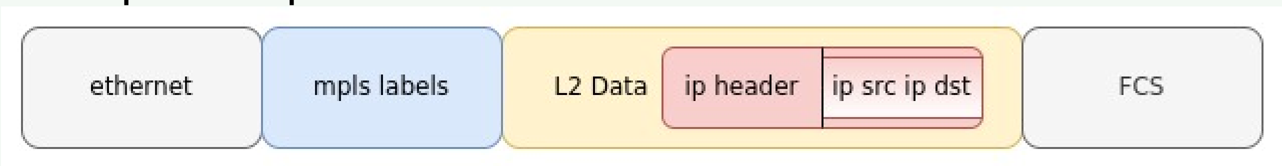


By default, lbd uses only MPLS tags to calculate the hash and then distribute the load to the different CPUs. This behavior is not always an advantage, especially when there are 'large' homogeneous streams of MPLS traffic. Additional functionality can be included to add entropy to the hash:

cpu load-balance mpls passenger ip
Enables possibility to 'look beyond' the MPLS header to find the IP header, and add ip-src and ip-dst to the hash calculation.
For L2VPN: search for a pair of ip-src and ip-dst in the ip header located behind the mpls header.



For L2VPN: ESR tries to look into the ethernet frame (which is behind the mpls header) and get the ip-src and ip-dst in the ip header to add to the hash calculation.



cpu load-balance mpls passenger ip-over-ethernet-pseudowire-with-cw
cpu load-balance mpls passenger ip-over-ethernet-pseudowire-without-cw
Allows to explicitly specify whether Control Word functionality is used when building L2VPN. Allows to prevent an error occurring when a package with Control Word present can be mistakenly recognized as a package without Control Word.

When hashing MPLS labels, the following restrictions apply:

- Labels 0-15 (Special-Purpose Labels) are not added to the calculation – see RFC 7274;
- A label is not added to the calculation if it is immediately preceded by label 15 (Extension Label) – see RFC 7274;
- No more than three tags are added to the hash calculation.

To avoid LDP session crash under heavy load on the CPU of the router on models ESR-200, ESR-1000, ESR-1200, ESR-1500, ESR-1700, after enabling the functionality, all LDP protocol packets will be processed by the control CPUs (Management CPU), which do not participate in traffic processing. For ESR-200, ESR-1000, ESR-1200, ESR-1500 is CPU 0, ESR-1700 is CPU 0-1.

12.8.1 Configuration example

Objective:

Enable L2VPN traffic balancing without using Control Word functionality.

Solution:

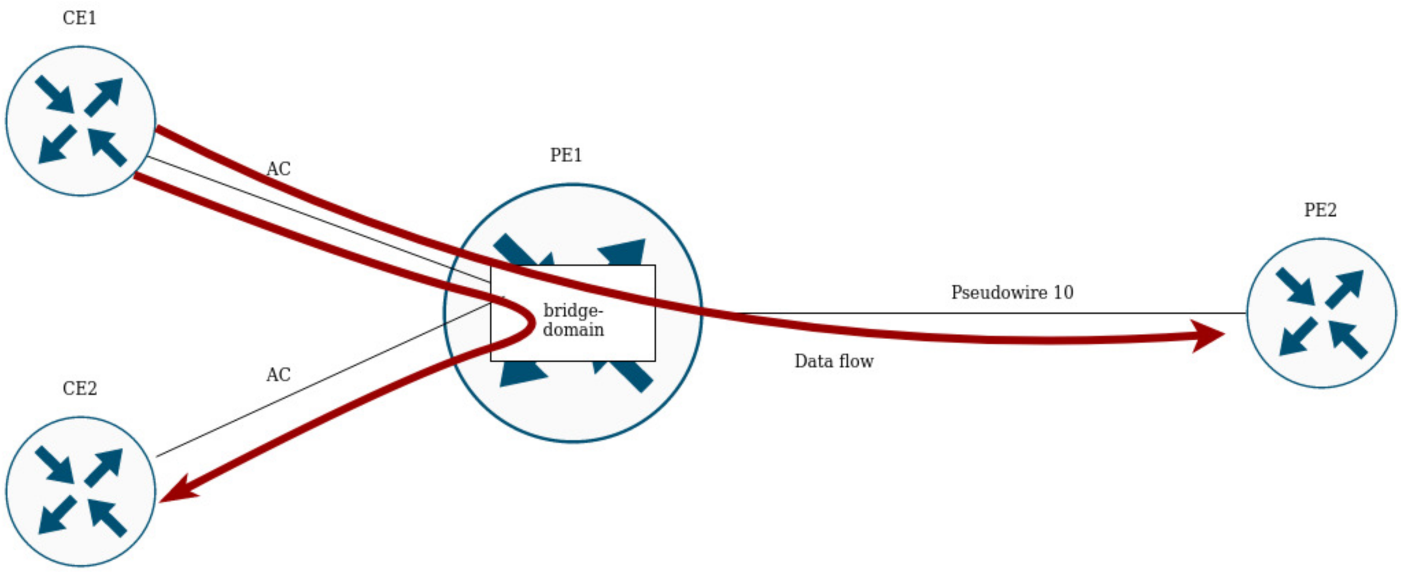
ESR

```
ESR(config)# system cpu load-balance mpls passenger ip
ESR(config)# system cpu load-balance mpls passenger ipoe-pw-without-cw
```

12.9 Operation with the bridge domain within MPLS

To organize L2VPN service, configure a bridge domain on the device, create the required AC, PW (LDP-signaling) and link all the necessary elements with this bridge domain.

For point-to-point, a bridge domain is created automatically.



Traffic is switched between elements of the bridge domain based on the listed rules:

1. A MAC address table is automatically created for each bridge domain, similar to Ethernet switches. Ethernet frames are switched based on analysis of the destination MAC address (DST MAC).
2. Frames with a known DST MAC will be sent to the appropriate AC/PW.
3. Frames with unknown DST MAC, broadcast- and multicast-frames (so called BUM traffic, 'Broadcast, Unknown unicast and Multicast') will be sent to all elements of the bridge domain, except for the element (AC or PW) from which you entered the bridge domain.
4. Switching takes into account the DST MAC in the frames, but does not take into account the VLAN tags present on the frames — thus, switching within a bridge domain is not 'VLAN-aware'.

In the current implementation, the bridge domain does not allow traffic of data link layer protocols such as STP, LLDP, CDP, etc.

The bridge domain can operate in two transport modes: ethernet or vlan. Transport mode sets the rules for handling traffic to and from the bridge domain.

In LDP signaling, ethernet mode (Raw mode, type 5) is used by default. A transport mode can be set for each individual VPLS instance.

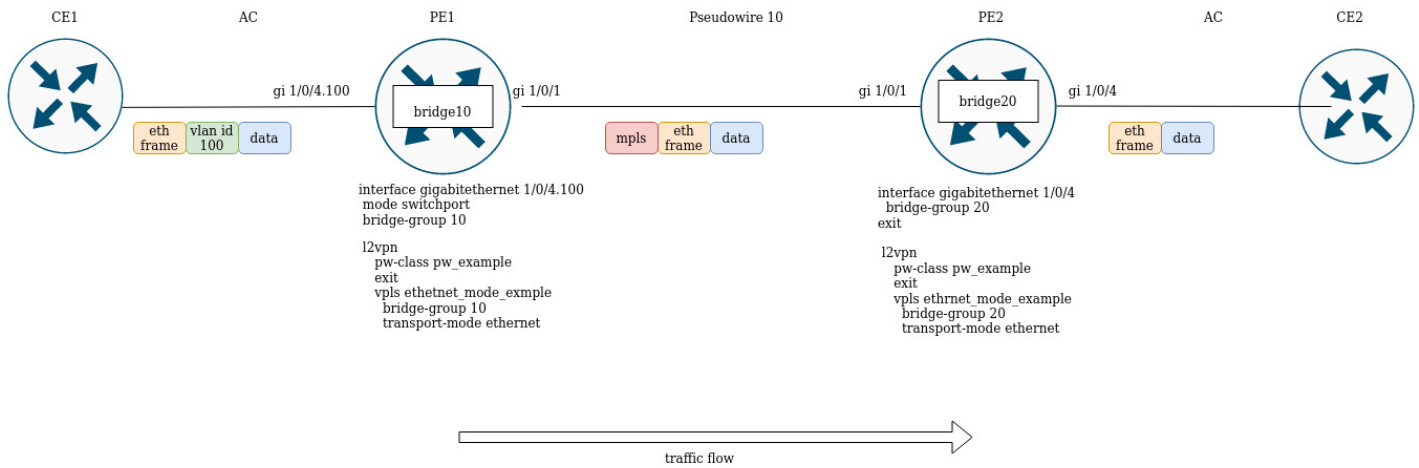
In BGP signaling, the bridge domain only operate in ethernet mode.

```
PE1# config
PE1(config)# mpls
PE1(config-mpls)# l2vpn
PE1(config-l2vpn)# vpls MARTINI_br
PE1(config-l2vpn-vpls)# transport-mode vlan

PE1# sh mpls l2vpn pseudowire
Neighbor                               PW ID      Sig Type       Status
--------------------------------------- ---------- --- ---------- ------
10.10.0.2                              200        LDP Eth Tagged Up
```

In LDP signaling, the transport mode is matched between PEs during pseudowire creation, so it must match on both PEs.

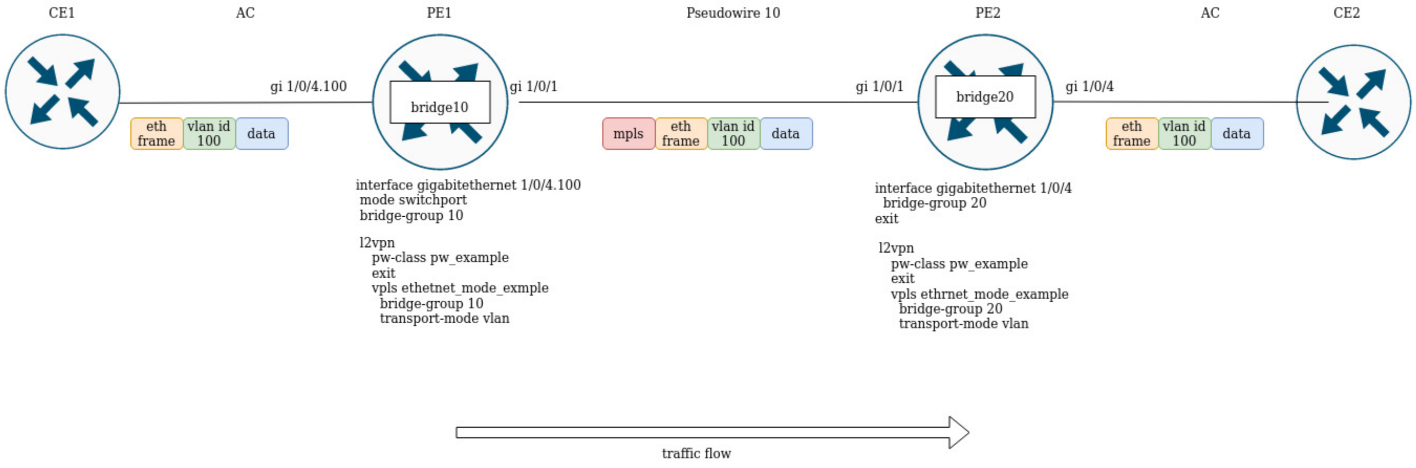
Rules of the traffic processing are given below:

1. Ethernet (Raw) mode:
   - If AC is a subinterface, the vlan tag is removed before putting it in the bridge. Upon leaving the bridge, the vlan tag is restored.
   - If AC is an interface, then tagged and untagged traffic flows in both directions without modification.



PE1 and PE2 are configured in ethernet mode (Figure 2). On the PE1 side, gigabitethernet 1/0/4.100 subinterface is included in the bridge domain, so the vlan tag (vlan id 100) from incoming traffic will be removed before being placed in Pseudowire 10 (respectively, restored when traffic to the AC side). On the other side the AC on PE2, is an interface, which means that traffic will pass through without modification in either direction.

2. Vlan (Tagged) mode:
    - If AC is a subinterface, the vlan tag is saved before putting it in the bridge. The vlan tag can be saved or overwritten depending on the configuration when you exit the bridge.
    - If AC is an interface, traffic modification does not occur in either direction.



12.10 Assignment of MTU when operating with MPLS

It is very important to correctly configure the MTU parameter on the interfaces through which a packet is transmitted. This is true for the installation of the pseudowire and for the transmission of service traffic. There are two important moments:

1. Sizes of the Ethernet header (18 bytes), inner tag (4bytes), outer tag (4 bytes) are not taken into account on the AC interfaces;
2. On the interfaces participating in the forwarding of MPLS traffic, it is necessary to increase the MTU by the number of labels (each label is equal to 4 bytes).

MTU value is involved in signaling when constructing a pseudowire in both LDP-signaling and BGP-signaling. Examples for both cases are given below.

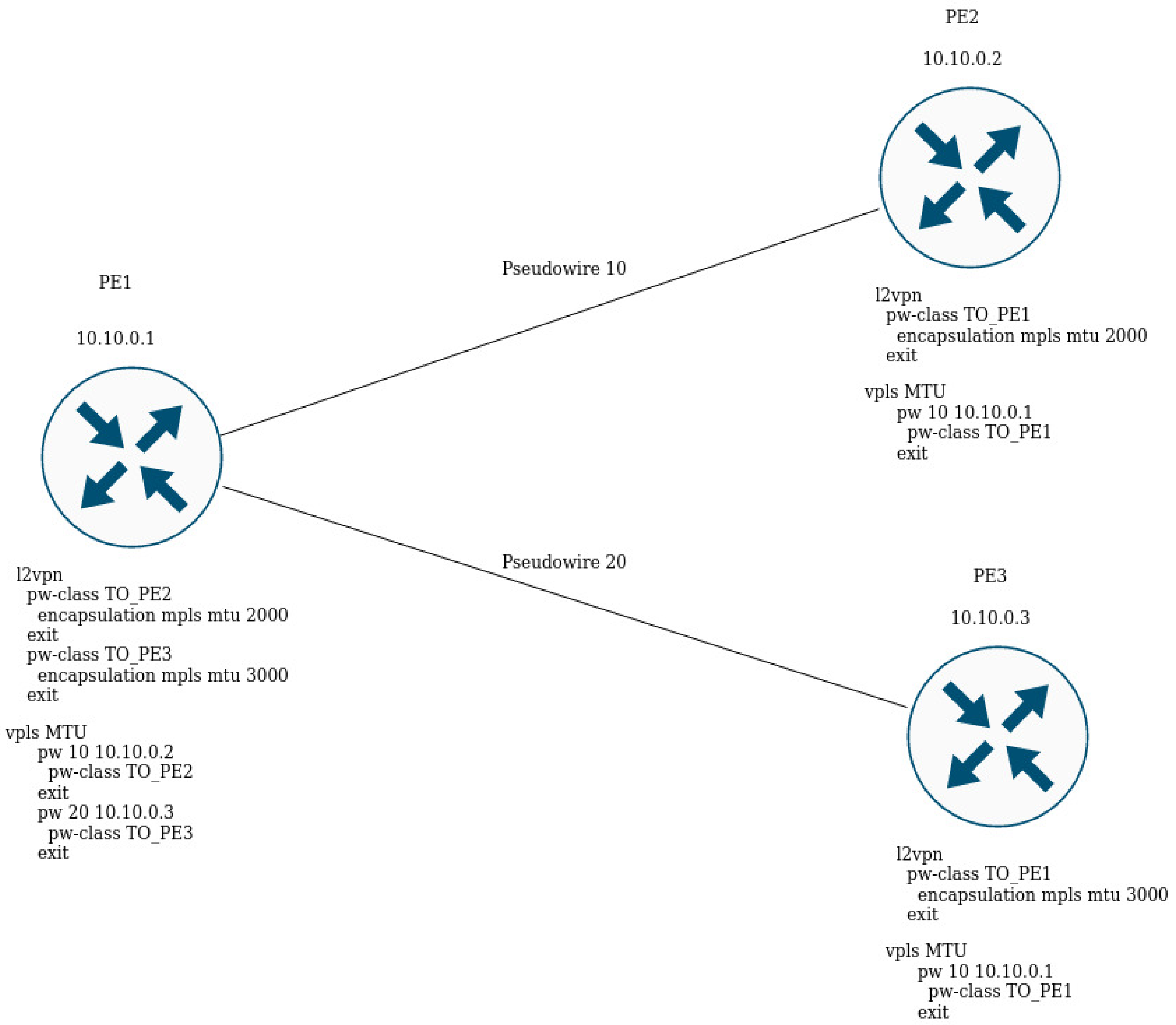
For signaling (LDP, BGP) the default MTU value is 1500.

The MTU values involved in signaling do not affect the actual packet size passing through the pseudowire.

In LDP-signaling, the MTU is set as part of the pw-class setting:

LDP-signaling. Configuration of MTU for matching

```
PE2(config)# mpls
PE2(config-mpls)# l2vpn
PE2(config-l2vpn)# pw-class MTU_example
PE2(config-l2vpn-pw-class)# encapsulation mpls mtu 9000
PE2(config-l2vpn-pw-class)# exit
PE2(config-mpls)# l2vpn
PE2(config-l2vpn)# vpls MTU_Example_PW
PE2(config-l2vpn-vpls)# pw 200 10.10.0.1
PE2(config-l2vpn-pw)# pw-class
PE2(config-l2vpn-pw)# pw-class MTU_example

*View created pw-classes*
PE2#  sh mpls l2vpn pw-class
PW-class                          Neighbor   PW ID      Status Status-tlv MTU
--------------------------------- ---------- ---------- ------ ---------- ------
MTU_example                        10.10.0.1  200        Up     Enable     9000


PE2# sh mpls l2vpn vpls  MTU_Example_PW
VPLS:  MTU_Example_PW
...
    PWs:
      PW ID 2, Neighbor 10.10.0.1:
        MTU:          9000
        Last change: 01:27:42
        Status:       Up

* For PW 2 signaling of this VPLS MTU 9000 is selected*
```



In the figure above, PE1 raises two pseudowires: pseudowire 10 to PE2, and pseudowire 20 to PE3 respectively. For signaling with PE2, the MTU will be set to 2000 (pw-class TO_PE2), for PE3 the MTU will be equal to 3000 (pw-class TO_PE3).

For BGP-signaling, the MTU is specified as part of the l2vpn service configuration:

BGP -signaling. Configuration of MTU for matching

```
PE1(config)# mpls
PE1(config-mpls)# l2vpn
PE1(config-l2vpn)# vpls l2vpn_MTU
PE1(config-l2vpn-vpls)# autodiscovery bgp
PE1(config-bgp)# mtu 1500

PE2# sh mpls l2vpn vpls  l2vpn_MTU
VPLS: l2vpn_MTU
...
    PWs:
      PW ID 2, Neighbor 10.10.0.1:
        MTU:          1500
        Last change: 01:27:42
        Status:       Up

* The MTU 1500 will be selected for signaling all pseudowires of this VPLS *
```

If the MTU value is different when matching, the status of the pseudowire will be 'DOWN', 'Reason : MTU mismatch'.

```
PE1(config-l2vpn)# vpls l2vpn_MTU
PE1(config-l2vpn-vpls)# autodiscovery bgp
PE1(config-bgp)# mtu 2000

PE2# sh mpls l2vpn vpls l2vpn_MTU
...
    PWs:
      PW ID 2, Neighbor 10.10.0.1:
        MTU:          2000
        Last change: 00:00:10
        Status:       Down
        Reason:       MTU mismatch
```

MTU checking for BGP-signaling can be disabled:

```
PE1(config)# mpls
PE1(config-mpls)# l2vpn
PE1(config-l2vpn)# vpls l2vpn _MTU
PE1(config-l2vpn-vpls)# autodiscovery bgp
PE1(config-bgp)# ignore mtu-mismatch
```

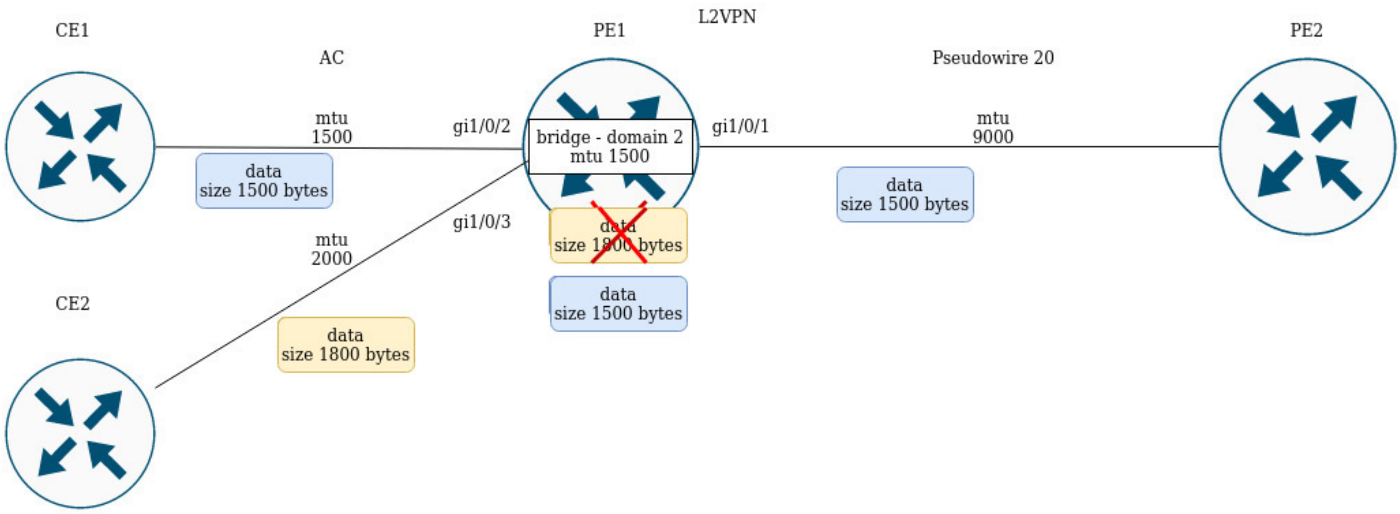
Now, when matching, the MTU value will be ignored.

By default, the bridge domain has an MTU of 1500 bytes. It is worth noting that bridge domain automatically selects the lowest MTU value based on its own MTU and the MTU of the interfaces included in the bridge domain.

```
* E.g., we have a bridge domain 100, which includes interfaces gi1/0/1 with MTU value 2000, and
gi1/0/2 with MTU value 3000 *
CE3(config)# bridge 100
CE3(config-bridge)# enable
CE3(config-bridge)# exit
CE3(config)# interface gigabitethernet 1/0/1
CE3(config-if-gi)# mtu 2000
CE3(config-if-gi)# bridge-group 100
CE3(config-if-gi)# exit
CE3(config)# interface gigabitethernet 1/0/2
CE3(config-if-gi)# mtu 3000
CE3(config-if-gi)# bridge-group 100
CE3(config-if-gi)# do com

* The MTU of the bridge domain will be 1500, since the bridge itself has a default MTU of 1500
 (the default value), which has become the lowest:
 MTU bridge 100 = 1500  <-- The lowest MTU value
 MTU gi1/0/1  = 2000
 MTU gi1/0/2 = 3000
*


CE3# sh interfaces bridge
Bridges      Interfaces
----------   --------------------------------------------------------------
bridge 100   gi1/0/1-2

CE3# sh interfaces status bridge 100
Interface 'bridge 100' status information:
 Description:            --
 Operational state:      UP
 Administrative state:   Up
 Supports broadcast:     Yes
 Supports multicast:     Yes
 MTU:                    1500
 MAC address:            a8:f9:4b:aa:11:00
 Last change:            1 minute and 46 seconds
 Mode:                   Routerport

* Change the MTU on the bridge domain itself: *

CE3(config)# bridge 100
CE3(config-bridge)# mtu 6000
CE3(config-bridge)# do com

* The MTU of the bridge domain became 2000 bytes, because gi1/0/2 has the lowest MTU:
 MTU bridge 100 = 6000
 MTU gi1/0/1  = 2000 <-- The lowest MTU value
 MTU gi1/0/2 = 3000
*

CE3# sh interfaces bridge
Bridges      Interfaces
----------   --------------------------------------------------------------
bridge 100   gi1/0/1-2

CE3# sh interfaces status bridge 100
Interface 'bridge 100' status information:
```

```
Description:            --
Operational state:      Up
Administrative state:   Up
Supports broadcast:     Yes
Supports multicast:     Yes
MTU:                    2000
MAC address:            a8:f9:4b:aa:11:00
Last change:            6 minutes and 42 seconds
Mode:                   Routerport
```

Consider the example of traffic passing through the L2VPN service:

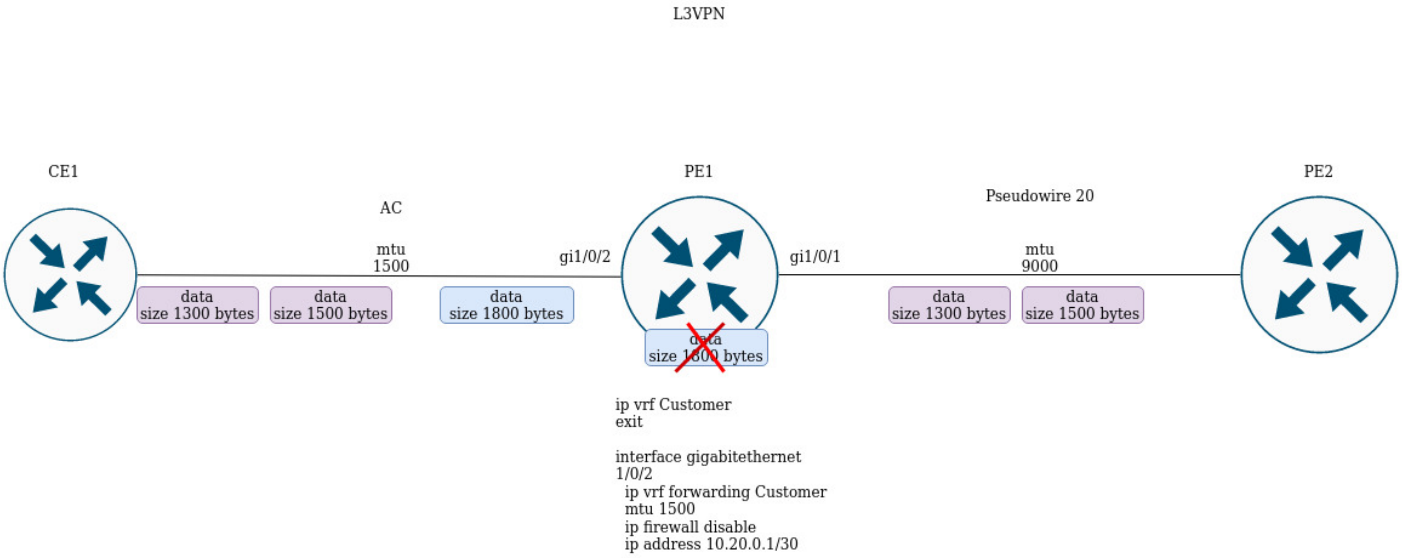


PE1 has the following MTU values on the interfaces:

```
PE1# sh interfaces status
Interface    Admin   Link    MTU     MAC address          Last change                  Mode
             state   state
----------   -----   -----   ------  ------------------   -------------------------
----------
gi1/0/1      Up      Up      9000    a8:f9:4b:ac:4d:16    5 hours, 25 minutes and 2
Routerport
                                                          seconds

gi1/0/2      Up      Up      1500    a8:f9:4b:ac:4d:17    4 days, 4 hours, 49
Switchport
                                                          minutes and 40 seconds

gi1/0/3      Up      Up      1800    a8:f9:4b:ac:4d:18    4 days, 1 hour, 49
Switchport
                                                          minutes and 38 seconds

bridge 2     Up      Up      1500    a8:f9:4b:ac:4d:15    1 day, 1 hour, 27 minutes
Routerport
                                                          and 28 seconds
```

CE1 sends packets of 1500 bytes, CE2 sends packets of 1800 bytes respectively. Since the MTU of the bridge domain is smaller than the MTU of the packet from CE2, the packet from CE2 will be discarded before hitting the bridge domain. Similar will be the case if the MTU interface facing the mpls-core (gi1/0/1) is less than the MTU coming from the CE packets (taking into account the mpls header).

Similar behavior when passing traffic in the L3VPN service:

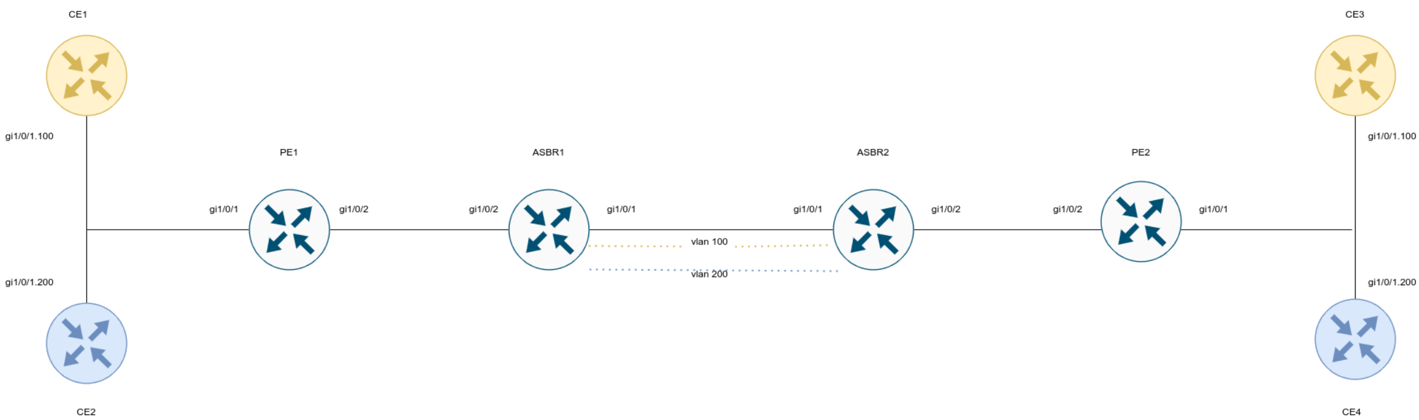


If CE1 sends a packet with a higher MTU than on the interface facing the client (gi1/0/2) or towards the mpls-core (gi1/0/1), the packet will be discarded.

12.11 Inter-AS Option A

This section provides examples of configuration based on the construction of l3vpn and l2vpn services. The main feature of inter-AS Option A is the absence of mpls-tags in traffic when transferring between ABSR. To separate client service traffic between ASBRs, VRF is usually used for l3vpn or tagging (dot1q, q-in-q) for l2vpn services.

12.11.1 L2VPN



CE configuration:

CE1

```
ESR# config
ESR(config)# hostname CE1
ESR(config)# interface gigabitethernet 1/0/1.100
ESR(config-if-gi)# ip firewall disable
ESR(config-if-gi)# ip address 192.168.1.1/24
ESR(config-if-gi)# do com
ESR(config-if-gi)# do conf
```

CE2

```
ESR# config
ESR(config)# hostname CE2
ESR(config)# interface gigabitethernet 1/0/1.200
ESR(config-if-gi)# ip firewall disable
ESR(config-if-gi)# ip address 192.168.2.1/24
ESR(config-if-gi)# do com
ESR(config-if-gi)# do conf
```

CE3

```
ESR# config
ESR(config)# hostname CE3
ESR(config)# interface gigabitethernet 1/0/1.100
ESR(config-if-gi)# ip firewall disable
ESR(config-if-gi)# ip address 192.168.1.2/24
ESR(config-if-gi)# do com
ESR(config-if-gi)# do conf
```

CE4

```
ESR# config
ESR(config)# hostname CE4
ESR(config)# interface gigabitethernet 1/0/1.200
ESR(config-if-gi)# ip firewall disable
ESR(config-if-gi)# ip address 192.168.2.2/24
ESR(config-if-gi)# do com
ESR(config-if-gi)# do conf
```

Configure PE1 and PE2. Assign service labels advertising to BGP (Kompella mode):

PE1

```
ESR(config)# hostname PE1
ESR(config)# system jumbo-frames
ESR(config)# router bgp log-neighbor-changes
ESR(config)# router bgp 65500
ESR(config-bgp)#   neighbor 10.10.1.2
ESR(config-bgp-neighbor)#     remote-as 65500
ESR(config-bgp-neighbor)#     update-source 10.10.1.1
ESR(config-bgp-neighbor)#     address-family l2vpn vpls
ESR(config-bgp-neighbor-af)#       send-community extended
ESR(config-bgp-neighbor-af)#       enable
ESR(config-bgp-neighbor-af)#     exit
ESR(config-bgp-neighbor)#     enable
ESR(config-bgp-neighbor)#   exit
ESR(config-bgp)#   enable
ESR(config-bgp)# exit
ESR(config)#
ESR(config)# router ospf 1
ESR(config-ospf)#   area 0.0.0.0
ESR(config-ospf-area)#     enable
ESR(config-ospf-area)#   exit
ESR(config-ospf)#   enable
ESR(config-ospf)# exit
ESR(config)#
ESR(config)# bridge 100
ESR(config-bridge)#   enable
ESR(config-bridge)# exit
ESR(config)# bridge 200
ESR(config-bridge)#   enable
ESR(config-bridge)# exit
ESR(config)#
ESR(config)# interface gigabitethernet 1/0/1.100
ESR(config-subif)#   description "to CE1"
ESR(config-subif)#   bridge-group 100
ESR(config-subif)# exit
ESR(config)# interface gigabitethernet 1/0/1.200
ESR(config-subif)#   description "to CE2"
ESR(config-subif)#   bridge-group 200
ESR(config-subif)# exit
ESR(config)# interface gigabitethernet 1/0/2
ESR(config-if-gi)#   mtu 1522
ESR(config-if-gi)#   ip firewall disable
ESR(config-if-gi)#   ip address 10.100.0.1/30
ESR(config-if-gi)#   ip ospf instance 1
ESR(config-if-gi)#   ip ospf
ESR(config-if-gi)# exit
ESR(config)# interface loopback 1
ESR(config-loopback)#   ip address 10.10.1.1/32
ESR(config-loopback)#   ip ospf instance 1
ESR(config-loopback)#   ip ospf
ESR(config-loopback)# exit
ESR(config)# mpls
ESR(config-mpls)#   ldp
ESR(config-ldp)#     router-id 10.10.1.1
ESR(config-ldp)#     address-family ipv4
ESR(config-ldp-af-ipv4)#       interface gigabitethernet 1/0/2
ESR(config-ldp-af-ipv4-if)#       exit
```

```
ESR(config-ldp-af-ipv4)#     exit
ESR(config-ldp)#     enable
ESR(config-ldp)#   exit
ESR(config-mpls)#   l2vpn
ESR(config-l2vpn)#     vpls CE1
ESR(config-l2vpn-vpls)#       bridge-group 100
ESR(config-l2vpn-vpls)#       autodiscovery bgp
ESR(config-bgp)#         vpn id 1
ESR(config-bgp)#         ve id 2
ESR(config-bgp)#         rd 65500:1
ESR(config-bgp)#         route-target export 65500:1
ESR(config-bgp)#         route-target import 65500:1
ESR(config-bgp)#       exit
ESR(config-l2vpn-vpls)#       enable
ESR(config-l2vpn-vpls)#     exit
ESR(config-l2vpn)#     vpls CE2
ESR(config-l2vpn-vpls)#       bridge-group 200
ESR(config-l2vpn-vpls)#       autodiscovery bgp
ESR(config-bgp)#         vpn id 2
ESR(config-bgp)#         ve id 2
ESR(config-bgp)#         rd 65500:2
ESR(config-bgp)#         route-target export 65500:2
ESR(config-bgp)#         route-target import 65500:2
ESR(config-bgp)#       exit
ESR(config-l2vpn-vpls)#       enable
ESR(config-l2vpn-vpls)#     exit
ESR(config-l2vpn)#   exit
ESR(config-mpls)#   forwarding interface gigabitethernet 1/0/2
ESR(config-mpls)# exit
ESR(config)# do com
ESR(config)# do conf
```

PE2

```
ESR(config)# hostname ESR
ESR(config)# system jumbo-frames
ESR(config)#
ESR(config)# router bgp log-neighbor-changes
ESR(config)# router bgp 65500
ESR(config-bgp)#   router-id 10.11.1.1
ESR(config-bgp)#   neighbor 10.11.1.2
ESR(config-bgp-neighbor)#     remote-as 65500
ESR(config-bgp-neighbor)#     update-source 10.11.1.1
ESR(config-bgp-neighbor)#     address-family l2vpn vpls
ESR(config-bgp-neighbor-af)#       send-community extended
ESR(config-bgp-neighbor-af)#       enable
ESR(config-bgp-neighbor-af)#     exit
ESR(config-bgp-neighbor)#     enable
ESR(config-bgp-neighbor)#   exit
ESR(config-bgp)#   enable
ESR(config-bgp)# exit
ESR(config)#
ESR(config)# router ospf 1
ESR(config-ospf)#   area 0.0.0.0
ESR(config-ospf-area)#     enable
ESR(config-ospf-area)#   exit
ESR(config-ospf)#   enable
ESR(config-ospf)# exit
ESR(config)#
ESR(config)# bridge 100
ESR(config-bridge)#   enable
ESR(config-bridge)# exit
ESR(config)# bridge 200
ESR(config-bridge)#   enable
ESR(config-bridge)# exit
ESR(config)#
ESR(config)# interface gigabitethernet 1/0/1.100
ESR(config-subif)#   description "to CE3"
ESR(config-subif)#   bridge-group 100
ESR(config-subif)# exit
ESR(config)# interface gigabitethernet 1/0/1.200
ESR(config-subif)#   description "to CE4"
ESR(config-subif)#   bridge-group 200
ESR(config-subif)# exit
ESR(config)# interface gigabitethernet 1/0/2
ESR(config-if-gi)#   mtu 1522
ESR(config-if-gi)#   ip firewall disable
ESR(config-if-gi)#   ip address 10.101.0.1/30
ESR(config-if-gi)#   ip ospf instance 1
ESR(config-if-gi)#   ip ospf
ESR(config-if-gi)# exit
ESR(config)# interface loopback 1
ESR(config-loopback)#   ip address 10.11.1.1/32
ESR(config-loopback)#   ip ospf instance 1
ESR(config-loopback)#   ip ospf
ESR(config-loopback)# exit
ESR(config)# mpls
ESR(config-mpls)#   ldp
ESR(config-ldp)#     router-id 10.11.1.1
ESR(config-ldp)#     address-family ipv4
```

```
ESR(config-ldp-af-ipv4)#       interface gigabitethernet 1/0/2
ESR(config-ldp-af-ipv4-if)#       exit
ESR(config-ldp-af-ipv4)#     exit
ESR(config-ldp)#     enable
ESR(config-ldp)#   exit
ESR(config-mpls)#   l2vpn
ESR(config-l2vpn)#     vpls CE1
ESR(config-l2vpn-vpls)#       bridge-group 100
ESR(config-l2vpn-vpls)#       autodiscovery bgp
ESR(config-bgp)#         vpn id 1
ESR(config-bgp)#         ve id 2
ESR(config-bgp)#         rd 65500:1
ESR(config-bgp)#         route-target export 65500:1
ESR(config-bgp)#         route-target import 65500:1
ESR(config-bgp)#       exit
ESR(config-l2vpn-vpls)#       enable
ESR(config-l2vpn-vpls)#     exit
ESR(config-l2vpn)#     vpls CE2
ESR(config-l2vpn-vpls)#       bridge-group 200
ESR(config-l2vpn-vpls)#       autodiscovery bgp
ESR(config-bgp)#         vpn id 2
ESR(config-bgp)#         ve id 2
ESR(config-bgp)#         rd 65500:2
ESR(config-bgp)#         route-target export 65500:2
ESR(config-bgp)#         route-target import 65500:2
ESR(config-bgp)#       exit
ESR(config-l2vpn-vpls)#       enable
ESR(config-l2vpn-vpls)#     exit
ESR(config-l2vpn)#   exit
ESR(config-mpls)#   forwarding interface gigabitethernet 1/0/2
ESR(config-mpls)# exit
ESR(config)# do com
ESR(config)# do conf
```

Configure ASBR1 and ASBR2. To separate traffic from CE1 and CE2 towards ASBR2, make the gi1/0/1 interface as trunk. Vlan 100 and 200 will be dedicated to traffic from CE1 and CE2:

ASBR1

```
ESR(config)#  hostname ASBR1
ESR(config)#
ESR(config)# system jumbo-frames
ESR(config)#
ESR(config)# vlan 100,200
ESR(config-vlan)# exit
ESR(config)#
ESR(config)# router bgp 65500
ESR(config-bgp)#   router-id 10.10.1.2
ESR(config-bgp)#   neighbor 10.10.1.1
ESR(config-bgp-neighbor)#     remote-as 65500
ESR(config-bgp-neighbor)#     update-source 10.10.1.2
ESR(config-bgp-neighbor)#     address-family l2vpn vpls
ESR(config-bgp-neighbor-af)#       send-community extended
ESR(config-bgp-neighbor-af)#       enable
ESR(config-bgp-neighbor-af)#     exit
ESR(config-bgp-neighbor)#     enable
ESR(config-bgp-neighbor)#   exit
ESR(config-bgp)#   enable
ESR(config-bgp)# exit
ESR(config)#
ESR(config)# router ospf 1
ESR(config-ospf)#   area 0.0.0.0
ESR(config-ospf-area)#     enable
ESR(config-ospf-area)#   exit
ESR(config-ospf)#   enable
ESR(config-ospf)# exit
ESR(config)#
ESR(config)# bridge 10
ESR(config-bridge)#   vlan 100
ESR(config-bridge)#   enable
ESR(config-bridge)# exit
ESR(config)# bridge 20
ESR(config-bridge)#   vlan 200
ESR(config-bridge)#   enable
ESR(config-bridge)# exit
ESR(config)#
ESR(config)# interface gigabitethernet 1/0/1
ESR(config-if-gi)#   description "to ASBR2"
ESR(config-if-gi)#   mode switchport
ESR(config-if-gi)#   spanning-tree disable
ESR(config-if-gi)#   switchport forbidden default-vlan
ESR(config-if-gi)#   switchport mode trunk
ESR(config-if-gi)#   switchport trunk allowed vlan add 100,200
ESR(config-if-gi)# exit
ESR(config)# interface gigabitethernet 1/0/2
ESR(config-if-gi)#   description "to PE1"
ESR(config-if-gi)#   mtu 1522
ESR(config-if-gi)#   ip firewall disable
ESR(config-if-gi)#   ip address 10.100.0.2/30
ESR(config-if-gi)#   ip ospf instance 1
ESR(config-if-gi)#   ip ospf
ESR(config-if-gi)# exit
ESR(config)# interface loopback 1
ESR(config-loopback)#   ip address 10.10.1.2/32
ESR(config-loopback)#   ip ospf instance 1
```

```
ESR(config-loopback)#   ip ospf
ESR(config-loopback)# exit
ESR(config)# mpls
ESR(config-mpls)#   ldp
ESR(config-ldp)#     router-id 10.10.1.2
ESR(config-ldp)#     address-family ipv4
ESR(config-ldp-af-ipv4)#       interface gigabitethernet 1/0/2
ESR(config-ldp-af-ipv4-if)#       exit
ESR(config-ldp-af-ipv4)#     exit
ESR(config-ldp)#     enable
ESR(config-ldp)#   exit
ESR(config-mpls)#   l2vpn
ESR(config-l2vpn)#     vpls CE1
ESR(config-l2vpn-vpls)#       bridge-group 10
ESR(config-l2vpn-vpls)#       autodiscovery bgp
ESR(config-bgp)#         vpn id 1
ESR(config-bgp)#         ve id 1
ESR(config-bgp)#         rd 65500:1
ESR(config-bgp)#         route-target export 65500:1
ESR(config-bgp)#         route-target import 65500:1
ESR(config-bgp)#       exit
ESR(config-l2vpn-vpls)#       enable
ESR(config-l2vpn-vpls)#     exit
ESR(config-l2vpn)#     vpls CE2
ESR(config-l2vpn-vpls)#       bridge-group 20
ESR(config-l2vpn-vpls)#       autodiscovery bgp
ESR(config-bgp)#         vpn id 2
ESR(config-bgp)#         ve id 1
ESR(config-bgp)#         rd 65500:2
ESR(config-bgp)#         route-target export 65500:2
ESR(config-bgp)#         route-target import 65500:2
ESR(config-bgp)#       exit
ESR(config-l2vpn-vpls)#       enable
ESR(config-l2vpn-vpls)#     exit
ESR(config-l2vpn)#   exit
ESR(config-mpls)#   forwarding interface gigabitethernet 1/0/2
ESR(config-mpls)# exit
ESR(config)# do com
ESR(config)# do conf
```

ASBR2

```
ESR(config)#  hostname ASBR2
ESR(config)#
ESR(config)# system jumbo-frames
ESR(config)#
ESR(config)# vlan 100,200
ESR(config-vlan)# exit
ESR(config)#
ESR(config)# router bgp 65500
ESR(config-bgp)#   router-id 10.10.1.2
ESR(config-bgp)#   neighbor 10.10.1.1
ESR(config-bgp-neighbor)#     remote-as 65500
ESR(config-bgp-neighbor)#     update-source 10.10.1.2
ESR(config-bgp-neighbor)#     address-family l2vpn vpls
ESR(config-bgp-neighbor-af)#       send-community extended
ESR(config-bgp-neighbor-af)#       enable
ESR(config-bgp-neighbor-af)#     exit
ESR(config-bgp-neighbor)#     enable
ESR(config-bgp-neighbor)#   exit
ESR(config-bgp)#   enable
ESR(config-bgp)# exit
ESR(config)#
ESR(config)# router ospf 1
ESR(config-ospf)#   area 0.0.0.0
ESR(config-ospf-area)#     enable
ESR(config-ospf-area)#   exit
ESR(config-ospf)#   enable
ESR(config-ospf)# exit
ESR(config)#
ESR(config)# bridge 10
ESR(config-bridge)#   vlan 100
ESR(config-bridge)#   enable
ESR(config-bridge)# exit
ESR(config)# bridge 20
ESR(config-bridge)#   vlan 200
ESR(config-bridge)#   enable
ESR(config-bridge)# exit
ESR(config)#
ESR(config)# interface gigabitethernet 1/0/1
ESR(config-if-gi)#   description "to ASBR1"
ESR(config-if-gi)#   mode switchport
ESR(config-if-gi)#   spanning-tree disable
ESR(config-if-gi)#   switchport forbidden default-vlan
ESR(config-if-gi)#   switchport mode trunk
ESR(config-if-gi)#   switchport trunk allowed vlan add 100,200
ESR(config-if-gi)# exit
ESR(config)# interface gigabitethernet 1/0/2
ESR(config-if-gi)#   description "to PE1"
ESR(config-if-gi)#   mtu 1522
ESR(config-if-gi)#   ip firewall disable
ESR(config-if-gi)#   ip address 10.100.0.2/30
ESR(config-if-gi)#   ip ospf instance 1
ESR(config-if-gi)#   ip ospf
ESR(config-if-gi)# exit
ESR(config)# interface loopback 1
ESR(config-loopback)#   ip address 10.10.1.2/32
ESR(config-loopback)#   ip ospf instance 1
```

```
ESR(config-loopback)#   ip ospf
ESR(config-loopback)# exit
ESR(config)# mpls
ESR(config-mpls)#   ldp
ESR(config-ldp)#     router-id 10.10.1.2
ESR(config-ldp)#     address-family ipv4
ESR(config-ldp-af-ipv4)#      interface gigabitethernet 1/0/2
ESR(config-ldp-af-ipv4-if)#       exit
ESR(config-ldp-af-ipv4)#     exit
ESR(config-ldp)#     enable
ESR(config-ldp)#   exit
ESR(config-mpls)#   l2vpn
ESR(config-l2vpn)#     vpls CE1
ESR(config-l2vpn-vpls)#       bridge-group 10
ESR(config-l2vpn-vpls)#       autodiscovery bgp
ESR(config-bgp)#         vpn id 1
ESR(config-bgp)#         ve id 1
ESR(config-bgp)#         rd 65500:1
ESR(config-bgp)#         route-target export 65500:1
ESR(config-bgp)#         route-target import 65500:1
ESR(config-bgp)#       exit
ESR(config-l2vpn-vpls)#       enable
ESR(config-l2vpn-vpls)#     exit
ESR(config-l2vpn)#     vpls CE2
ESR(config-l2vpn-vpls)#       bridge-group 20
ESR(config-l2vpn-vpls)#       autodiscovery bgp
ESR(config-bgp)#         vpn id 2
ESR(config-bgp)#         ve id 1
ESR(config-bgp)#         rd 65500:2
ESR(config-bgp)#         route-target export 65500:2
ESR(config-bgp)#         route-target import 65500:2
ESR(config-bgp)#       exit
ESR(config-l2vpn-vpls)#       enable
ESR(config-l2vpn-vpls)#     exit
ESR(config-l2vpn)#   exit
ESR(config-mpls)#   forwarding interface gigabitethernet 1/0/2
ESR(config-mpls)# exit
ESR(config)# do com
ESR(config)# do conf
```

Check label assignment, service status, and network availability between CEs:

Labels information

```
ASBR2# sh 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   65500:1               2     1     10    10.11.1.1       --         100        0      i
*>i   65500:2               2     1     10    10.11.1.1       --         100        0      i
*>    65500:1               1     1     10    --              --         --         --
*>    65500:2               1     1     10    --              --         --         --

ASBR2# sh mpls forwarding-table
Local    Outgoing Prefix                                    Outgoing         Next Hop
label    label    or tunnel ID                              Interface
-------- -------- ----------------------------------------- ---------------- -----------
56       imp-null 10.11.1.1/32                              gi1/0/2          10.101.0.1
47       37       PW ID 1                                   --               10.11.1.1
37       47       PW ID 2                                   --               10.11.1.1
```

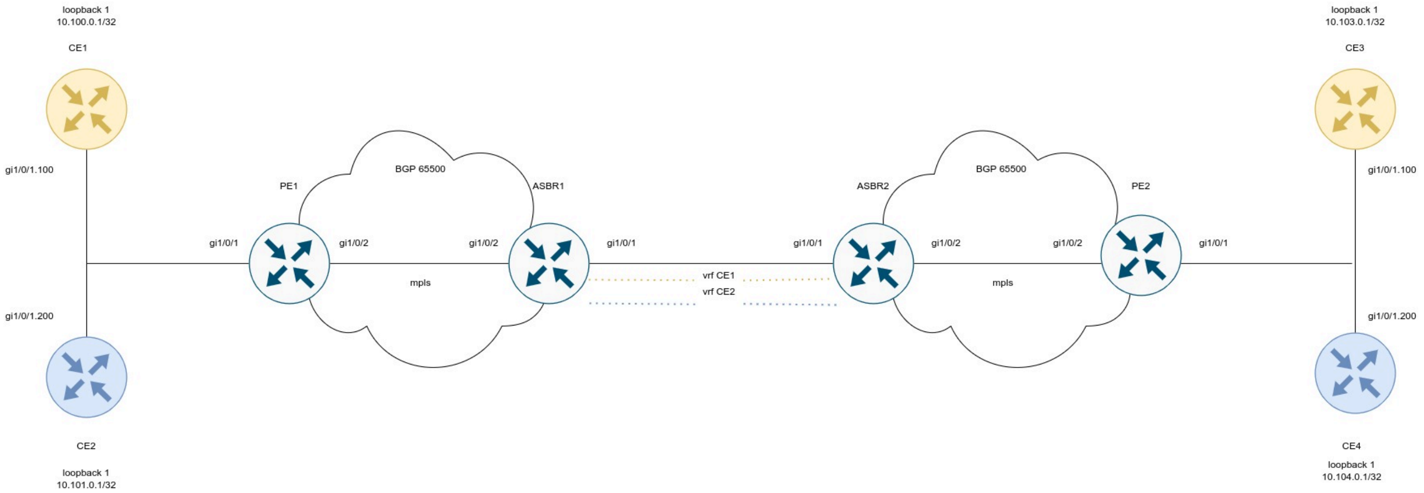
Status of services

```
ASBR2# sh mpls l2vpn vpls
VPLS: CE1
    bridge 10:
        MTU:    1500
        Status: Up
    PWs:
      PW ID 1, Neighbor 10.11.1.1:
        MTU:         1500
        Last change: 00:16:59
        Status:      Up
VPLS: CE2
    bridge 20:
        MTU:    1500
        Status: Up
    PWs:
      PW ID 2, Neighbor 10.11.1.1:
        MTU:         1500
        Last change: 00:16:59
        Status:      Up
```

Checking network availability

```
CE1# ping 192.168.1.2 detailed
PING 192.168.1.2 (192.168.1.2) 56 bytes of data.
64 bytes from 192.168.1.2: icmp_seq=1 ttl=0 time=1.08 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=0 time=1.06 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=0 time=1.01 ms
64 bytes from 192.168.1.2: icmp_seq=4 ttl=0 time=0.971 ms
64 bytes from 192.168.1.2: icmp_seq=5 ttl=0 time=0.972 ms

CE2# ping 192.168.2.2 detailed  packets
PING 192.168.2.2 (192.168.2.2) 56 bytes of data.
64 bytes from 192.168.2.2: icmp_seq=1 ttl=0 time=1.17 ms
64 bytes from 192.168.2.2: icmp_seq=2 ttl=0 time=0.972 ms
64 bytes from 192.168.2.2: icmp_seq=3 ttl=0 time=0.960 ms
64 bytes from 192.168.2.2: icmp_seq=4 ttl=0 time=1.04 ms
64 bytes from 192.168.2.2: icmp_seq=5 ttl=0 time=0.976 ms

ASBR2# sh mac address-table bridge 10
VID     MAC Address          Interface                        Type
-----   ------------------   ------------------------------   -------
--      e4:5a:d4:01:b9:73    vlan 100                         Dynamic
--      e4:5a:d4:a1:34:61    dypseudowire 1_10.11.1.1         Dynamic
2 valid mac entries
ASBR2# sh mac address-table bridge 20
VID     MAC Address          Interface                        Type
-----   ------------------   ------------------------------   -------
--      e4:5a:d4:01:c1:80    vlan 200                         Dynamic
--      e4:5a:d4:a1:34:61    dypseudowire 2_10.11.1.1         Dynamic
2 valid mac entries
```

12.11.2 L3VPN



Configure CE:

CE1

```
ESR(config)# hostname CE1
ESR(config)#
ESR(config)# route-map BGP
ESR(config-route-map)#   rule 1
ESR(config-route-map-rule)#   exit
ESR(config-route-map)# exit
ESR(config)# router bgp 65501
ESR(config-bgp)#   neighbor 192.168.1.2
ESR(config-bgp-neighbor)#     remote-as 65500
ESR(config-bgp-neighbor)#     address-family ipv4 unicast
ESR(config-bgp-neighbor-af)#       route-map BGP out
ESR(config-bgp-neighbor-af)#       enable
ESR(config-bgp-neighbor-af)#     exit
ESR(config-bgp-neighbor)#     enable
ESR(config-bgp-neighbor)#   exit
ESR(config-bgp)#   address-family ipv4 unicast
ESR(config-bgp-af)#     network 10.110.0.1/32
ESR(config-bgp-af)#   exit
ESR(config-bgp)#   enable
ESR(config-bgp)# exit
ESR(config)# interface gigabitethernet 1/0/1.100
ESR(config-subif)#   ip firewall disable
ESR(config-subif)#   ip address 192.168.1.1/30
ESR(config-subif)# exit
ESR(config)# interface loopback 1
ESR(config-loopback)#   ip address 10.110.0.1/32
ESR(config-loopback)# exit
ESR(config)# do com
ESR(config)# do conf
```

CE2

```
ESR(config)# hostname CE2
ESR(config)#
ESR(config)# route-map BGP
ESR(config-route-map)#   rule 1
ESR(config-route-map-rule)#   exit
ESR(config-route-map)# exit
ESR(config)# router bgp 65501
ESR(config-bgp)#   neighbor 192.168.2.2
ESR(config-bgp-neighbor)#     remote-as 65500
ESR(config-bgp-neighbor)#     address-family ipv4 unicast
ESR(config-bgp-neighbor-af)#       route-map BGP out
ESR(config-bgp-neighbor-af)#       enable
ESR(config-bgp-neighbor-af)#     exit
ESR(config-bgp-neighbor)#     enable
ESR(config-bgp-neighbor)#   exit
ESR(config-bgp)#   address-family ipv4 unicast
ESR(config-bgp-af)#     network 10.112.0.1/32
ESR(config-bgp-af)#   exit
ESR(config-bgp)#   enable
ESR(config-bgp)# exit
ESR(config)# interface gigabitethernet 1/0/1.100
ESR(config-subif)#   ip firewall disable
ESR(config-subif)#   ip address 192.168.2.1/30
ESR(config-subif)# exit
ESR(config)#
ESR(config)# interface loopback 1
ESR(config-loopback)#   ip address 10.112.0.1/32
ESR(config-loopback)# exit
ESR(config)# do com
ESR(config)# do conf
```

CE3

```
ESR(config)# hostname CE3
ESR(config)#
ESR(config)# route-map BGP
ESR(config-route-map)#   rule 1
ESR(config-route-map-rule)#   exit
ESR(config-route-map)# exit
ESR(config)# router bgp 65501
ESR(config-bgp)#   neighbor 192.168.3.2
ESR(config-bgp-neighbor)#     remote-as 65500
ESR(config-bgp-neighbor)#     address-family ipv4 unicast
ESR(config-bgp-neighbor-af)#       route-map BGP out
ESR(config-bgp-neighbor-af)#       enable
ESR(config-bgp-neighbor-af)#     exit
ESR(config-bgp-neighbor)#     enable
ESR(config-bgp-neighbor)#   exit
ESR(config-bgp)#   address-family ipv4 unicast
ESR(config-bgp-af)#     network 10.113.0.1/32
ESR(config-bgp-af)#   exit
ESR(config-bgp)#   enable
ESR(config-bgp)# exit
ESR(config)# interface gigabitethernet 1/0/1.100
ESR(config-subif)#   ip firewall disable
ESR(config-subif)#   ip address 192.168.3.1/30
ESR(config-subif)# exit
ESR(config)#
ESR(config)# interface loopback 1
ESR(config-loopback)#   ip address 10.113.0.1/32
ESR(config-loopback)# exit
ESR(config)# do com
ESR(config)# do conf
```

CE4

```
ESR(config)# hostname CE4
ESR(config)#
ESR(config)# route-map BGP
ESR(config-route-map)#   rule 1
ESR(config-route-map-rule)#   exit
ESR(config-route-map)# exit
ESR(config)# router bgp 65501
ESR(config-bgp)#   neighbor 192.168.4.2
ESR(config-bgp-neighbor)#     remote-as 65500
ESR(config-bgp-neighbor)#     address-family ipv4 unicast
ESR(config-bgp-neighbor-af)#       route-map BGP out
ESR(config-bgp-neighbor-af)#       enable
ESR(config-bgp-neighbor-af)#     exit
ESR(config-bgp-neighbor)#     enable
ESR(config-bgp-neighbor)#   exit
ESR(config-bgp)#   address-family ipv4 unicast
ESR(config-bgp-af)#     network 10.114.0.1/32
ESR(config-bgp-af)#   exit
ESR(config-bgp)#   enable
ESR(config-bgp)# exit
ESR(config)# interface gigabitethernet 1/0/1.100
ESR(config-subif)#   ip firewall disable
ESR(config-subif)#   ip address 192.168.4.1/30
ESR(config-subif)# exit
ESR(config)#
ESR(config)# interface loopback 1
ESR(config-loopback)#   ip address 10.114.0.1/32
ESR(config-loopback)# exit
ESR(config)# do com
ESR(config)# do conf
```

Configure PE1 and PE2:

PE1

```
ESR(config)# hostname PE1
ESR(config)#
ESR(config)# ip vrf CE1
ESR(config-vrf)#   ip protocols bgp max-routes 100
ESR(config-vrf)#   rd 65500:1
ESR(config-vrf)#   route-target export 65500:1
ESR(config-vrf)#   route-target import 65500:1
ESR(config-vrf)# exit
ESR(config)# ip vrf CE2
ESR(config-vrf)#   ip protocols bgp max-routes 100
ESR(config-vrf)#   rd 65500:2
ESR(config-vrf)#   route-target export 65500:2
ESR(config-vrf)#   route-target import 65500:2
ESR(config-vrf)# exit
ESR(config)#
ESR(config)# system jumbo-frames
ESR(config)#
ESR(config)# route-map BGP
ESR(config-route-map)#   rule 1
ESR(config-route-map-rule)#   exit
ESR(config-route-map)# exit
ESR(config)# router bgp log-neighbor-changes
ESR(config)# router bgp 65500
ESR(config-bgp)#   neighbor 10.10.1.2
ESR(config-bgp-neighbor)#     remote-as 65500
ESR(config-bgp-neighbor)#     update-source 10.10.1.1
ESR(config-bgp-neighbor)#     address-family vpnv4 unicast
ESR(config-bgp-neighbor-af)#       send-community extended
ESR(config-bgp-neighbor-af)#       enable
ESR(config-bgp-neighbor-af)#     exit
ESR(config-bgp-neighbor)#     enable
ESR(config-bgp-neighbor)#   exit
ESR(config-bgp)#   enable
ESR(config-bgp)#   vrf CE1
ESR(config-bgp-vrf)#     neighbor 192.168.1.1
ESR(config-bgp-vrf-neighbor)#       remote-as 65501
ESR(config-bgp-vrf-neighbor)#       address-family ipv4 unicast
ESR(config-bgp-neighbor-af-vrf)#         route-map BGP out
ESR(config-bgp-neighbor-af-vrf)#         enable
ESR(config-bgp-neighbor-af-vrf)#       exit
ESR(config-bgp-vrf-neighbor)#       enable
ESR(config-bgp-vrf-neighbor)#     exit
ESR(config-bgp-vrf)#     address-family ipv4 unicast
ESR(config-bgp-vrf-af)#       redistribute bgp 65500 route-map BGP
ESR(config-bgp-vrf-af)#     exit
ESR(config-bgp-vrf)#     enable
ESR(config-bgp-vrf)#   exit
ESR(config-bgp)#   vrf CE2
ESR(config-bgp-vrf)#     neighbor 192.168.2.1
ESR(config-bgp-vrf-neighbor)#       remote-as 65501
ESR(config-bgp-vrf-neighbor)#       address-family ipv4 unicast
ESR(config-bgp-neighbor-af-vrf)#         route-map BGP out
ESR(config-bgp-neighbor-af-vrf)#         enable
ESR(config-bgp-neighbor-af-vrf)#       exit
ESR(config-bgp-vrf-neighbor)#       enable
ESR(config-bgp-vrf-neighbor)#     exit
```

```
ESR(config-bgp-vrf)#     address-family ipv4 unicast
ESR(config-bgp-vrf-af)#       redistribute bgp 65500 route-map BGP
ESR(config-bgp-vrf-af)#     exit
ESR(config-bgp-vrf)#     enable
ESR(config-bgp-vrf)#   exit
ESR(config-bgp)# exit
ESR(config)#
ESR(config)# router ospf 1
ESR(config-ospf)#   area 0.0.0.0
ESR(config-ospf-area)#     enable
ESR(config-ospf-area)#   exit
ESR(config-ospf)#   enable
ESR(config-ospf)# exit
ESR(config)#
ESR(config)# interface gigabitethernet 1/0/1.100
ESR(config-subif)#   ip vrf forwarding CE1
ESR(config-subif)#   description "to CE1"
ESR(config-subif)#   ip firewall disable
ESR(config-subif)#   ip address 192.168.1.2/30
ESR(config-subif)# exit
ESR(config)# interface gigabitethernet 1/0/1.200
ESR(config-subif)#   ip vrf forwarding CE2
ESR(config-subif)#   description "to CE2"
ESR(config-subif)#   ip firewall disable
ESR(config-subif)#   ip address 192.168.2.2/30
ESR(config-subif)# exit
ESR(config)# interface gigabitethernet 1/0/2
ESR(config-if-gi)#   mtu 1522
ESR(config-if-gi)#   ip firewall disable
ESR(config-if-gi)#   ip address 10.100.0.1/30
ESR(config-if-gi)#   ip ospf instance 1
ESR(config-if-gi)#   ip ospf
ESR(config-if-gi)# exit
ESR(config)# interface loopback 1
ESR(config-loopback)#   ip address 10.10.1.1/32
ESR(config-loopback)#   ip ospf instance 1
ESR(config-loopback)#   ip ospf
ESR(config-loopback)# exit
ESR(config)# mpls
ESR(config-mpls)#   ldp
ESR(config-ldp)#     router-id 10.10.1.1
ESR(config-ldp)#     address-family ipv4
ESR(config-ldp-af-ipv4)#       interface gigabitethernet 1/0/2
ESR(config-ldp-af-ipv4-if)#       exit
ESR(config-ldp-af-ipv4)#     exit
ESR(config-ldp)#     enable
ESR(config-ldp)#   exit
ESR(config-mpls)#   forwarding interface gigabitethernet 1/0/2
ESR(config-mpls)# exit
ESR(config)# do com
ESR(config)# do conf
```

PE2

```
ESR(config)# hostname PE2
ESR(config)#
ESR(config)# ip vrf CE1
ESR(config-vrf)#   ip protocols bgp max-routes 100
ESR(config-vrf)#   rd 65500:1
ESR(config-vrf)#   route-target export 65500:1
ESR(config-vrf)#   route-target import 65500:1
ESR(config-vrf)# exit
ESR(config)# ip vrf CE2
ESR(config-vrf)#   ip protocols bgp max-routes 100
ESR(config-vrf)#   rd 65500:2
ESR(config-vrf)#   route-target export 65500:2
ESR(config-vrf)#   route-target import 65500:2
ESR(config-vrf)# exit
ESR(config)#
ESR(config)# system jumbo-frames
ESR(config)#
ESR(config)# route-map BGP
ESR(config-route-map)#   rule 1
ESR(config-route-map-rule)#   exit
ESR(config-route-map)# exit
ESR(config)# router bgp log-neighbor-changes
ESR(config)# router bgp 65500
ESR(config-bgp)#   router-id 10.11.1.1
ESR(config-bgp)#   neighbor 10.11.1.2
ESR(config-bgp-neighbor)#     remote-as 65500
ESR(config-bgp-neighbor)#     update-source 10.11.1.1
ESR(config-bgp-neighbor)#     address-family vpnv4 unicast
ESR(config-bgp-neighbor-af)#       send-community extended
ESR(config-bgp-neighbor-af)#       enable
ESR(config-bgp-neighbor-af)#     exit
ESR(config-bgp-neighbor)#     enable
ESR(config-bgp-neighbor)#   exit
ESR(config-bgp)#   enable
ESR(config-bgp)#   vrf CE1
ESR(config-bgp-vrf)#     neighbor 192.168.3.1
ESR(config-bgp-vrf-neighbor)#       remote-as 65501
ESR(config-bgp-vrf-neighbor)#       address-family ipv4 unicast
ESR(config-bgp-neighbor-af-vrf)#         route-map BGP out
ESR(config-bgp-neighbor-af-vrf)#         enable
ESR(config-bgp-neighbor-af-vrf)#       exit
ESR(config-bgp-vrf-neighbor)#       enable
ESR(config-bgp-vrf-neighbor)#     exit
ESR(config-bgp-vrf)#     address-family ipv4 unicast
ESR(config-bgp-vrf-af)#       redistribute bgp 65500 route-map BGP
ESR(config-bgp-vrf-af)#     exit
ESR(config-bgp-vrf)#     enable
ESR(config-bgp-vrf)#   exit
ESR(config-bgp)#   vrf CE2
ESR(config-bgp-vrf)#     neighbor 192.168.4.1
ESR(config-bgp-vrf-neighbor)#       remote-as 65501
ESR(config-bgp-vrf-neighbor)#       address-family ipv4 unicast
ESR(config-bgp-neighbor-af-vrf)#         route-map BGP out
ESR(config-bgp-neighbor-af-vrf)#         enable
ESR(config-bgp-neighbor-af-vrf)#       exit
ESR(config-bgp-vrf-neighbor)#       enable
```

```
ESR(config-bgp-vrf-neighbor)#     exit
ESR(config-bgp-vrf)#     address-family ipv4 unicast
ESR(config-bgp-vrf-af)#       redistribute bgp 65500 route-map BGP
ESR(config-bgp-vrf-af)#     exit
ESR(config-bgp-vrf)#     enable
ESR(config-bgp-vrf)#   exit
ESR(config-bgp)# exit
ESR(config)#
ESR(config)# router ospf 1
ESR(config-ospf)#   area 0.0.0.0
ESR(config-ospf-area)#     enable
ESR(config-ospf-area)#   exit
ESR(config-ospf)#   enable
ESR(config-ospf)# exit
ESR(config)#
ESR(config)# interface gigabitethernet 1/0/1.100
ESR(config-subif)#   ip vrf forwarding CE1
ESR(config-subif)#   description "to CE3"
ESR(config-subif)#   ip firewall disable
ESR(config-subif)#   ip address 192.168.3.2/30
ESR(config-subif)# exit
ESR(config)# interface gigabitethernet 1/0/1.200
ESR(config-subif)#   ip vrf forwarding CE2
ESR(config-subif)#   description "to CE4"
ESR(config-subif)#   ip firewall disable
ESR(config-subif)#   ip address 192.168.4.2/30
ESR(config-subif)# exit
ESR(config)# interface gigabitethernet 1/0/2
ESR(config-if-gi)#   mtu 1522
ESR(config-if-gi)#   ip firewall disable
ESR(config-if-gi)#   ip address 10.101.0.1/30
ESR(config-if-gi)#   ip ospf instance 1
ESR(config-if-gi)#   ip ospf
ESR(config-if-gi)# exit
ESR(config)# interface loopback 1
ESR(config-loopback)#   ip address 10.11.1.1/32
ESR(config-loopback)#   ip ospf instance 1
ESR(config-loopback)#   ip ospf
ESR(config-loopback)# exit
ESR(config)# mpls
ESR(config-mpls)#   ldp
ESR(config-ldp)#     router-id 10.11.1.1
ESR(config-ldp)#     address-family ipv4
ESR(config-ldp-af-ipv4)#       interface gigabitethernet 1/0/2
ESR(config-ldp-af-ipv4-if)#       exit
ESR(config-ldp-af-ipv4)#     exit
ESR(config-ldp)#     enable
ESR(config-ldp)#   exit
ESR(config-mpls)#   forwarding interface gigabitethernet 1/0/2
ESR(config-mpls)# exit
ESR(config)# do com
ESR(config)# do conf
```

Configure ASBR1 and ASBR2. To transfer routing information between them, use the OSPF protocol in the corresponding VRFs:

ASBR1

```
ESR(config)# hostname ASBR1
ESR(config)#
ESR(config)# ip vrf CE1
ESR(config-vrf)#   ip protocols ospf max-routes 100
ESR(config-vrf)#   rd 65500:1
ESR(config-vrf)#   route-target export 65500:1
ESR(config-vrf)#   route-target import 65500:1
ESR(config-vrf)# exit
ESR(config)# ip vrf CE2
ESR(config-vrf)#   ip protocols ospf max-routes 100
ESR(config-vrf)#   rd 65500:2
ESR(config-vrf)#   route-target export 65500:2
ESR(config-vrf)#   route-target import 65500:2
ESR(config-vrf)# exit
ESR(config)#
ESR(config)# system jumbo-frames
ESR(config)#
ESR(config)# vlan 100,200
ESR(config-vlan)# exit
ESR(config)#
ESR(config)# router bgp 65500
ESR(config-bgp)#   router-id 10.10.1.2
ESR(config-bgp)#   neighbor 10.10.1.1
ESR(config-bgp-neighbor)#     remote-as 65500
ESR(config-bgp-neighbor)#     update-source 10.10.1.2
ESR(config-bgp-neighbor)#     address-family vpnv4 unicast
ESR(config-bgp-neighbor-af)#       send-community extended
ESR(config-bgp-neighbor-af)#       enable
ESR(config-bgp-neighbor-af)#     exit
ESR(config-bgp-neighbor)#     enable
ESR(config-bgp-neighbor)#   exit
ESR(config-bgp)#   enable
ESR(config-bgp)#   vrf CE1
ESR(config-bgp-vrf)#     address-family ipv4 unicast
ESR(config-bgp-vrf-af)#       redistribute ospf 1 intra-area inter-area external1 external2
ESR(config-bgp-vrf-af)#     exit
ESR(config-bgp-vrf)#   exit
ESR(config-bgp)#   vrf CE2
ESR(config-bgp-vrf)#     address-family ipv4 unicast
ESR(config-bgp-vrf-af)#       redistribute ospf 1 intra-area inter-area external1 external2
ESR(config-bgp-vrf-af)#     exit
ESR(config-bgp-vrf)#   exit
ESR(config-bgp)# exit
ESR(config)#
ESR(config)# router ospf log-adjacency-changes
ESR(config)# router ospf 1
ESR(config-ospf)#   area 0.0.0.0
ESR(config-ospf-area)#     enable
ESR(config-ospf-area)#   exit
ESR(config-ospf)#   enable
ESR(config-ospf)# exit
ESR(config)# router ospf 1 vrf CE1
ESR(config-ospf)#   redistribute bgp 65500
ESR(config-ospf)#   area 0.0.0.0
ESR(config-ospf-area)#     enable
ESR(config-ospf-area)#   exit
```

```
ESR(config-ospf)#   enable
ESR(config-ospf)# exit
ESR(config)# router ospf 1 vrf CE2
ESR(config-ospf)#   area 0.0.0.0
ESR(config-ospf-area)#     enable
ESR(config-ospf-area)#   exit
ESR(config-ospf)#   enable
ESR(config-ospf)# exit
ESR(config)#
ESR(config)# bridge 10
ESR(config-bridge)#   ip vrf forwarding CE1
ESR(config-bridge)#   vlan 100
ESR(config-bridge)#   ip firewall disable
ESR(config-bridge)#   ip address 172.16.32.1/30
ESR(config-bridge)#   ip ospf instance 1
ESR(config-bridge)#   ip ospf
ESR(config-bridge)#   enable
ESR(config-bridge)# exit
ESR(config)# bridge 20
ESR(config-bridge)#   ip vrf forwarding CE2
ESR(config-bridge)#   vlan 200
ESR(config-bridge)#   ip firewall disable
ESR(config-bridge)#   ip address 172.16.32.5/30
ESR(config-bridge)#   ip ospf instance 1
ESR(config-bridge)#   ip ospf
ESR(config-bridge)#   enable
ESR(config-bridge)# exit
ESR(config)#
ESR(config)# interface gigabitethernet 1/0/1
ESR(config-if-gi)#   description "to ASBR2"
ESR(config-if-gi)#   mode switchport
ESR(config-if-gi)#   mtu 1522
ESR(config-if-gi)#   spanning-tree disable
ESR(config-if-gi)#   switchport forbidden default-vlan
ESR(config-if-gi)#   switchport mode trunk
ESR(config-if-gi)#   switchport trunk allowed vlan add 100,200
ESR(config-if-gi)# exit
ESR(config)# interface gigabitethernet 1/0/2
ESR(config-if-gi)#   description "to PE1"
ESR(config-if-gi)#   mtu 1522
ESR(config-if-gi)#   ip firewall disable
ESR(config-if-gi)#   ip address 10.100.0.2/30
ESR(config-if-gi)#   ip ospf instance 1
ESR(config-if-gi)#   ip ospf
ESR(config-if-gi)# exit
ESR(config)# interface loopback 1
ESR(config-loopback)#   ip address 10.10.1.2/32
ESR(config-loopback)#   ip ospf instance 1
ESR(config-loopback)#   ip ospf
ESR(config-loopback)# exit
ESR(config)# mpls
ESR(config-mpls)#   ldp
ESR(config-ldp)#     router-id 10.10.1.2
ESR(config-ldp)#     address-family ipv4
ESR(config-ldp-af-ipv4)#       interface gigabitethernet 1/0/2
ESR(config-ldp-af-ipv4-if)#       exit
ESR(config-ldp-af-ipv4)#     exit
ESR(config-ldp)#     enable
ESR(config-ldp)#   exit
```

```
ESR(config-mpls)#   forwarding interface gigabitethernet 1/0/2
ESR(config-mpls)# exit
ESR(config)# do com
ESR(config)# do conf
```

ASBR2

```
ESR(config)# hostname ASBR2
ESR(config)#
ESR(config)# ip vrf CE1
ESR(config-vrf)#   ip protocols ospf max-routes 100
ESR(config-vrf)#   rd 65500:1
ESR(config-vrf)#   route-target export 65500:1
ESR(config-vrf)#   route-target import 65500:1
ESR(config-vrf)# exit
ESR(config)# ip vrf CE2
ESR(config-vrf)#   ip protocols ospf max-routes 100
ESR(config-vrf)#   rd 65500:2
ESR(config-vrf)#   route-target export 65500:2
ESR(config-vrf)#   route-target import 65500:2
ESR(config-vrf)# exit
ESR(config)#
ESR(config)# system jumbo-frames
ESR(config)#
ESR(config)# vlan 100,200
ESR(config-vlan)# exit
ESR(config)#
ESR(config)# router bgp 65500
ESR(config-bgp)#   router-id 10.11.1.2
ESR(config-bgp)#   neighbor 10.11.1.1
ESR(config-bgp-neighbor)#     remote-as 65500
ESR(config-bgp-neighbor)#     update-source 10.11.1.2
ESR(config-bgp-neighbor)#     address-family vpnv4 unicast
ESR(config-bgp-neighbor-af)#       send-community extended
ESR(config-bgp-neighbor-af)#       enable
ESR(config-bgp-neighbor-af)#     exit
ESR(config-bgp-neighbor)#     enable
ESR(config-bgp-neighbor)#   exit
ESR(config-bgp)#   enable
ESR(config-bgp)#   vrf CE1
ESR(config-bgp-vrf)#     address-family ipv4 unicast
ESR(config-bgp-vrf-af)#       redistribute ospf 1 intra-area inter-area external1 external2
ESR(config-bgp-vrf-af)#     exit
ESR(config-bgp-vrf)#   exit
ESR(config-bgp)#   vrf CE2
ESR(config-bgp-vrf)#     address-family ipv4 unicast
ESR(config-bgp-vrf-af)#       redistribute ospf 1 intra-area inter-area external1 external2
ESR(config-bgp-vrf-af)#     exit
ESR(config-bgp-vrf)#   exit
ESR(config-bgp)# exit
ESR(config)#
ESR(config)# router ospf log-adjacency-changes
ESR(config)# router ospf 1
ESR(config-ospf)#   area 0.0.0.0
ESR(config-ospf-area)#     enable
ESR(config-ospf-area)#   exit
ESR(config-ospf)#   enable
ESR(config-ospf)# exit
ESR(config)# router ospf 1 vrf CE1
ESR(config-ospf)#   redistribute bgp 65500
ESR(config-ospf)#   area 0.0.0.0
ESR(config-ospf-area)#     enable
ESR(config-ospf-area)#   exit
```

```
ESR(config-ospf)#   enable
ESR(config-ospf)# exit
ESR(config)# router ospf 1 vrf CE2
ESR(config-ospf)#   redistribute bgp 65500
ESR(config-ospf)#   area 0.0.0.0
ESR(config-ospf-area)#     enable
ESR(config-ospf-area)#   exit
ESR(config-ospf)#   enable
ESR(config-ospf)# exit
ESR(config)#
ESR(config)# bridge 10
ESR(config-bridge)#   ip vrf forwarding CE1
ESR(config-bridge)#   vlan 100
ESR(config-bridge)#   ip firewall disable
ESR(config-bridge)#   ip address 172.16.32.2/30
ESR(config-bridge)#   ip ospf instance 1
ESR(config-bridge)#   ip ospf
ESR(config-bridge)#   enable
ESR(config-bridge)# exit
ESR(config)# bridge 20
ESR(config-bridge)#   ip vrf forwarding CE2
ESR(config-bridge)#   vlan 200
ESR(config-bridge)#   ip firewall disable
ESR(config-bridge)#   ip address 172.16.32.6/30
ESR(config-bridge)#   ip ospf instance 1
ESR(config-bridge)#   ip ospf
ESR(config-bridge)#   enable
ESR(config-bridge)# exit
ESR(config)#
ESR(config)# interface gigabitethernet 1/0/1
ESR(config-if-gi)#   description "to ASBR1"
ESR(config-if-gi)#   mode switchport
ESR(config-if-gi)#   mtu 1522
ESR(config-if-gi)#   spanning-tree disable
ESR(config-if-gi)#   switchport forbidden default-vlan
ESR(config-if-gi)#   switchport mode trunk
ESR(config-if-gi)#   switchport trunk allowed vlan add 100,200
ESR(config-if-gi)# exit
ESR(config)# interface gigabitethernet 1/0/2
ESR(config-if-gi)#   description "to PE2"
ESR(config-if-gi)#   mtu 1522
ESR(config-if-gi)#   ip firewall disable
ESR(config-if-gi)#   ip address 10.101.0.2/30
ESR(config-if-gi)#   ip ospf instance 1
ESR(config-if-gi)#   ip ospf
ESR(config-if-gi)# exit
ESR(config)# interface loopback 1
ESR(config-loopback)#   ip address 10.11.1.2/32
ESR(config-loopback)#   ip ospf instance 1
ESR(config-loopback)#   ip ospf
ESR(config-loopback)# exit
ESR(config)# mpls
ESR(config-mpls)#   ldp
ESR(config-ldp)#     router-id 10.11.1.2
ESR(config-ldp)#     address-family ipv4
ESR(config-ldp-af-ipv4)#       interface gigabitethernet 1/0/2
ESR(config-ldp-af-ipv4-if)#       exit
ESR(config-ldp-af-ipv4)#     exit
ESR(config-ldp)#     enable
```

```
ESR(config-ldp)#   exit
ESR(config-mpls)#   forwarding interface gigabitethernet 1/0/2
ESR(config-mpls)# exit
ESR(config)# do com
ESR(config)# do conf
```

Configuration is completed. Check distribution of routing information and network availability of nodes:

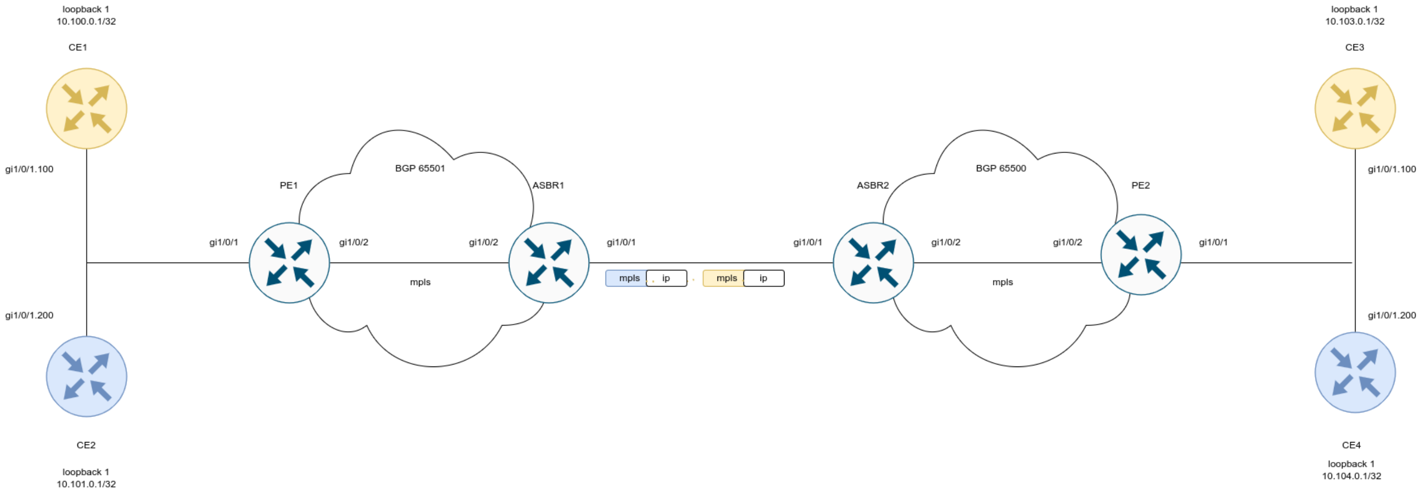
```
PE1# sh bgp vpnv4 unicast all
Status codes: * - valid, > - best, i - internal, S - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Codes Route Distinguisher   IP Prefix          Next hop        Metric     Label   LocPrf
Weight Path
----- --------------------- ------------------ --------------- ---------- ------- ----------
------ ----------------
*>    65500:1               10.110.0.1/32      --              --         37      100        --
65501 i
*>    65500:1               10.111.0.1/32      --              --         35      100        --
65501 i
*>i   65500:1               10.113.0.1/32      10.10.1.2       --         43      100        0
     ?
*>i   65500:1               10.114.0.1/32      10.10.1.2       --         48      100        0
     ?

CE1# ping 10.113.0.1 source ip 10.110.0.1 detailed
PING 10.113.0.1 (10.113.0.1) from 10.110.0.1 : 56 bytes of data.
64 bytes from 10.113.0.1: icmp_seq=1 ttl=0 time=1.31 ms
64 bytes from 10.113.0.1: icmp_seq=2 ttl=0 time=1.14 ms
64 bytes from 10.113.0.1: icmp_seq=3 ttl=0 time=1.08 ms
64 bytes from 10.113.0.1: icmp_seq=4 ttl=0 time=1.06 ms
64 bytes from 10.113.0.1: icmp_seq=5 ttl=0 time=1.16 ms
```

12.12 Inter-AS Option B

Unlike Option A, there is no need to use VRF between ASBRs: when passing traffic between ASBRs, an MPLS label will be assigned. This scheme has the best scalability.

In the current implementation, Option B is only supported for VPN-IPv4 routes (AFI = 1, SAFI = 128).

12.12.1 L3VPN



Configure CE:

CE1

```
ESR(config)# hostname CE1
ESR(config)#
ESR(config)# route-map BGP
ESR(config-route-map)#   rule 1
ESR(config-route-map-rule)#   exit
ESR(config-route-map)# exit
ESR(config)# router bgp 65501
ESR(config-bgp)#   neighbor 192.168.1.2
ESR(config-bgp-neighbor)#     remote-as 65500
ESR(config-bgp-neighbor)#     address-family ipv4 unicast
ESR(config-bgp-neighbor-af)#       route-map BGP out
ESR(config-bgp-neighbor-af)#       enable
ESR(config-bgp-neighbor-af)#     exit
ESR(config-bgp-neighbor)#     enable
ESR(config-bgp-neighbor)#   exit
ESR(config-bgp)#   address-family ipv4 unicast
ESR(config-bgp-af)#     network 10.110.0.1/32
ESR(config-bgp-af)#   exit
ESR(config-bgp)#   enable
ESR(config-bgp)# exit
ESR(config)# interface gigabitethernet 1/0/1.100
ESR(config-subif)#   ip firewall disable
ESR(config-subif)#   ip address 192.168.1.1/30
ESR(config-subif)# exit
ESR(config)# interface loopback 1
ESR(config-loopback)#   ip address 10.110.0.1/32
ESR(config-loopback)# exit
ESR(config)# do com
ESR(config)# do conf
```

CE2

```
ESR(config)# hostname CE2
ESR(config)#
ESR(config)# route-map BGP
ESR(config-route-map)#   rule 1
ESR(config-route-map-rule)#   exit
ESR(config-route-map)# exit
ESR(config)# router bgp 65501
ESR(config-bgp)#   neighbor 192.168.2.2
ESR(config-bgp-neighbor)#     remote-as 65500
ESR(config-bgp-neighbor)#     address-family ipv4 unicast
ESR(config-bgp-neighbor-af)#       route-map BGP out
ESR(config-bgp-neighbor-af)#       enable
ESR(config-bgp-neighbor-af)#     exit
ESR(config-bgp-neighbor)#     enable
ESR(config-bgp-neighbor)#   exit
ESR(config-bgp)#   address-family ipv4 unicast
ESR(config-bgp-af)#     network 10.112.0.1/32
ESR(config-bgp-af)#   exit
ESR(config-bgp)#   enable
ESR(config-bgp)# exit
ESR(config)# interface gigabitethernet 1/0/1.100
ESR(config-subif)#   ip firewall disable
ESR(config-subif)#   ip address 192.168.2.1/30
ESR(config-subif)# exit
ESR(config)#
ESR(config)# interface loopback 1
ESR(config-loopback)#   ip address 10.112.0.1/32
ESR(config-loopback)# exit
ESR(config)# do com
ESR(config)# do conf
```

CE3

```
ESR(config)# hostname CE3
ESR(config)#
ESR(config)# route-map BGP
ESR(config-route-map)#   rule 1
ESR(config-route-map-rule)#   exit
ESR(config-route-map)# exit
ESR(config)# router bgp 65501
ESR(config-bgp)#   neighbor 192.168.3.2
ESR(config-bgp-neighbor)#     remote-as 65500
ESR(config-bgp-neighbor)#     address-family ipv4 unicast
ESR(config-bgp-neighbor-af)#       route-map BGP out
ESR(config-bgp-neighbor-af)#       enable
ESR(config-bgp-neighbor-af)#     exit
ESR(config-bgp-neighbor)#     enable
ESR(config-bgp-neighbor)#   exit
ESR(config-bgp)#   address-family ipv4 unicast
ESR(config-bgp-af)#     network 10.113.0.1/32
ESR(config-bgp-af)#   exit
ESR(config-bgp)#   enable
ESR(config-bgp)# exit
ESR(config)# interface gigabitethernet 1/0/1.100
ESR(config-subif)#   ip firewall disable
ESR(config-subif)#   ip address 192.168.3.1/30
ESR(config-subif)# exit
ESR(config)#
ESR(config)# interface loopback 1
ESR(config-loopback)#   ip address 10.113.0.1/32
ESR(config-loopback)# exit
ESR(config)# do com
ESR(config)# do conf
```

CE4

```
ESR(config)# hostname CE4
ESR(config)#
ESR(config)# route-map BGP
ESR(config-route-map)#   rule 1
ESR(config-route-map-rule)#   exit
ESR(config-route-map)# exit
ESR(config)# router bgp 65501
ESR(config-bgp)#   neighbor 192.168.4.2
ESR(config-bgp-neighbor)#     remote-as 65500
ESR(config-bgp-neighbor)#     address-family ipv4 unicast
ESR(config-bgp-neighbor-af)#       route-map BGP out
ESR(config-bgp-neighbor-af)#       enable
ESR(config-bgp-neighbor-af)#     exit
ESR(config-bgp-neighbor)#     enable
ESR(config-bgp-neighbor)#   exit
ESR(config-bgp)#   address-family ipv4 unicast
ESR(config-bgp-af)#     network 10.114.0.1/32
ESR(config-bgp-af)#   exit
ESR(config-bgp)#   enable
ESR(config-bgp)# exit
ESR(config)# interface gigabitethernet 1/0/1.100
ESR(config-subif)#   ip firewall disable
ESR(config-subif)#   ip address 192.168.4.1/30
ESR(config-subif)# exit
ESR(config)#
ESR(config)# interface loopback 1
ESR(config-loopback)#   ip address 10.114.0.1/32
ESR(config-loopback)# exit
ESR(config)# do com
ESR(config)# do conf
```

Configure PE1 and PE2:

PE1

```
PE1(config)# hostname PE1
PE1(config)#
PE1(config)# ip vrf CE1
PE1(config-vrf)# ip protocols bgp max-routes 100
PE1(config-vrf)# rd 65501:1
PE1(config-vrf)# route-target export 65501:1
PE1(config-vrf)# route-target import 65501:1
PE1(config-vrf)# exit
PE1(config)# ip vrf CE2
PE1(config-vrf)# ip protocols bgp max-routes 100
PE1(config-vrf)# rd 65501:2
PE1(config-vrf)# route-target export 65501:2
PE1(config-vrf)# route-target import 65501:2
PE1(config-vrf)# exit
PE1(config)#
PE1(config)# system jumbo-frames
PE1(config)#
PE1(config)# route-map BGP_OUT
PE1(config-route-map)# rule 1
PE1(config-route-map-rule)# exit
PE1(config-route-map)# exit
PE1(config)# router bgp 65501
PE1(config-bgp)# neighbor 10.10.1.2
PE1(config-bgp-neighbor)# remote-as 65501
PE1(config-bgp-neighbor)# update-source 10.10.1.1
PE1(config-bgp-neighbor)# address-family vpnv4 unicast
PE1(config-bgp-neighbor-af)# send-community extended
PE1(config-bgp-neighbor-af)# enable
PE1(config-bgp-neighbor-af)# exit
PE1(config-bgp-neighbor)# enable
PE1(config-bgp-neighbor)# exit
PE1(config-bgp)# enable
PE1(config-bgp)# vrf CE1
PE1(config-bgp-vrf)# neighbor 192.168.1.1
PE1(config-bgp-vrf-neighbor)# remote-as 65510
PE1(config-bgp-vrf-neighbor)# address-family ipv4 unicast
PE1(config-bgp-neighbor-af-vrf)# route-map BGP_OUT out
PE1(config-bgp-neighbor-af-vrf)# enable
PE1(config-bgp-neighbor-af-vrf)# exit
PE1(config-bgp-vrf-neighbor)# enable
PE1(config-bgp-vrf-neighbor)# exit
PE1(config-bgp-vrf)# address-family ipv4 unicast
PE1(config-bgp-vrf-af)# redistribute bgp 65501 route-map BGP_OUT
PE1(config-bgp-vrf-af)# exit
PE1(config-bgp-vrf)# enable
PE1(config-bgp-vrf)# exit
PE1(config-bgp)# vrf CE2
PE1(config-bgp-vrf)# neighbor 192.168.2.1
PE1(config-bgp-vrf-neighbor)# remote-as 65511
PE1(config-bgp-vrf-neighbor)# address-family ipv4 unicast
PE1(config-bgp-neighbor-af-vrf)# route-map BGP_OUT out
PE1(config-bgp-neighbor-af-vrf)# enable
PE1(config-bgp-neighbor-af-vrf)# exit
PE1(config-bgp-vrf-neighbor)# enable
PE1(config-bgp-vrf-neighbor)# exit
PE1(config-bgp-vrf)# address-family ipv4 unicast
```

```
PE1(config-bgp-vrf-af)# redistribute bgp 65501 route-map BGP_OUT
PE1(config-bgp-vrf-af)# exit
PE1(config-bgp-vrf)# enable
PE1(config-bgp-vrf)# exit
PE1(config-bgp)# exit
PE1(config)#
PE1(config)# router ospf 1
PE1(config-ospf)# area 0.0.0.0
PE1(config-ospf-area)# enable
PE1(config-ospf-area)# exit
PE1(config-ospf)# enable
PE1(config-ospf)# exit
PE1(config)#
PE1(config)# interface gigabitethernet 1/0/1.100
PE1(config-subif)# ip vrf forwarding CE1
PE1(config-subif)# description "to CE1"
PE1(config-subif)# ip firewall disable
PE1(config-subif)# ip address 192.168.1.2/30
PE1(config-subif)# exit
PE1(config)# interface gigabitethernet 1/0/1.200
PE1(config-subif)# ip vrf forwarding CE2
PE1(config-subif)# description "to CE2"
PE1(config-subif)# ip firewall disable
PE1(config-subif)# ip address 192.168.2.2/30
PE1(config-subif)# exit
PE1(config)# interface gigabitethernet 1/0/2
PE1(config-if-gi)# description "to ASBR1"
PE1(config-if-gi)# mtu 1522
PE1(config-if-gi)# ip firewall disable
PE1(config-if-gi)# ip address 10.100.0.1/30
PE1(config-if-gi)# ip ospf instance 1
PE1(config-if-gi)# ip ospf
PE1(config-if-gi)# exit
PE1(config)# interface loopback 1
PE1(config-loopback)# ip address 10.10.1.1/32
PE1(config-loopback)# ip ospf instance 1
PE1(config-loopback)# ip ospf
PE1(config-loopback)# exit
PE1(config)# mpls
PE1(config-mpls)# ldp
PE1(config-ldp)# router-id 10.10.1.1
PE1(config-ldp)# address-family ipv4
PE1(config-ldp-af-ipv4)# interface gigabitethernet 1/0/2
PE1(config-ldp-af-ipv4-if)# exit
PE1(config-ldp-af-ipv4)# exit
PE1(config-ldp)# enable
PE1(config-ldp)# exit
PE1(config-mpls)# forwarding interface gigabitethernet 1/0/2
PE1(config-mpls)# exit
PE1(config)# do com
PE1(config)# do conf
```

PE2

```
PE2(config)# hostname PE2
PE2(config)#
PE2(config)# ip vrf CE1
PE2(config-vrf)#   ip protocols bgp max-routes 100
PE2(config-vrf)#   rd 65501:1
PE2(config-vrf)#   route-target export 65501:1
PE2(config-vrf)#   route-target import 65501:1
PE2(config-vrf)# exit
PE2(config)# ip vrf CE2
PE2(config-vrf)#   ip protocols bgp max-routes 100
PE2(config-vrf)#   rd 65501:2
PE2(config-vrf)#   route-target export 65501:2
PE2(config-vrf)#   route-target import 65501:2
PE2(config-vrf)# exit
PE2(config)#
PE2(config)# system jumbo-frames
PE2(config)#
PE2(config)# route-map BGP_OUT
PE2(config-route-map)#   rule 1
PE2(config-route-map-rule)#   exit
PE2(config-route-map)# exit
PE2(config)# router bgp 65500
PE2(config-bgp)#   neighbor 10.11.1.2
PE2(config-bgp-neighbor)#     remote-as 65500
PE2(config-bgp-neighbor)#     update-source 10.11.1.1
PE2(config-bgp-neighbor)#     address-family vpnv4 unicast
PE2(config-bgp-neighbor-af)#       send-community extended
PE2(config-bgp-neighbor-af)#       enable
PE2(config-bgp-neighbor-af)#     exit
PE2(config-bgp-neighbor)#     enable
PE2(config-bgp-neighbor)#   exit
PE2(config-bgp)#   enable
PE2(config-bgp)#   vrf CE1
PE2(config-bgp-vrf)#     neighbor 192.168.3.1
PE2(config-bgp-vrf-neighbor)#       remote-as 65512
PE2(config-bgp-vrf-neighbor)#       address-family ipv4 unicast
PE2(config-bgp-neighbor-af-vrf)#         route-map BGP_OUT out
PE2(config-bgp-neighbor-af-vrf)#         enable
PE2(config-bgp-neighbor-af-vrf)#       exit
PE2(config-bgp-vrf-neighbor)#       enable
PE2(config-bgp-vrf-neighbor)#     exit
PE2(config-bgp-vrf)#     address-family ipv4 unicast
PE2(config-bgp-vrf-af)#       redistribute bgp 65500 route-map BGP_OUT
PE2(config-bgp-vrf-af)#     exit
PE2(config-bgp-vrf)#     enable
PE2(config-bgp-vrf)#   exit
PE2(config-bgp)#   vrf CE2
PE2(config-bgp-vrf)#     neighbor 192.168.4.1
PE2(config-bgp-vrf-neighbor)#       remote-as 65513
PE2(config-bgp-vrf-neighbor)#       address-family ipv4 unicast
PE2(config-bgp-neighbor-af-vrf)#         route-map BGP_OUT out
PE2(config-bgp-neighbor-af-vrf)#         enable
PE2(config-bgp-neighbor-af-vrf)#       exit
PE2(config-bgp-vrf-neighbor)#       enable
PE2(config-bgp-vrf-neighbor)#     exit
PE2(config-bgp-vrf)#     address-family ipv4 unicast
```

```
PE2(config-bgp-vrf-af)#       redistribute bgp 65500 route-map BGP_OUT
PE2(config-bgp-vrf-af)#     exit
PE2(config-bgp-vrf)#     enable
PE2(config-bgp-vrf)#   exit
PE2(config-bgp)# exit
PE2(config)#
PE2(config)# router ospf 1
PE2(config-ospf)#   router-id 10.11.1.1
PE2(config-ospf)#   area 0.0.0.0
PE2(config-ospf-area)#     enable
PE2(config-ospf-area)#   exit
PE2(config-ospf)#   enable
PE2(config-ospf)# exit
PE2(config)#
PE2(config)# interface gigabitethernet 1/0/1.100
PE2(config-subif)#   ip vrf forwarding CE1
PE2(config-subif)#   description "to CE3"
PE2(config-subif)#   ip firewall disable
PE2(config-subif)#   ip address 192.168.3.2/30
PE2(config-subif)# exit
PE2(config)# interface gigabitethernet 1/0/1.200
PE2(config-subif)#   ip vrf forwarding CE2
PE2(config-subif)#   description "CE4"
PE2(config-subif)#   ip firewall disable
PE2(config-subif)#   ip address 192.168.4.2/30
PE2(config-subif)# exit
PE2(config)# interface gigabitethernet 1/0/2
PE2(config-if-gi)#   description "to ASBR2"
PE2(config-if-gi)#   mtu 1522
PE2(config-if-gi)#   ip firewall disable
PE2(config-if-gi)#   ip address 10.102.0.1/30
PE2(config-if-gi)#   ip ospf instance 1
PE2(config-if-gi)#   ip ospf
PE2(config-if-gi)# exit
PE2(config)# interface loopback 1
PE2(config-loopback)#   ip address 10.11.1.1/32
PE2(config-loopback)#   ip ospf instance 1
PE2(config-loopback)#   ip ospf
PE2(config-loopback)# exit
PE2(config)# mpls
PE2(config-mpls)#   ldp
PE2(config-ldp)#     router-id 10.11.1.1
PE2(config-ldp)#     address-family ipv4
PE2(config-ldp-af-ipv4)#       interface gigabitethernet 1/0/2
PE2(config-ldp-af-ipv4-if)#       exit
PE2(config-ldp-af-ipv4)#     exit
PE2(config-ldp)#     enable
PE2(config-ldp)#   exit
PE2(config-mpls)#   forwarding interface gigabitethernet 1/0/2
PE2(config-mpls)# exit
PE2(config)# do com
PE2(config)# do conf
```

Configure ASBR1 and ASBR2:

ASBR1

```
ASBR1(config)# hostname ASBR1
ASBR1(config)#
ASBR1(config)# system jumbo-frames
ASBR1(config)#
ASBR1(config)# route-map VPNv4
ASBR1(config-route-map)#   rule 1
ASBR1(config-route-map-rule)#   exit
ASBR1(config-route-map)# exit
ASBR1(config)# router bgp 65501
ASBR1(config-bgp)#   router-id 10.10.1.2
ASBR1(config-bgp)#   neighbor 10.10.1.1
ASBR1(config-bgp-neighbor)#     remote-as 65501
ASBR1(config-bgp-neighbor)#     update-source 10.10.1.2
ASBR1(config-bgp-neighbor)#     address-family vpnv4 unicast
ASBR1(config-bgp-neighbor-af)#       next-hop-self
ASBR1(config-bgp-neighbor-af)#       send-community extended
ASBR1(config-bgp-neighbor-af)#       enable
ASBR1(config-bgp-neighbor-af)#     exit
ASBR1(config-bgp-neighbor)#     enable
ASBR1(config-bgp-neighbor)#   exit
ASBR1(config-bgp)#   neighbor 10.101.0.1
ASBR1(config-bgp-neighbor)#     remote-as 65500
ASBR1(config-bgp-neighbor)#     address-family vpnv4 unicast
ASBR1(config-bgp-neighbor-af)#       route-map VPNv4 out
ASBR1(config-bgp-neighbor-af)#       send-community extended
ASBR1(config-bgp-neighbor-af)#       enable
ASBR1(config-bgp-neighbor-af)#     exit
ASBR1(config-bgp-neighbor)#     enable
ASBR1(config-bgp-neighbor)#   exit
ASBR1(config-bgp)#   enable
ASBR1(config-bgp)# exit
ASBR1(config)#
ASBR1(config)# router ospf 1
ASBR1(config-ospf)#   area 0.0.0.0
ASBR1(config-ospf-area)#     enable
ASBR1(config-ospf-area)#   exit
ASBR1(config-ospf)#   enable
ASBR1(config-ospf)# exit
ASBR1(config)#
ASBR1(config)# interface gigabitethernet 1/0/1
ASBR1(config-if-gi)#   description "to ASBR2"
ASBR1(config-if-gi)#   ip firewall disable
ASBR1(config-if-gi)#   ip address 10.101.0.2/30
ASBR1(config-if-gi)# exit
ASBR1(config)# interface gigabitethernet 1/0/2
ASBR1(config-if-gi)#   description "to PE1"
ASBR1(config-if-gi)#   mtu 1522
ASBR1(config-if-gi)#   ip firewall disable
ASBR1(config-if-gi)#   ip address 10.100.0.2/30
ASBR1(config-if-gi)#   ip ospf instance 1
ASBR1(config-if-gi)#   ip ospf
ASBR1(config-if-gi)# exit
ASBR1(config)# interface loopback 1
ASBR1(config-loopback)#   ip address 10.10.1.2/32
ASBR1(config-loopback)#   ip ospf instance 1
ASBR1(config-loopback)#   ip ospf
```

```
ASBR1(config-loopback)# exit
ASBR1(config)# mpls
ASBR1(config-mpls)#   ldp
ASBR1(config-ldp)#     router-id 10.10.1.2
ASBR1(config-ldp)#     address-family ipv4
ASBR1(config-ldp-af-ipv4)#       interface gigabitethernet 1/0/2
ASBR1(config-ldp-af-ipv4-if)#       exit
ASBR1(config-ldp-af-ipv4)#     exit
ASBR1(config-ldp)#     enable
ASBR1(config-ldp)#   exit
ASBR1(config-mpls)#   forwarding interface gigabitethernet 1/0/1
ASBR1(config-mpls)#   forwarding interface gigabitethernet 1/0/2
ASBR1(config-mpls)# exit
ASBR1(config)# do com
ASBR1(config)# do conf
```

ASBR2

```
ASBR2(config)# hostname ASBR2
ASBR2(config)#
ASBR2(config)# system jumbo-frames
ASBR2(config)#
ASBR2(config)# route-map VPNv4
ASBR2(config-route-map)#   rule 1
ASBR2(config-route-map-rule)#   exit
ASBR2(config-route-map)# exit
ASBR2(config)# router bgp 65500
ASBR2(config-bgp)#   router-id 10.11.1.2
ASBR2(config-bgp)#   neighbor 10.101.0.2
ASBR2(config-bgp-neighbor)#     remote-as 65501
ASBR2(config-bgp-neighbor)#     address-family vpnv4 unicast
ASBR2(config-bgp-neighbor-af)#       route-map VPNv4 out
ASBR2(config-bgp-neighbor-af)#       send-community extended
ASBR2(config-bgp-neighbor-af)#       enable
ASBR2(config-bgp-neighbor-af)#     exit
ASBR2(config-bgp-neighbor)#     enable
ASBR2(config-bgp-neighbor)#   exit
ASBR2(config-bgp)#   neighbor 10.11.1.1
ASBR2(config-bgp-neighbor)#     remote-as 65500
ASBR2(config-bgp-neighbor)#     update-source 10.11.1.2
ASBR2(config-bgp-neighbor)#     address-family vpnv4 unicast
ASBR2(config-bgp-neighbor-af)#       next-hop-self
ASBR2(config-bgp-neighbor-af)#       send-community extended
ASBR2(config-bgp-neighbor-af)#       enable
ASBR2(config-bgp-neighbor-af)#     exit
ASBR2(config-bgp-neighbor)#     enable
ASBR2(config-bgp-neighbor)#   exit
ASBR2(config-bgp)#   enable
ASBR2(config-bgp)# exit
ASBR2(config)#
ASBR2(config)# router ospf 1
ASBR2(config-ospf)#   router-id 10.11.1.2
ASBR2(config-ospf)#   area 0.0.0.0
ASBR2(config-ospf-area)#     enable
ASBR2(config-ospf-area)#   exit
ASBR2(config-ospf)#   enable
ASBR2(config-ospf)# exit
ASBR2(config)#
ASBR2(config)# interface gigabitethernet 1/0/1
ASBR2(config-if-gi)#   description "to ASBR1"
ASBR2(config-if-gi)#   ip firewall disable
ASBR2(config-if-gi)#   ip address 10.101.0.1/30
ASBR2(config-if-gi)# exit
ASBR2(config)# interface gigabitethernet 1/0/2
ASBR2(config-if-gi)#   description "to PE2"
ASBR2(config-if-gi)#   mtu 1522
ASBR2(config-if-gi)#   ip firewall disable
ASBR2(config-if-gi)#   ip address 10.102.0.2/30
ASBR2(config-if-gi)#   ip ospf instance 1
ASBR2(config-if-gi)#   ip ospf
ASBR2(config-if-gi)# exit
ASBR2(config)# interface loopback 1
ASBR2(config-loopback)#   ip address 10.11.1.2/32
ASBR2(config-loopback)#   ip ospf instance 1
```

```
ASBR2(config-loopback)#   ip ospf
ASBR2(config-loopback)# exit
ASBR2(config)# mpls
ASBR2(config-mpls)#   ldp
ASBR2(config-ldp)#     router-id 10.11.1.2
ASBR2(config-ldp)#     address-family ipv4
ASBR2(config-ldp-af-ipv4)#       interface gigabitethernet 1/0/2
ASBR2(config-ldp-af-ipv4-if)#       exit
ASBR2(config-ldp-af-ipv4)#     exit
ASBR2(config-ldp)#     enable
ASBR2(config-ldp)#   exit
ASBR2(config-mpls)#   forwarding interface gigabitethernet 1/0/1
ASBR2(config-mpls)#   forwarding interface gigabitethernet 1/0/2
ASBR2(config-mpls)# exit
ASBR2(config)# do com
ASBR2(config)# do conf
```

After completing the configuration, check the distribution of routing information and the network availability of the nodes:

```
PE1# sh 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   65501:2               10.104.0.1/32      10.10.1.2       --         23      100        0
65500 65513 i
*>i   65501:1               10.103.0.1/32      10.10.1.2       --         19      100        0
65500 65512 i
*>    65501:2               10.101.0.1/32      --              --         29      100        --
65511 i
*>    65501:1               10.100.0.1/32      --              --         28      100        --
65510 i
ASBR1#    sh 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
----- --------------------- ------------------ --------------- ---------- ------- ----------
------ ----------------
*>    65501:2               10.104.0.1/32      10.101.0.1      --         24      100        0
65500 65513 i
*>    65501:1               10.103.0.1/32      10.101.0.1      --         20      100        0
65500 65512 i
*>i   65501:2               10.101.0.1/32      10.10.1.1       --         29      100        0
65511 i
*>i   65501:1               10.100.0.1/32      10.10.1.1       --         28      100        0
65510 i

ASBR2# sh 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   65501:2               10.104.0.1/32      10.11.1.1       --         19      100        0
65513 i
*>i   65501:1               10.103.0.1/32      10.11.1.1       --         18      100        0
65512 i
*>    65501:2               10.101.0.1/32      10.101.0.2      --         30      100        0
65501 65511 i
*>    65501:1               10.100.0.1/32      10.101.0.2      --         31      100        0
65501 65510 i

PE2# sh 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
----- --------------------- ------------------ --------------- ---------- ------- ----------
------ ----------------
```

```
*>     65501:2            10.104.0.1/32      --              --          19      100         --
65513 i
*>     65501:1            10.103.0.1/32      --              --          18      100         --
65512 i
*>i    65501:2            10.101.0.1/32      10.11.1.2       --          29      100         0
65501 65511 i
*>i    65501:1            10.100.0.1/32      10.11.1.2       --          30      100         0
65501 65510 i

CE4# ping 10.104.0.1 source ip 10.101.0.1  detailed
PING 10.104.0.1 (10.104.0.1) from 10.101.0.1 : 56 bytes of data.
64 bytes from 10.104.0.1: icmp_seq=1 ttl=0 time=2.25 ms
64 bytes from 10.104.0.1: icmp_seq=2 ttl=0 time=2.08 ms
64 bytes from 10.104.0.1: icmp_seq=3 ttl=0 time=2.15 ms
64 bytes from 10.104.0.1: icmp_seq=4 ttl=0 time=2.12 ms
64 bytes from 10.104.0.1: icmp_seq=5 ttl=0 time=2.09 ms

CE1# ping 10.103.0.1 source ip 10.100.0.1 detailed
PING 10.103.0.1 (10.103.0.1) from 10.100.0.1 : 56 bytes of data.
64 bytes from 10.103.0.1: icmp_seq=1 ttl=0 time=2.22 ms
64 bytes from 10.103.0.1: icmp_seq=2 ttl=0 time=2.11 ms
64 bytes from 10.103.0.1: icmp_seq=3 ttl=0 time=2.09 ms
64 bytes from 10.103.0.1: icmp_seq=4 ttl=0 time=2.09 ms
64 bytes from 10.103.0.1: icmp_seq=5 ttl=0 time=2.11 ms
```

12.13 MPLS over GRE

This section provides example of configuration of VPN services built through a GRE tunnel.

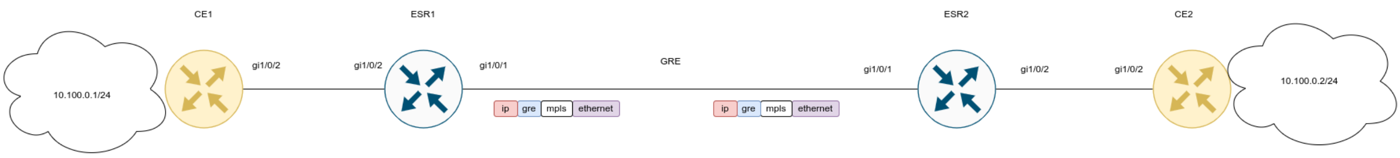
12.13.1 L2VPN

Example of configuration of EoMPLS over GRE as a l2vpn service is given below. It is also possible to build VPLS over GRE (BGP or LDP signaling).

When configuring MTU on a tunnel, consider the following:

- **At least one mpls label will be present when passing through the tunnel. Accounting should include all labels in the stack, for example,** explicit null **or entropy label;**
- **vlan and q-in-q headers must be taken into account (if they present);**
- **If MTU of the outgoing interface is exceeded, the packet will be dropped (if GRE traffic** match fragment **is disabled);**
- **Control world is not supported;**
- **DF bit will be set to 1.**

Approximate packet structure is given in figure below:



Configure CE1 and CE2:

CE1

```
hostname CE1

interface gigabitethernet 1/0/2
  ip firewall disable
  ip address 10.100.0.1/24
exit
```

CE2

```
hostname CE2

interface gigabitethernet 1/0/2
  ip firewall disable
  ip address 10.100.0.2/24
exit
```

Configure ESR1 and ESR2:

ESR1

```
ESR1(config)# hostname ESR1
ESR1(config)#
ESR1(config)# system cpu load-balance mpls passenger ip
ESR1(config)# system cpu load-balance mpls passenger ipoe-pw-without-cw
ESR1(config)# security zone trusted
ESR1(config-zone)# exit
ESR1(config)# security zone untrusted
ESR1(config-zone)# exit
ESR1(config)#
ESR1(config)# router ospf 1
ESR1(config-ospf)#   area 0.0.0.0
ESR1(config-ospf-area)#     enable
ESR1(config-ospf-area)#   exit
ESR1(config-ospf)#   enable
ESR1(config-ospf)# exit
ESR1(config)#
ESR1(config)# interface gigabitethernet 1/0/1
ESR1(config-if-gi)#   security-zone untrusted
ESR1(config-if-gi)#   ip address 192.0.2.1/30
ESR1(config-if-gi)# exit
ESR1(config)# interface gigabitethernet 1/0/2
ESR1(config-if-gi)#   description "From CE1"
ESR1(config-if-gi)#   mode switchport
ESR1(config-if-gi)# exit
ESR1(config)# interface loopback 1
ESR1(config-loopback)#   ip address 10.100.0.1/32
ESR1(config-loopback)#   ip ospf instance 1
ESR1(config-loopback)#   ip ospf
ESR1(config-loopback)# exit
ESR1(config)# tunnel gre 1
ESR1(config-gre)#   key 60
ESR1(config-gre)#   ttl 64
ESR1(config-gre)#   mtu 1458
ESR1(config-gre)#   ip firewall disable
ESR1(config-gre)#   local address 192.0.2.1
ESR1(config-gre)#   remote address 192.0.2.2
ESR1(config-gre)#   ip address 10.0.0.1/30
ESR1(config-gre)#   ip ospf instance 1
ESR1(config-gre)#   ip ospf network point-to-point
ESR1(config-gre)#   ip ospf
ESR1(config-gre)#   enable
ESR1(config-gre)# exit
ESR1(config)#
ESR1(config)# mpls
ESR1(config-mpls)#   ldp
ESR1(config-ldp)#     router-id 10.100.0.1
ESR1(config-ldp)#     address-family ipv4
ESR1(config-ldp-af-ipv4)#       interface gre 1
ESR1(config-ldp-af-ipv4-if)#       exit
ESR1(config-ldp-af-ipv4)#     exit
ESR1(config-ldp)#     enable
ESR1(config-ldp)#   exit
ESR1(config-mpls)#   l2vpn
ESR1(config-l2vpn)#     pw-class VPWS
ESR1(config-l2vpn-pw-class)#     exit
ESR1(config-l2vpn)#     p2p EoMPLS
```

```
ESR1(config-l2vpn-p2p)#       interface gigabitethernet 1/0/2
ESR1(config-l2vpn-p2p)#       pw 100 10.100.0.2
ESR1(config-l2vpn-pw)#         pw-class VPWS
ESR1(config-l2vpn-pw)#         enable
ESR1(config-l2vpn-pw)#       exit
ESR1(config-l2vpn-p2p)#       enable
ESR1(config-l2vpn-p2p)#     exit
ESR1(config-l2vpn)#   exit
ESR1(config-mpls)#   forwarding interface gre 1
ESR1(config-mpls)# exit
ESR1(config)# security zone-pair untrusted self
ESR1(config-zone-pair)#   rule 1
ESR1(config-zone-pair-rule)#     action permit
ESR1(config-zone-pair-rule)#     match protocol gre
ESR1(config-zone-pair-rule)#     enable
ESR1(config-zone-pair-rule)#   exit
ESR1(config-zone-pair)# exit
ESR1(config)# do com
ESR1(config)# do conf
```

ESR2

```
ESR2(config)# hostname ESR2
ESR2(config)#
ESR2(config)# system cpu load-balance mpls passenger ip
ESR2(config)# system cpu load-balance mpls passenger ipoe-pw-without-cw
ESR2(config)# security zone trusted
ESR2(config-zone)# exit
ESR2(config)# security zone untrusted
ESR2(config-zone)# exit
ESR2(config)#
ESR2(config)# router ospf 1
ESR2(config-ospf)#   area 0.0.0.0
ESR2(config-ospf-area)#     enable
ESR2(config-ospf-area)#   exit
ESR2(config-ospf)#   enable
ESR2(config-ospf)# exit
ESR2(config)#
ESR2(config)# interface gigabitethernet 1/0/1
ESR2(config-if-gi)#   security-zone untrusted
ESR2(config-if-gi)#   ip address 192.0.2.2/30
ESR2(config-if-gi)# exit
ESR2(config)# interface gigabitethernet 1/0/2
ESR2(config-if-gi)#   description "From CE2"
ESR2(config-if-gi)#   mode switchport
ESR2(config-if-gi)# exit
ESR2(config)# interface loopback 1
ESR2(config-loopback)#   ip address 10.100.0.2/32
ESR2(config-loopback)#   ip ospf instance 1
ESR2(config-loopback)#   ip ospf
ESR2(config-loopback)# exit
ESR2(config)# tunnel gre 1
ESR2(config-gre)#   key 60
ESR2(config-gre)#   ttl 64
ESR2(config-gre)#   mtu 1458
ESR2(config-gre)#   ip firewall disable
ESR2(config-gre)#   local address 192.0.2.2
ESR2(config-gre)#   remote address 192.0.2.1
ESR2(config-gre)#   ip address 10.0.0.2/30
ESR2(config-gre)#   ip ospf instance 1
ESR2(config-gre)#   ip ospf network point-to-point
ESR2(config-gre)#   ip ospf
ESR2(config-gre)#   enable
ESR2(config-gre)# exit
ESR2(config)#
ESR2(config)# mpls
ESR2(config-mpls)#   ldp
ESR2(config-ldp)#     router-id 10.100.0.2
ESR2(config-ldp)#     address-family ipv4
ESR2(config-ldp-af-ipv4)#       interface gre 1
ESR2(config-ldp-af-ipv4-if)#       exit
ESR2(config-ldp-af-ipv4)#     exit
ESR2(config-ldp)#     enable
ESR2(config-ldp)#   exit
ESR2(config-mpls)#   l2vpn
ESR2(config-l2vpn)#     pw-class VPWS
ESR2(config-l2vpn-pw-class)#     exit
ESR2(config-l2vpn)#     p2p EoMPLS
```

```
ESR2(config-l2vpn-p2p)#       interface gigabitethernet 1/0/2
ESR2(config-l2vpn-p2p)#       pw 100 10.100.0.1
ESR2(config-l2vpn-pw)#         pw-class VPWS
ESR2(config-l2vpn-pw)#         enable
ESR2(config-l2vpn-pw)#       exit
ESR2(config-l2vpn-p2p)#       enable
ESR2(config-l2vpn-p2p)#     exit
ESR2(config-l2vpn)#   exit
ESR2(config-mpls)#   forwarding interface gre 1
ESR2(config-mpls)# exit
ESR2(config)# security zone-pair untrusted self
ESR2(config-zone-pair)#   rule 1
ESR2(config-zone-pair-rule)#     action deny
ESR2(config-zone-pair-rule)#     match protocol gre
ESR2(config-zone-pair-rule)#     enable
ESR2(config-zone-pair-rule)#   exit
ESR2(config-zone-pair)# exit
ESR2(config)# do com
ESR2(config)# do conf
```

Configuration is complete. Check the service state and nodes availability:

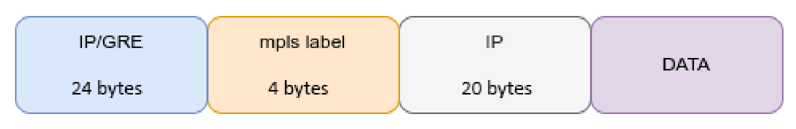
```
*Tunnel configuration*
ESR2# sh tunnels configuration gre 1
State:                                          Enabled
Description:                                    --
Mode:                                           ip
Bridge group:                                   --
VRF:                                            --
Local address:                                  192.0.2.2
Remote address:                                 192.0.2.1
Calculates checksums for outgoing GRE packets:  No
Requires that all input GRE packets were checksum: No
key:                                            60
TTL:                                            64
DSCP:                                           Inherit
MTU:                                            1458
Path MTU discovery:                             Enabled
Don't fragment bit suppression:                 Disabled
Security zone:                                  --
Multipoint mode:                                Disabled
Keepalive:
    State:                                          Disabled
    Timeout:                                        10
    Retries:                                        6
    Destination address:                            --

*Service status and selected tags*
sh mpls l2vpn p2p
P2P: EoMPLS
    gigabitethernet 1/0/2:
        MTU:    1500
        Status: Up
    PW ID 100, Neighbor 10.100.0.1:
        MTU:         1500
        Status TLV:  Enable
        Last change: 00:14:27
        Status:      Up

ESR2# sh mpls forwarding-table
Local    Outgoing Prefix                                     Outgoing    Next Hop
label    label    or tunnel ID                               Interface
-------- -------- ------------------------------------------- ----------- -----------
17       imp-null 10.100.0.1/32                              gre 1       10.0.0.1
16       16       PW ID 100                                  --          10.100.0.1

*Availability*CE1# ping 10.100.0.2 detailed
PING 10.100.0.2 (10.100.0.2) 56 bytes of data.
64 bytes from 10.100.0.2: icmp_seq=1 ttl=0 time=1.38 ms
64 bytes from 10.100.0.2: icmp_seq=2 ttl=0 time=1.22 ms
64 bytes from 10.100.0.2: icmp_seq=3 ttl=0 time=1.33 ms
64 bytes from 10.100.0.2: icmp_seq=4 ttl=0 time=1.26 ms
64 bytes from 10.100.0.2: icmp_seq=5 ttl=0 time=1.17 ms
```

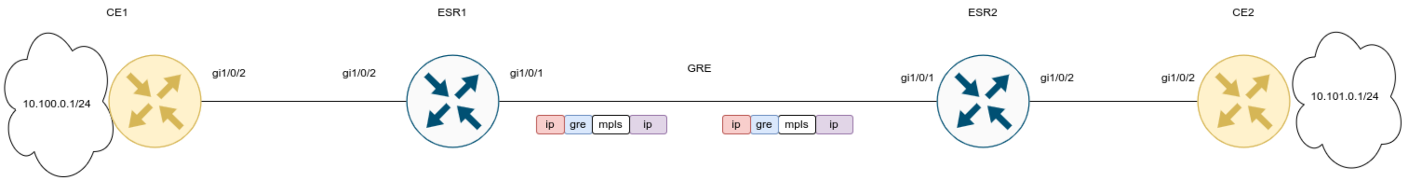
12.13.2 L3VPN

When configuring MTU on a tunnel, consider the following:

- **At least one mpls label will be present when passing through the tunnel. Accounting should include all labels in the stack, for example,** explicit null **or entropy label;**
- **If MTU of the outgoing interface is exceeded, the packet will be dropped (if GRE traffic** match fragment **is disabled);**
- **Control world is not supported;**
- **DF bit will be set to 1.**

Approximate packet structure is given in figure below:





Configure CE1 and CE2:

CE1

```
CE1(config)# hostname CE1
CE1(config)#
CE1(config)# route-map BGP_OUT
CE1(config-route-map)#   rule 1
CE1(config-route-map-rule)#   exit
CE1(config-route-map)# exit
CE1(config)# router bgp 65501
CE1(config-bgp)#   neighbor 10.10.0.2
CE1(config-bgp-neighbor)#     remote-as 65500
CE1(config-bgp-neighbor)#     address-family ipv4 unicast
CE1(config-bgp-neighbor-af)#       route-map BGP_OUT out
CE1(config-bgp-neighbor-af)#       enable
CE1(config-bgp-neighbor-af)#     exit
CE1(config-bgp-neighbor)#     enable
CE1(config-bgp-neighbor)#   exit
CE1(config-bgp)#   address-family ipv4 unicast
CE1(config-bgp-af)#     network 10.100.0.0/24
CE1(config-bgp-af)#   exit
CE1(config-bgp)#   enable
CE1(config-bgp)# exit
CE1(config)#
CE1(config)#
CE1(config)# interface gigabitethernet 1/0/2
CE1(config-if-gi)#   description "to ESR1"
CE1(config-if-gi)#   ip firewall disable
CE1(config-if-gi)#   ip address 10.10.0.1/30
CE1(config-if-gi)# exit
CE1(config)# interface loopback 1
CE1(config-loopback)#   ip address 10.100.0.1/24
CE1(config-loopback)# exit
```

CE2

```
CE2(config)# hostname CE2
CE2(config)#
CE2(config)# route-map BGP_OUT
CE2(config-route-map)#   rule 1
CE2(config-route-map-rule)#   exit
CE2(config-route-map)# exit
CE2(config)# router bgp 65502
CE2(config-bgp)#   neighbor 10.10.0.5
CE2(config-bgp-neighbor)#     remote-as 65500
CE2(config-bgp-neighbor)#     address-family ipv4 unicast
CE2(config-bgp-neighbor-af)#       route-map BGP_OUT out
CE2(config-bgp-neighbor-af)#       enable
CE2(config-bgp-neighbor-af)#     exit
CE2(config-bgp-neighbor)#     enable
CE2(config-bgp-neighbor)#   exit
CE2(config-bgp)#   address-family ipv4 unicast
CE2(config-bgp-af)#     network 10.101.0.0/24
CE2(config-bgp-af)#   exit
CE2(config-bgp)#   enable
CE2(config-bgp)# exit
CE2(config)#
CE2(config)#
CE2(config)# interface gigabitethernet 1/0/2
CE2(config-if-gi)#   description "to ESR2"
CE2(config-if-gi)#   ip firewall disable
CE2(config-if-gi)#   ip address 10.10.0.6/30
CE2(config-if-gi)# exit
CE2(config)# interface loopback 1
CE2(config-loopback)#   ip address 10.101.0.1/24
CE2(config-loopback)# exit
```

Configure ESR1 and ESR2:

ESR1

```
ESR1(config)# hostname ESR1
ESR1(config)#
ESR1(config)# ip vrf l3vpn_service
ESR1(config-vrf)#   ip protocols bgp max-routes 100
ESR1(config-vrf)#   rd 65500:1
ESR1(config-vrf)#   route-target export 65500:1
ESR1(config-vrf)#   route-target import 65500:1
ESR1(config-vrf)# exit
ESR1(config)#
ESR1(config)#
ESR1(config)# system cpu load-balance mpls passenger ip
ESR1(config)# security zone untrusted
ESR1(config-zone)# exit
ESR1(config)# security zone trusted
ESR1(config-zone)# exit
ESR1(config)#
ESR1(config)# route-map BGP_OUT
ESR1(config-route-map)#   rule 1
ESR1(config-route-map-rule)#   exit
ESR1(config-route-map)# exit
ESR1(config)# router bgp 65500
ESR1(config-bgp)#   router-id 10.12.0.1
ESR1(config-bgp)#   neighbor 10.12.0.2
ESR1(config-bgp-neighbor)#     remote-as 65500
ESR1(config-bgp-neighbor)#     update-source 10.12.0.1
ESR1(config-bgp-neighbor)#     address-family vpnv4 unicast
ESR1(config-bgp-neighbor-af)#       send-community extended
ESR1(config-bgp-neighbor-af)#       enable
ESR1(config-bgp-neighbor-af)#     exit
ESR1(config-bgp-neighbor)#     enable
ESR1(config-bgp-neighbor)#   exit
ESR1(config-bgp)#   enable
ESR1(config-bgp)#   vrf l3vpn_service
ESR1(config-bgp-vrf)#     neighbor 10.10.0.1
ESR1(config-bgp-vrf-neighbor)#       remote-as 65501
ESR1(config-bgp-vrf-neighbor)#       address-family ipv4 unicast
ESR1(config-bgp-neighbor-af-vrf)#         route-map BGP_OUT out
ESR1(config-bgp-neighbor-af-vrf)#         enable
ESR1(config-bgp-neighbor-af-vrf)#       exit
ESR1(config-bgp-vrf-neighbor)#       enable
ESR1(config-bgp-vrf-neighbor)#     exit
ESR1(config-bgp-vrf)#     address-family ipv4 unicast
ESR1(config-bgp-vrf-af)#       redistribute bgp 65500 route-map BGP_OUT
ESR1(config-bgp-vrf-af)#     exit
ESR1(config-bgp-vrf)#     enable
ESR1(config-bgp-vrf)#   exit
ESR1(config-bgp)# exit
ESR1(config)#
ESR1(config)# router ospf 1
ESR1(config-ospf)#   router-id 10.12.0.1
ESR1(config-ospf)#   area 0.0.0.0
ESR1(config-ospf-area)#     enable
ESR1(config-ospf-area)#   exit
ESR1(config-ospf)#   enable
ESR1(config-ospf)# exit
ESR1(config)#
```

```
ESR1(config)# interface gigabitethernet 1/0/1
ESR1(config-if-gi)#   security-zone untrusted
ESR1(config-if-gi)#   ip address 192.0.2.1/30
ESR1(config-if-gi)# exit
ESR1(config)# interface gigabitethernet 1/0/2
ESR1(config-if-gi)#   ip vrf forwarding l3vpn_service
ESR1(config-if-gi)#   description "from CE1"
ESR1(config-if-gi)#   ip firewall disable
ESR1(config-if-gi)#   ip address 10.10.0.2/30
ESR1(config-if-gi)# exit
ESR1(config)# interface loopback 1
ESR1(config-loopback)#   ip address 10.12.0.1/32
ESR1(config-loopback)#   ip ospf instance 1
ESR1(config-loopback)#   ip ospf
ESR1(config-loopback)# exit
ESR1(config)# tunnel gre 1
ESR1(config-gre)#   key 60
ESR1(config-gre)#   ttl 64
ESR1(config-gre)#   mtu 1472
ESR1(config-gre)#   ip firewall disable
ESR1(config-gre)#   local address 192.0.2.1
ESR1(config-gre)#   remote address 192.0.2.2
ESR1(config-gre)#   ip address 10.11.0.1/30
ESR1(config-gre)#   ip ospf instance 1
ESR1(config-gre)#   ip ospf
ESR1(config-gre)#   enable
ESR1(config-gre)# exit
ESR1(config)#
ESR1(config)# mpls
ESR1(config-mpls)#   ldp
ESR1(config-ldp)#     router-id 10.12.0.1
ESR1(config-ldp)#     address-family ipv4
ESR1(config-ldp-af-ipv4)#       interface gre 1
ESR1(config-ldp-af-ipv4-if)#       exit
ESR1(config-ldp-af-ipv4)#     exit
ESR1(config-ldp)#     enable
ESR1(config-ldp)#   exit
ESR1(config-mpls)#   forwarding interface gre 1
ESR1(config-mpls)# exit
ESR1(config)# security zone-pair untrusted self
ESR1(config-zone-pair)#   rule 1
ESR1(config-zone-pair-rule)#     action permit
ESR1(config-zone-pair-rule)#     match protocol gre
ESR1(config-zone-pair-rule)#     enable
ESR1(config-zone-pair-rule)#   exit
ESR1(config-zone-pair)# exit
```

```
ESR2(config)# hostname ESR2
ESR2(config)#
ESR2(config)# ip vrf l3vpn_service
ESR2(config-vrf)#   ip protocols bgp max-routes 100
ESR2(config-vrf)#   rd 65500:1
ESR2(config-vrf)#   route-target export 65500:1
ESR2(config-vrf)#   route-target import 65500:1
ESR2(config-vrf)# exit
ESR2(config)#
ESR2(config)#
ESR2(config)# system cpu load-balance mpls passenger ip
ESR2(config)# security zone untrusted
ESR2(config-zone)# exit
ESR2(config)# security zone trusted
ESR2(config-zone)# exit
ESR2(config)#
ESR2(config)# route-map BGP_OUT
ESR2(config-route-map)#   rule 1
ESR2(config-route-map-rule)#   exit
ESR2(config-route-map)# exit
ESR2(config)# router bgp 65500
ESR2(config-bgp)#   router-id 10.12.0.2
ESR2(config-bgp)#   neighbor 10.12.0.1
ESR2(config-bgp-neighbor)#     remote-as 65500
ESR2(config-bgp-neighbor)#     update-source 10.12.0.2
ESR2(config-bgp-neighbor)#     address-family vpnv4 unicast
ESR2(config-bgp-neighbor-af)#       send-community extended
ESR2(config-bgp-neighbor-af)#       enable
ESR2(config-bgp-neighbor-af)#     exit
ESR2(config-bgp-neighbor)#     enable
ESR2(config-bgp-neighbor)#   exit
ESR2(config-bgp)#   enable
ESR2(config-bgp)#   vrf l3vpn_service
ESR2(config-bgp-vrf)#     neighbor 10.10.0.6
ESR2(config-bgp-vrf-neighbor)#       remote-as 65502
ESR2(config-bgp-vrf-neighbor)#       address-family ipv4 unicast
ESR2(config-bgp-neighbor-af-vrf)#         route-map BGP_OUT out
ESR2(config-bgp-neighbor-af-vrf)#         enable
ESR2(config-bgp-neighbor-af-vrf)#       exit
ESR2(config-bgp-vrf-neighbor)#       enable
ESR2(config-bgp-vrf-neighbor)#     exit
ESR2(config-bgp-vrf)#     address-family ipv4 unicast
ESR2(config-bgp-vrf-af)#       redistribute bgp 65500 route-map BGP_OUT
ESR2(config-bgp-vrf-af)#     exit
ESR2(config-bgp-vrf)#     enable
ESR2(config-bgp-vrf)#   exit
ESR2(config-bgp)# exit
ESR2(config)#
ESR2(config)# router ospf 1
ESR2(config-ospf)#   router-id 10.12.0.2
ESR2(config-ospf)#   area 0.0.0.0
ESR2(config-ospf-area)#     enable
ESR2(config-ospf-area)#   exit
ESR2(config-ospf)#   enable
ESR2(config-ospf)# exit
ESR2(config)#
ESR2(config)# interface gigabitethernet 1/0/1
ESR2(config-if-gi)#   security-zone untrusted
```

```
ESR2(config-if-gi)#   ip address 192.0.2.2/30
ESR2(config-if-gi)# exit
ESR2(config)# interface gigabitethernet 1/0/2
ESR2(config-if-gi)#   ip vrf forwarding l3vpn_service
ESR2(config-if-gi)#   description "from CE2"
ESR2(config-if-gi)#   ip firewall disable
ESR2(config-if-gi)#   ip address 10.10.0.5/30
ESR2(config-if-gi)# exit
ESR2(config)# interface loopback 1
ESR2(config-loopback)#   ip address 10.12.0.2/32
ESR2(config-loopback)#   ip ospf instance 1
ESR2(config-loopback)#   ip ospf
ESR2(config-loopback)# exit
ESR2(config)# tunnel gre 1
ESR2(config-gre)#   key 60
ESR2(config-gre)#   ttl 64
ESR2(config-gre)#   mtu 1472
ESR2(config-gre)#   ip firewall disable
ESR2(config-gre)#   local address 192.0.2.2
ESR2(config-gre)#   remote address 192.0.2.1
ESR2(config-gre)#   ip address 10.11.0.2/30
ESR2(config-gre)#   ip ospf instance 1
ESR2(config-gre)#   ip ospf
ESR2(config-gre)#   enable
ESR2(config-gre)# exit
ESR2(config)#
ESR2(config)# mpls
ESR2(config-mpls)#   ldp
ESR2(config-ldp)#     router-id 10.12.0.2
ESR2(config-ldp)#     address-family ipv4
ESR2(config-ldp-af-ipv4)#       interface gre 1
ESR2(config-ldp-af-ipv4-if)#       exit
ESR2(config-ldp-af-ipv4)#     exit
ESR2(config-ldp)#     enable
ESR2(config-ldp)#   exit
ESR2(config-mpls)#   forwarding interface gre 1
ESR2(config-mpls)# exit
ESR2(config)# security zone-pair untrusted self
ESR2(config-zone-pair)#   rule 1
ESR2(config-zone-pair-rule)#     action permit
ESR2(config-zone-pair-rule)#     match protocol gre
ESR2(config-zone-pair-rule)#     enable
ESR2(config-zone-pair-rule)#   exit
ESR2(config-zone-pair)# exit
```

After completing the configuration, check the distribution of routing information and the network availability of the nodes:

```
*GRE tunnel configuration*
ESR2# sh tunnels configuration
Tunnel               State      Description
----------------     --------   ------------------------------
gre 1                Enabled    --

ESR2# sh tunnels configuration gre 1
State:                                              Enabled
Description:                                        --
Mode:                                               ip
Bridge group:                                       --
VRF:                                                --
Local address:                                      192.0.2.2
Remote address:                                     192.0.2.1
Calculates checksums for outgoing GRE packets:      No
Requires that all input GRE packets were checksum: No
key:                                                60
TTL:                                                64
DSCP:                                               Inherit
MTU:                                                1472
Path MTU discovery:                                 Enabled
Don't fragment bit suppression:                     Disabled
Security zone:                                      --
Multipoint mode:                                    Disabled
Keepalive:
    State:                                              Disabled
    Timeout:                                            10
    Retries:                                            6
    Destination address:                                --

*vpnv4 routes presence*
SR2# sh 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
----- --------------------- ------------------ --------------- ---------- ------- ----------
------ ----------------
*>    65500:1               10.101.0.0/24      --              --         34      100        --
65502 i
*>i   65500:1               10.100.0.0/24      10.12.0.1       --         16      100        0
65501 i

*LDP state*
ESR2# sh mpls ldp neighbor
Peer LDP ID: 10.12.0.1; Local LDP ID 10.12.0.2
    State:                     Operational
    TCP connection:            10.12.0.1:646 - 10.12.0.2:46444
    Messages sent/received: 60/60
    Uptime:                    00:53:59
    LDP discovery sources:
        gre 1
ESR2# sh mpls forwarding-table
Local    Outgoing Prefix                                     Outgoing          Next Hop
label    label    or tunnel ID                               Interface
-------- -------- ------------------------------------------ ----------------
---------------------------------------
```

```
35       imp-null 10.12.0.1/32                                gre 1            10.11.0.1

*Availability of nodes in the network*
CE2# ping 10.100.0.1 source ip 10.101.0.1 detailed
PING 10.100.0.1 (10.100.0.1) from 10.101.0.1 : 56 bytes of data.
64 bytes from 10.100.0.1: icmp_seq=1 ttl=0 time=1.32 ms
64 bytes from 10.100.0.1: icmp_seq=2 ttl=0 time=1.12 ms
64 bytes from 10.100.0.1: icmp_seq=3 ttl=0 time=1.14 ms
64 bytes from 10.100.0.1: icmp_seq=4 ttl=0 time=1.09 ms
64 bytes from 10.100.0.1: icmp_seq=5 ttl=0 time=1.15 ms
```

13 Security management

13.1 AAA configuration

AAA (Authentication, Authorization, Accounting) is used for description of access provisioning and control.

- Authentication is a matching of a person (request) for the existing account in the security system. Performed by the login and password.
- Authorization (authorization, privilege verification, access level verification) is a matching of the existing account in the system (passed authentication) and specific privileges.
- Accounting (accounting) is a monitoring of user connection or changes made by the user.

13.1.1 Local authentication configuration algorithm

Step	Description	Command	Keys
1	Set local as authentication method.	**esr(config)# aaa authentication login { default | <NAME> } <METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]**	<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.
2	Set enable as authentication method of user privileges elevation.	**esr(config)# aaa authentication enable <NAME><METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]**	<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.
3	Set the method for iterating over authentication methods (optional).	**esr(config)# aaa authentication mode <MODE>**	<MODE> – options of iterating over methods: • chain – if the server returned FAIL, proceed to the following authentication method in the chain; • break – if the server returned FAIL, abandon authentication attempts. If the server is unavailable, continue authentication attempts by the following methods in the chain. Default value: chain.

Step	Description	Command	Keys
4	Specify the number of failed authentication attempts to block the user login and time of the lock (optional)	**esr(config)# aaa authentication attempts max-fail <COUNT> <TIME>**	<COUNT> – amount of failed authentication attempts after which a user is blocked, takes the values of [1..65535]; <TIME> – user blocking time in minutes, takes the values of [1..65535]. Default value: <COUNT> – 5; <TIME> – 300
5	Enable request for change the default password for the 'admin' user (optional)	**esr(config)# security passwords default-expired**	
6	Enable the inhibit mode on the use of previously set local user passwords (optional)	**esr(config)# security passwords history <COUNT>**	<COUNT> – number of passwords saved in the router memory. Takes values in the range of [1..15]. Default value: 0
7	Set the lifetime of local user password (optional)	**esr(config)# security passwords lifetime <TIME>**	<TIME> – password lifetime in days. Takes values in the range of [1..365]. Default: The lifetime of local user password is unlimited.
8	Set a limit on the minimum length of local user password and ENABLE password (optional)	**esr(config)# security passwords min-length <NUM>**	<NUM> – minimum number of characters in the password. Takes values in the range of [8..128]. Default value: 0
9	Set a limit on the maximum length of local user password and ENABLE password (optional)	**esr(config)# security passwords max-length <NUM>**	<NUM> – maximum number of characters in the password. Takes values in the range of [8..128]. Default value: no limit.
10	Set the minimum number of character types that must be present in the local user password and ENABLE password (optional)	**esr(config)# security passwords symbol-types <COUNT>**	<COUNT> – minimum number of character types in the password. Takes values in the range of [1..4]. Default value: 1

Step	Description	Command	Keys
11	Set the minimum number of lower case letters in the local user password and ENABLE password (optional)	**esr(config)# security passwords lower-case <COUNT>**	<COUNT> – minimum number of lower case letters in the local user password and ENABLE password. Takes values in the range of [0..128]. Default value: 0
12	Set the minimum number of upper case letters in the local user password and ENABLE password (optional)	**esr(config)# security passwords upper-case <COUNT>**	<COUNT> – minimum number of upper case letters in the password. Takes values in the range of [0..128]. Default value: 0
13	Set the minimum number of digits in the local user password and ENABLE password (optional)	**esr(config)# security passwords numeric-count <COUNT>**	<COUNT> – minimum number of digits in the password. Takes values in the range of [0..128]. Default value: 0
14	Set the minimum number of special characters in the local user password and ENABLE password (optional)	**esr(config)# security passwords special-case <COUNT>**	<COUNT> – minimum number of special characters in the password. Takes values in the range of [0..128]. Default value: 0
15	Add user in the local database and switch to the user parameters configuration mode	**esr(config)# username <NAME>**	<NAME> – user name, set by the string of up to 31 characters.
16	Set user password	**esr(config-user)# password { <CLEAR-TEXT> | encrypted <HASH_SHA512> }**	<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.
17	Set user privileges level	**esr(config-user)# privilege <PRIV>**	<PRIV> – required privilege level. Takes values in the range of [1..15].

Step	Description	Command	Keys
18	Switch to the corresponding terminal configuration mode	**esr(config)# line console** or **esr(config)# line telnet** or **esr(config)# line ssh**	
19	Activate user login authentication list	**esr(config-line-ssh)# login authentication <NAME>**	<NAME> – list name, set by the string of up to 31 characters.
20	Activate authentication list of user privileges elevation	**esr(config-line-ssh)# enable authentication <NAME>**	<NAME> – list name, set by the string of up to 31 characters.
21	Set the interval after which the idle session will be terminated	**esr(config-line-ssh)# exec-timeout <SEC>**	<SEC> – time interval in minutes, takes values of [1..65535].

13.1.2 AAA configuration algorithm via RADIUS

Step	Description	Command	Keys
1	Set the DSCP code global value for the use in IP headers of RADIUS server egress packets (optional).	**esr(config)# radius-server dscp <DSCP>**	<DSCP> – DSCP code value, takes values in the range of [0..63]. Default value: 63.
2	Set the global number of re-requests to the last active RADIUS server (optional).	**esr(config)# radius-server retransmit <COUNT>**	<COUNT> – amount of iterative requests to RADIUS server, takes values of [1..10]. Default value: 1.
3	Set the global value of the interval after which the router assumes that the RADIUS server is not available (optional).	**esr(config)# radius-server timeout <SEC>**	<SEC> – time interval in seconds, takes values of [1..30]. Default value: 3 seconds.

Step	Description	Command	Keys
4	Add RADIUS server to the list of used servers and switch to its configuration mode.	**esr(config)# radius-server host { <IP-ADDR> | <IPV6-ADDR> } [vrf <VRF>]** **esr(config-radius-server)#**	<IP-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] <VRF> – VRF instance name, set by the string of up to 31 characters.
5	Specify the number of failed authentication attempts to block the user login and time of the lock (optional).	**aaa authentication attempts max-fail <COUNT> <TIME>**	<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: <COUNT> – 5; <TIME> – 300
6	Set the password for authentication on remote RADIUS server.	**esr(config-radius-server)# key ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> }**	<TEXT> – string [8..16] ASCII characters; <ENCRYPTED-TEXT> – encrypted password, [8..16] bytes size, set by the string of [16..32] characters.
7	Set the priority for using a remote RADIUS server (optional).	**esr(config-radius-server)# priority <PRIORITY>**	<PRIORITY> – remote server priority, takes values in the range of [1..65535]. The lower value, the higher the priority of server is. Default value: 1.
8	Set the interval after which the router assumes that the RADIUS server is not available (optional).	**esr(config-radius-server)# timeout <SEC>**	<SEC> – time interval in seconds, takes values of [1..30]. Default value: global timer value is used.

Step	Description	Command	Keys
9	Set IPv4/IPv6 address that will be used as source IPv4/IPv6 address in transmitted RADIUS packets.	**esr(config-radius-server)# source-address { <ADDR> | <IPV6-ADDR> }**	<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].
10	Specify the interface or tunnel of the router which IPv4/IPv6 address will be used as the source IPv4/IPv6 address in outgoing RADIUS packets.	**esr(config-radius-server)# source-interface { <IF> | <TUN> }**	<IF> – interface specified in form given in the Types and naming procedure of router interfaces section of CLI command reference guide. <TUN> – tunnel name specified in form given in the Types and naming procedure of router tunnels section.
11	Set radius as authentication method.	**esr(config)# aaa authentication login { default | <NAME> } <METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]**	<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.
12	Set radius as authentication method of user privileges elevation.	**esr(config)# aaa authentication enable <NAME><METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]**	<NAME> – list name, set by the 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.

Step	Description	Command	Keys
13	Specify authentication methods to be tried in case of failure (optional).	**esr(config)# aaa authentication mode <MODE>**	<MODE> – options of iterating over methods: • chain – if the server returned FAIL, proceed to the following authentication method in the chain; • break – if the server returned FAIL, abandon authentication attempts. If the server is unavailable, continue authentication attempts by the following methods in the chain. Default value: chain.
14	Configure radius in the list of user session accounting methods (optional).	**esr(config)# aaa accounting login start-stop <METHOD 1> [<METHOD 2>]**	<METHOD> – accounting methods: • tacacs – session accounting by TACACS; • radius – session accounting by RADIUS.
15	Switch to the corresponding terminal configuration mode.	**esr(config)# line <TYPE>**	<TYPE> – console type: • console – local console; • ssh – secure remote console.
16	Activate user login authentication list.	**esr(config-line-console)# login authentication <NAME>**	<NAME> – list name, set by the string of up to 31 characters. Created in step 8.
17	Activate authentication list of user privileges elevation.	**esr(config-line-console)# enable authentication <NAME>**	<NAME> – list name, set by the string of up to 31 characters. Created in step 9.

13.1.3 AAA configuration algorithm via TACACS

Step	Description	Command	Keys
1	Set the DSCP code global value for the use in IP headers of TACACS server egress packets (optional).	**esr(config)# tacacs-server dscp <DSCP>**	<DSCP> – DSCP code value, takes values in the range of [0..63]. Default value: 63.
2	Set the global value of the interval after which the router assumes that the TACACS server is not available (optional).	**esr(config)# tacacs-server timeout <SEC>**	<SEC> – time interval in seconds, takes values of [1..30]. Default value: 3 seconds.

Step	Description	Command	Keys
3	Add TACACS server to the list of used servers and switch to its configuration mode.	**esr(config)# tacacs -server host { <IP-ADDR> | <IPV6-ADDR> } [vrf <VRF>]** **esr(config-tacacs-server)#**	<IP-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] <VRF> – VRF instance name, set by the string of up to 31 characters.
4	Specify the number of failed authentication attempts to block the user login and time of the lock (optional).	**aaa authentication attempts max-fail <COUNT> <TIME>**	<COUNT> – amount of failed authentication attempts after which a user is blocked, takes the values of [1..65535]; <TIME> – user blocking time in minutes, takes the values of [1..65535]. Default value: <COUNT> – 5; <TIME> – 300
5	Set the password for authentication on remote TACACS server.	**esr(config-tacacs-server)# key ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> }**	<TEXT> – string [8..16] ASCII characters; <ENCRYPTED-TEXT> – encrypted password, [8..16] bytes size, set by the string of [16..32] characters.
6	Set the port number to communicate with remote TACACS server (optional).	**esr(config-tacacs-server)# port <PORT>**	<PORT> – number of TCP port to exchange data with a remote server, takes values of [1..65535]. Default value: 49 for TACACS server.
7	Set the priority for using a remote TACACS server (optional).	**esr(config-tacacs-server)# priority <PRIORITY>**	<PRIORITY> – remote server priority, takes values in the range of [1..65535]. The lower value, the higher the priority of server is. Default value: 1.
8	Set IPv4/IPv6 address that will be used as source IPv4/IPv6 address in transmitted TACACS packets.	**esr(config-tacacs-server)# source-address { <ADDR> | <IPV6-ADDR> }**	<ADDR> – source IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255];

Step	Description	Command	Keys
9	Specify the interface or tunnel of the router whose IPv4/IPv6 address will be used as the source IPv4/IPv6 address in outgoing TACACS packets.	**esr(config-tacacs-server)# source-interface { <IF> | <TUN> }**	<IF> – interface specified in form given in the Types and naming procedure of router interfaces section of CLI command reference guide. <TUN> – tunnel name specified in form given in the Types and naming procedure of router tunnels section.
10	Set TACACS as authentication method of user privileges elevation.	**esr(config)# aaa authentication enable <NAME><METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]**	<NAME> – list name, set by the 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.
11	Set the method for iterating over authentication methods (optional).	**esr(config)# aaa authentication mode <MODE>**	<MODE> – options of iterating over methods: • chain – if the server returned FAIL, proceed to the following authentication method in the chain; • break – if the server returned FAIL, abandon authentication attempts. If the server is unavailable, continue authentication attempts by the following methods in the chain. Default value: chain.
12	Configure the list of CLI commands accounting methods (optional).	**esr(config)# aaa accounting commands stop-only tacacs**	
13	Configure tacacs in the list of user session accounting methods (optional).	**esr(config)# aaa accounting login start-stop <METHOD 1> [<METHOD 2>]**	<METHOD> – accounting methods: • tacacs – session accounting by TACACS; • radius – session accounting by RADIUS.

Step	Description	Command	Keys
14	Switch to the corresponding terminal configuration mode.	**esr(config)# line <TYPE>**	<TYPE> – console type: • console – local console; • ssh – secure remote console.
15	Activate user login authentication list.	**esr(config-line-console)# login authentication <NAME>**	<NAME> – list name, set by the string of up to 31 characters. Created in step 7.
16	Activate authentication list of user privileges elevation.	**esr(config-line-console)# enable authentication <NAME>**	<NAME> – list name, set by the string of up to 31 characters. Created in step 8.

13.1.4 AAA configuration algorithm via LDAP

Step	Description	Command	Keys
1	Specify basic DN (Distinguished name) which will be used when searching for users.	**esr(config)# ldap-server base-dn <NAME>**	<NAME> – basic DN, set by the string of up to 255 characters.
2	Set the interval after which the router assumes that the LDAP server is not available (optional).	**esr(config)# ldap-server bind timeout <SEC>**	<SEC> – time interval in seconds, takes values of [1..30]. Default value: 3 seconds.
3	Specify the DN (Distinguished name) of a user with administrator rights, under which authorization will take place on the LDAP server when searching for users.	**esr(config)# ldap-server bind authenticate root-dn <NAME>**	<NAME> – DN of a user with administration rights, set by the string of up to 255 characters.
4	Specify the password of a user with administrator rights, under which authorization will take place on the LDAP server when searching for users.	**esr(config)# ldap-server bind authenticate root-password ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> }**	<TEXT> – string [8..16] ASCII characters; <ENCRYPTED-TEXT> – encrypted password, [8..16] bytes size, set by the string of [16..32] characters.
5	Specify a class name of the objects among which it is necessary to search for users on LDAP server (optional).	**esr(config)# ldap-server search filter user-object-class <NAME>**	<NAME> – object class name, set by the string of up to 127 characters. Default value: posixAccount.

Step	Description	Command	Keys
6	Specify the user search scope in LDAP server tree (optional).	**esr(config)# ldap-server search scope <SCOPE>**	<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.
7	Specify the interval after which the device assumes that LDAP server has not found users entries satisfying the search condition (optional).	**esr(config)# ldap-server search timeout <SEC>**	<SEC> – time interval in seconds, takes values of [0..30] Default value: 0 – device is waiting for search completion and response from LDAP server.
8	Specify an attribute name of the object which is compared with the name of the desired user on LDAP server (optional).	**esr(config)# ldap-server naming-attribute <NAME>**	<NAME> – object attribute name, set by the string of up to 127 characters. Default value: uid.
9	Specify the object attribute name which is compared with the name of a desired user on LDAP server (optional).	**esr(config)# ldap-server privilege-level-attribute <NAME>**	<NAME> – object attribute name, set by the string of up to 127 characters. Default value: priv-lvl
10	Set the DSCP code global value for the use in IP headers of LDAP server egress packets (optional).	**esr(config)# ldap-server dscp <DSCP>**	<DSCP> – DSCP code value, takes values in the range of [0..63]. Default value: 63
11	Add LDAP server to the list of used servers and switch to its configuration mode.	**esr(config)# ldap -server host { <IP-ADDR> | <IPV6-ADDR> } [vrf <VRF>]** **esr(config-ldap-server)#**	<IP-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] <VRF> – VRF instance name, set by the string of up to 31 characters.

Step	Description	Command	Keys
12	Specify the number of failed authentication attempts to block the user login and time of the lock (optional)	**aaa authentication attempts max-fail <COUNT> <TIME>**	<COUNT> – amount of failed authentication attempts after which a user is blocked, takes the values of [1..65535]; <TIME> – user blocking time in minutes, takes the values of [1..65535]. Default value: <COUNT> – 5; <TIME> – 300
13	Set the port number to communicate with remote LDAP server (optional).	**esr(config-ldap-server)# port <PORT>**	<PORT> – number of TCP port to exchange data with a remote server, takes values of [1..65535]. Default value: 389 for LDAP server.
14	Prioritize the use of a remote LDAP server (optional).	**esr(config-ldap-server)# priority <PRIORITY>**	<PRIORITY> – remote server priority, takes values in the range of [1..65535]. The lower value, the higher the priority of server is. Default value: 1.
15	Set IPv4/IPv6 address that will be used as source IPv4/IPv6 address in transmitted LDAP packets.	**esr(config-ldap-server)# source-address { <ADDR> | <IPV6-ADDR> }**	<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].
16	Set router interface or tunnel, IPv4/IPv6 address of which will be used as IPv4/IPv6 source address in sent LDAP packets.	**esr(config-ldap-server)# source-interface { <IF> | <TUN> }**	<IF> – interface specified in form given in the Types and naming procedure of router interfaces section of CLI command reference guide. <TUN> – tunnel name specified in form given in the Types and naming procedure of router tunnels section.

Step	Description	Command	Keys
17	Set LDAP as authentication method.	**esr(config)# aaa authentication login { default | <NAME> } <METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]**	<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.
18	Set LDAP as authentication method of user privileges elevation.	**esr(config)# aaa authentication enable <NAME> <METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]**	<NAME> – list name, set by the 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.
19	Set the method for iterating over authentication methods.	**esr(config)# aaa authentication mode <MODE>**	<MODE> – options of iterating over methods: • chain – if the server returned FAIL, proceed to the following authentication method in the chain; • break – if the server returned FAIL, abandon authentication attempts. If the server is unavailable, continue authentication attempts by the following methods in the chain. Default value: chain.
20	Switch to the corresponding terminal configuration mode.	**esr(config)# line <TYPE>**	<TYPE> – console type: • console – local console; • ssh – secure remote console.
22	Activate user login authentication list.	**esr(config-line-console)# login authentication <NAME>**	<NAME> – list name, set by the string of up to 31 characters. Created in step 14.

Step	Description	Command	Keys
22	Activate authentication list of user privileges elevation.	**esr(config-line-console)# enable authentication <NAME>**	<NAME> – list name, set by the string of up to 31 characters. Created in step 15.

13.1.5 Example of authentication configuration using telnet via RADIUS server

Objective:

Configure authentication for users connected via Telnet and RADIUS (192.168.16.1/24).

Solution:

Configure connection to RADIUS server and specify the key (password):

```
esr# configure
esr(config)# radius-server host 192.168.16.1
esr(config-radius-server)# key ascii-text encrypted 8CB5107EA7005AFF
esr(config-radius-server)# exit
```

Create authentication profile:

```
esr(config)# aaa authentication login log radius
```

Specify authentication mode used for Telnet protocol connection:

```
esr(config)# line telnet
esr(config-line-telnet)# login authentication log
esr(config-line-telnet)# exit
esr(config)# exit
```

To view the information on RADIUS server connection settings, use the following command:

```
esr# show aaa radius-servers
```

To view the authentication profiles, use the following command:

```
esr# show aaa authentication
```

13.2 Command privilege configuration

Command privilege configuration is a flexible tool that allows you to assign baseline user privilege level (1–15) to a command set. In future, you may specify privilege level during user creation which will define a command set available to them.

- *Levels 1-9* enable all monitoring commands (show …);
- *Levels 10-14* enable all commands except for device reboot, user management and other specific commands;
- *Level 15* enables all monitoring commands.

13.2.1 Configuration algorithm

To change minimum privilege level required for CLI command execution, use the following command:

esr(config)# privilege <COMMAND-MODE> level <PRIV><COMMAND>

<COMMAND-MODE> – command mode;

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

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

13.2.2 Example of command privilege configuration

Objective:

Transfer all interface information display commands to the privilege level 10 except for 'show interfaces bridges' command. Transfer 'show interfaces bridges' command to the privilege level 3.

Solution:

In configuration mode, identify commands enabled for operation under privilege level 10 and privilege level 3:

```
esr(config)# privilege root level 3 "show interfaces bridge"
esr(config)# privilege root level 10 "show interfaces"
```

13.3 Logging and network attacks protection configuration

13.3.1 Configuration algorithm

Step	Description	Command	Keys
1	Enable protection against ICMP flood attacks.	**esr(config)# ip firewall screen dos-defense icmp-threshold { <NUM> }**	<NUM> – amount of ICMP packets per second, set in the range of [1..10000]
2	Enable protection against land attacks.	**esr(config)# firewall screen dos-defense land**	
3	Enable a limit on the number of packets sent per second per destination address	**esr(config)# ip firewall screen dos-defense limit-session-destination { <NUM> }**	<NUM> – limit number of IP packets per second, set in the range of [1..10000].
4	Enable a limit on the number of packets sent per second per source address	**esr(config)# ip firewall screen dos-defense limit-session-source { <NUM> }**	<NUM> – limit number of IP Packets per second, set in the range of [1..10000].

Step	Description	Command	Keys
5	Enable protection against SYN flood attacks.	**esr(config)# ip firewall screen dos-defense syn-flood { <NUM> } [src-dsr]**	<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.
6	Enable protection against UDP flood attacks.	**esr(config)# ip firewall screen dos-defense udp-threshold { <NUM> }**	<NUM> – maximum amount of UDP packets per second, set in the range of [1..10000].
7	Enable protection against winnuke attacks.	**esr(config)# ip firewall screen dos-defense winnuke**	
8	Enable blocking of TCP packets with the FIN flag set and the ACK flag not set.	**esr(config)# ip firewall screen spy-blocking fin-no-ack**	
9	Enable blocking of various type ICMP packets.	**esr(config)# ip firewall screen spy-blocking icmp-type**	<TYPE> – ICMP type, may take the following values: • destination-unreachable • echo-request • reserved • source-quench • time-exceeded
10	Enable protection against IP sweep attacks.	**esr(config)# ip firewall screen spy-blocking ip-sweep { <NUM> }**	<NUM> – ip sweep attack detection time, set in milliseconds [1..1000000].
11	Enable protection against port scan attacks.	**esr(config)# ip firewall screen spy-blocking port-scan { <threshold> } [<TIME>]**	<threshold> – interval in milliseconds during which the port scan attack will be recorded [1..1000000]. <TIME> – blocking time in milliseconds [1..1000000].
12	Enable protection against IP spoofing attacks.	**esr(config)# ip firewall screen spy-blocking spoofing**	
13	Enable blocking of TCP packets, with the SYN and FIN flags set.	**esr(config)# ip firewall screen spy-blocking syn-fin**	

Step	Description	Command	Keys
14	Enable blocking of TCP packets, with all flags or with the set of flags: FIN, PSH, URG. The given command provides the protection against XMAS attack.	**esr(config)# ip firewall screen spy-blocking tcp-all-flag**	
15	Enable blocking of TCP packets, with the zero 'flags' field.	**esr(config)# ip firewall screen spy-blocking tcp-no-flag**	
16	Enable blocking of fragmented ICMP packets.	**esr(config)# ip firewall screen suspicious-packets icmp-fragment**	
17	Enable blocking of fragmented IP packets.	**esr(config)# ip firewall screen suspicious-packets ip-fragment**	
18	Enable blocking of ICMP packets more than 1024 bytes.	**esr(config)# ip firewall screen suspicious-packets icmp-fragment**	
19	Enable blocking of fragmented TCP packets, with the SYN flag.	**esr(config)# ip firewall screen suspicious-packets syn-fragment**	
20	Enable blocking of fragmented UDP packets.	**esr(config)# ip firewall screen suspicious-packets udp-fragment**	
21	Enable blocking of packets with the protocol ID contained in IP header equal to 137 and more.	**esr(config)# ip firewall screen suspicious-packets unknown-protocols**	
22	Set the frequency of notification (via SNMP, syslog and in CLI) of detected and blocked network attacks.	**esr(config)# ip firewall logging interval <NUM>**	<NUM> – time interval in seconds [30 .. 2147483647]
23	Enable more detailed message output about detected and blocked network attacks in the CLI.	**esr(config)# logging firewall screen detailed**	
24	Enable mechanism of DoS attacks detection and logging via CLI, syslog and SNMP.	**esr(config)# logging firewall screen dos-defense <ATACK_TYPE>**	<ATACK_TYPE> – DoS attack type, takes the following values: icmp-threshold, land, limit-session-destination, limit-session-source, syn-flood, udp-threshold, winnuke.

Step	Description	Command	Keys
25	Enable mechanism of espionage activity detection and logging via CLI, syslog and SNMP.	**esr(config)# logging firewall screen spy-blocking { <ATACK_TYPE> | icmp-type <ICMP_TYPE> }**	<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 the following values: destination-unreachable, echo-request, reserved, source-quench, time-exceeded.
26	Enable mechanism of specialized packets detection and logging via CLI, syslog and SNMP.	**esr(config)# logging firewall screen suspicious-packets <PACKET_TYPE>**	<PACKET_TYPE> – non-standard packets type, takes the following values: icmp-fragment, ip-fragment, large-icmp, syn-fragment, udp-fragment, unknown-protocols.

13.3.2 Description of attack protection mechanisms

Command	Description
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 attack leads to the host reboot and its failure due to the necessity to process each query and respond to it.
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 attack leads to the host reboot and its failure due to the necessity to process each TCP SYN packet and the attempts of the host to establish a TCP session with itself.
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.
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.

Command	Description
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 attack leads to the host reboot and its failure due to the necessity to process each TCP SYN packet and the attempts to establish a TCP session.
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 attack lead to the host reboot and its failure due to the massive UDP traffic.
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 attack leads to the older Windows versions (up to 95 version) failure.
ip firewall screen spy-blocking fin-no-ack	The given command enables the blocking of TCP packets with the FIN flag set and the ACK flag not set. These packets are specialized and it is possible to determine a victim operational system by the respond.
ip firewall screen spy-blocking icmp-type destination-unreachable	The given command enables the blocking of all 3 type ICMP packets (destination-unreachable) including the packets generated by the router itself. The protection prevents an attacker from learning about network topology and hosts availability.
ip firewall screen spy-blocking icmp-type echo-request	The given command enables the blocking of all 8 type ICMP packets (echo-request) including the packets generated by the router itself. The protection prevents an attacker from learning about network topology and hosts availability.
ip firewall screen spy-blocking icmp-type reserved	The given command enables the blocking of all 2 and 7 type ICMP packets (reserved) including the packets generated by the router itself. The protection prevents an attacker from learning about network topology and hosts availability.
ip firewall screen spy-blocking icmp-type source-quench	The given command enables the blocking of all 4 type ICMP packets (source quench) including the packets generated by the router itself. The protection prevents an attacker from learning about network topology and hosts availability.
ip firewall screen spy-blocking icmp-type time-exceeded	The given command enables the blocking of all 11 type ICMP packets (time exceeded) including the packets generated by the router itself. The protection prevents an attacker from learning about network topology and hosts availability.

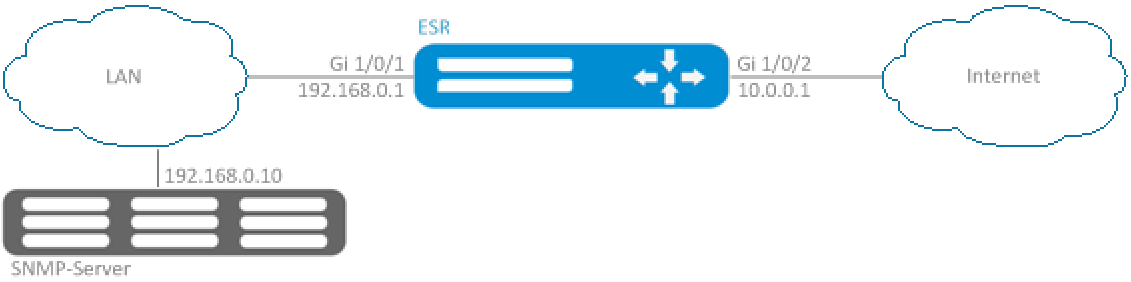
Command	Description
ip firewall screen spy-blocking ip-sweep	This command enables the protection against IP-sweep attacks. When the protection is enabled, if more than 10 ICMP queries from one source arrive within the specified interval, the first 10 queries are dropped by the router and 11th with the following ones are discarded for the remaining interval time. The protection prevents an attacker from learning about network topology and hosts availability.
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 behavior is recorded as port scan attack and all the following packets of that type are blocked for the second specified time interval (<TIME>). An attacker will not be able to scan the device open ports quickly.
ip firewall screen spy-blocking spoofing	The given 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. Protects from network intrusions with replaced source IP addresses.
ip firewall screen spy-blocking syn-fin	The given command enables the blocking of TCP packets, with the SYN and FIN flags set. These packets are specialized and it is possible to determine a victim operational system by the respond.
ip firewall screen spy-blocking tcp-all-flag	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.
ip firewall screen spy-blocking tcp-no-flag	This command enables the blocking of TCP packets with the zero 'flags' field. These packets are specialized and it is possible to determine a victim operational system by the respond.
ip firewall screen suspicious-packets icmp-fragment	The given command enables the blocking of fragmented ICMP packets. ICMP packets are usually small and there is no need to fragment them.
ip firewall screen suspicious-packets ip-fragment	The given command enables the blocking of fragmented packets.
ip firewall screen suspicious-packets large-icmp	The given command enables the blocking of ICMP packets more than 1024 bytes.
ip firewall screen suspicious-packets syn-fragment	This command enables the blocking of fragmented TCP packets with the SYN flag. TCP packets with the SYN flag are usually small and there is no need to fragment them. The protection prevents concentration of fragmented packets in a buffer.

Command	Description
ip firewall screen suspicious-packets udp-fragment	The given command enables the blocking of fragmented UDP packets.
ip firewall screen suspicious-packets unknown-protocols	The given command enables the blocking of packets, with the protocol ID contained in IP header equal to 137 and more.

13.3.3 Configuration example of logging and protection against network attacks

Objective:

Protect LAN and ESR router from land, syn-flood, ICMP flood network attacks and configure the notification of attacks by SNMP to SNMP server 192.168.0.10.



Solution:

First, configure interfaces and firewall (firewall configuration or its absence will not affect the operation of network attacks protection):

```
esr(config)# security zone LAN
esr(config-zone)# exit
esr(config)# security zone WAN
esr(config-zone)# exit
esr(config)# security zone-pair LAN WAN
esr(config-zone-pair)# rule 100
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# ex
esr(config-zone-pair)# exit
esr(config)# security zone-pair WAN LAN
esr(config-zone-pair)# rule 100
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# security-zone LAN
esr(config-if-gi)# ip address 192.168.0.1/24
esr(config-if-gi)# exit
esr(config)# interface gigabitethernet 1/0/2
esr(config-if-gi)# security-zone WAN
esr(config-if-gi)# ip address 10.0.0.1/24
esr(config-if-gi)# exit
```

Enable protection against land, syn-flood, ICMP flood attacks:

```
esr(config)# ip firewall screen dos-defense land
esr(config)# ip firewall screen dos-defense syn-flood 100 src-dst
esr(config)# ip firewall screen dos-defense icmp-threshold 100
```

Configure logging of detected attacks:

```
esr(config)# firewall logging screen dos-defense land
esr(config)# firewall logging screen dos-defense syn-flood
esr(config)# firewall logging screen dos-defense icmp-threshold
```

Configure SNMP server to which the traps will be sent:

```
esr(config)# snmp-server
esr(config)# snmp-server host 192.168.0.10
esr(config)# snmp-server enable traps screen land
esr(config)# snmp-server enable traps screen syn-flood
esr(config)# snmp-server enable traps screen icmp-threshold
```

To view the statistics on recorded network attacks, use the following command:

```
esr# show ip firewall screen counters
```

13.4 Firewall configuration

Firewall is a package of hardware or software tools that allows for control and filtering of transmitted network packets in accordance with the defined rules.

13.4.1 Configuration algorithm

Step	Description	Command	Keys
1	Create security zones.	**esr(config)# security zone <zone-name1>** **esr(config)# security zone <zone-name2>**	<zone-name> – up to 12 characters. Names all, any and self are reserved.
2	Specify a security zone description.	**esr(config-zone)# description <description>**	<description> – up to 255 characters..
3	Specify VRF instance, in which the given security zone will operate (optional).	**esr(config- zone)# ip vrf forwarding <VRF>**	<VRF> – VRF name, set by the string of up to 31 characters.
4	Enable session counters for NAT and Firewall (optional, may reduce the performance).	**esr(config)# ip firewall sessions counters**	

Step	Description	Command	Keys
5	Disable filtration of packets for which it was not possible to determine belonging to any known connection and which are not the beginning of a new connection (optional, may reduce the performance).	**esr(config)# ip firewall sessions allow-unknown**	
6	Select firewall operation mode (optional). In stateful mode, only the first packet of the session is checked, and if direct traffic is allowed, return traffic is allowed automatically. In stateless mode, each packet is checked. Direct and return traffic must be allowed in the corresponding zone-pair (see step 29). Firewall operation according to the list of applications is possible only in stateless mode.	**esr(config)# ip firewall mode <MODE>**	<MODE> – firewall operation mode, may take the following values: stateful, stateless. Default value: stateful.
7	Determine the session lifetime for unsupported protocols (optional).	**esr(config)# ip firewall sessions generic-timeout <TIME>**	<TIME> – session lifetime for unsupported protocols, takes values in seconds [1..8553600]. Default value: 60 seconds.
8	Determine ICMP session lifetime after which it is considered to be outdated (optional).	**esr(config)# ip firewall sessions icmp-timeout <TIME>**	<TIME> – ICMP session lifetime, takes values in seconds [1..8553600]. Default value: 30 seconds.
9	Determine ICMPv6 session lifetime after which it is considered to be outdated (optional).	**esr(config)# ip firewall sessions icmpv6-timeout <TIME>**	<TIME> – ICMP session lifetime, takes values in seconds [1..8553600]. Default value: 30 seconds.
10	Determine the size of outstanding sessions table (optional).	**esr(config)# ip firewall sessions max-expect <COUNT>**	<COUNT> – table size, takes values of [1..8553600]. Default value: 256.
11	Determine the size of trackable sessions table (optional).	**esr(config)# ip firewall sessions max-tracking <COUNT>**	<COUNT> – table size, takes values of [1..8553600]. Default value: 512000.

Step	Description	Command	Keys
12	Determine the lifetime of TCP session in 'connection is being established' state after which it is considered to be outdated (optional).	**esr(config)# ip firewall sessions tcp-connect-timeout <TIME>**	<TIME> – lifetime of TCP session in 'connection is being established' state, takes values in seconds [1..8553600]. Default value: 60 seconds.
13	Determine the lifetime of TCP session in 'connection is being closed' state after which it is considered to be outdated (optional).	**esr(config)# ip firewall sessions tcp-disconnect-timeout <TIME>**	<TIME> – lifetime of TCP session in 'connection is being closed' state, takes values in seconds [1..8553600]. Default value: 30 seconds.
14	Determine the lifetime of TCP session in 'connection is being established' state after which it is considered to be outdated (optional).	**esr(config)# ip firewall sessions tcp-established-timeout <TIME>**	<TIME> – lifetime of TCP session in 'connection is being established' state, takes values in seconds [1..8553600]. Default value: 120 seconds.
15	Determine the timeout after which the closed TCP session is actually deleted from the table of trackable sessions (optional).	**esr(config)# ip firewall sessions tcp-latecome-timeout <TIME>**	<TIME> – timeout, takes value in seconds [1..8553600]. Default value: 120 seconds.
16	Enable application-level session tracking for certain protocols (optional).	**esr(config)# ip firewall sessions tracking**	<PROTOCOL> – application-level protocol [ftp, h323, pptp, netbios-ns, tftp] sessions of which should be tracked. <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. By default – disabled for all protocols.

Step	Description	Command	Keys
17	Determine the lifetime of UDP session in 'connection is confirmed' state after which it is considered to be outdated (optional).	**esr(config)# ip firewall sessions udp-assured-timeout <TIME>**	<TIME> – lifetime of UDP session in 'connection is confirmed' state, takes values in seconds [1..8553600]. Default value: 180 seconds.
18	Determine the lifetime of UDP session in 'connection is not confirmed' state after which it is considered to be outdated.	**esr(config)# ip firewall sessions udp-wait-timeout <TIME>**	<TIME> – lifetime of UDP session in 'connection is not confirmed' state, takes values in seconds [1..8553600]. Default value: 30 seconds.
19	Create IP addresses lists which will be used during filtration.	**esr(config)# object-group network <obj-group-name>**	<obj-group-name> – up to 31 characters.
20	Specify IP addresses list description (optional).	**esr(config-object-group-network)# description <description>**	<description> – profile description, set by the string of up to 255 characters.
21	Add necessary IPv4/IPv6 addresses to the list.	**esr(config-object-group-network)# ip prefix <ADDR/LEN>**	<ADDR/LEN> – 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].
		esr(config-object-group-network)# ip address-range <FROM-ADDR>-<TO-ADDR>	<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].
		esr(config-object-group-network)# ipv6 prefix <IPV6-ADDR/LEN>	<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].

Step	Description	Command	Keys
		esr(config-object-group-network)# ipv6 address-range <FROM-ADDR>-<TO-ADDR>	<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].
22	Create services lists which will be used during filtration.	**esr(config)# object-group service <obj-group-name>**	<obj-group-name> – service profile name, set by the string of up to 31 characters.
23	Specify services list description (optional).	**esr(config-object-group-service)# description <description>**	<description> – profile description, set by the string of up to 255 characters.
24	Add necessary services (tcp/udp ports) to the list.	**esr(config-object-group-service)# port-range <port>**	<port> – takes values in the range of [1..65535]. You can specify several ports separated by commas ',' or you can specify the range of ports with '-'.
25	Create applications lists which will be used in DPI mechanism.	**esr(config)# object-group application <NAME>**	<NAME> – application profile name, set by the string of up to 31 characters.
26	Specify applications list description (optional).	**esr(config-object-group-application)# description <description>**	<description> – profile description, set by the string of up to 255 characters.
27	Add necessary applications to the lists.	**esr(config-object-group-application)# application < APPLICATION >**	<APPLICATION> – specifies the application covered by the given profile
28	Add interfaces (physical, logical, E1/Multilink and connected), remote-access server (l2tp, openvpn, pptp) or tunnels (gre, ip4ip4, l2tp, lt, pppoe, pptp) into security zones (optional).	**esr(config-if-gi)# security-zone <zone-name>**	<zone-name> – up to 12 characters.

Step	Description	Command	Keys
	Disable Firewall functions on the network interface (physical, logical, E1/Multilink and connected), remote-access server (l2tp, openvpn, pptp) or tunnels (gre, ip4ip4, l2tp, lt, pppoe, pptp) (optional).	**esr(config-if-gi)# ip firewall disable**	
29	Create an interzone interaction rule set. There is always a security zone named 'self' on the router. If the router itself acts as a traffic recipient, that is, the traffic is not transit, then the 'self' zone is specified as a parameter. The order of traffic processing for different zone-pairs is described in the note.	**esr(config)# security zone-pair <src-zone-name1> <dst-zone-name2>**	<src-zone-name> – up to 12 characters. <dst-zone-name> – up to 12 characters.
30	Create an interzone interaction rule set.	**esr(config-zone-pair)# rule <rule-number>**	<rule-number> – 1..10000.
31	Specify rule description (optional).	**esr(config-zone-rule)# description <description>**	<description> – up to 255 characters..
32	Specify the given rule force.	**esr(config-zone-rule)# action <action> [log]**	<action> – permit/deny/reject/netflow-sample/sflow-sample log – activation key for logging of sessions established according to the given rule.
33	Set name or number of IP for which the rule should work (optional).	**esr(config-zone-rule)# match [not] protocol <protocol-type>**	<protocol-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.
		esr(config-zone-rule)# match [not] protocol-id <protocol-id>	<protocol-id> – IP identification number, takes values of [0x00-0xFF].

Step	Description	Command	Keys
34	Specify the profile of transmitter IP addresses for which the rule should work (optional).	**esr(config-zone-rule)# match [not] source-address <OBJ-GROUP-NETWORK-NAME>**	<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 work for any sender/recipient IP address.
35	Set the profile of destination IP addresses for which the rule should work (optional).	**esr(config-zone-rule)# match [not] destination-address <OBJ-GROUP-NETWORK-NAME>**	
36	Set source MAC address for which the rule should work (optional).	**esr(config-zone-rule)# match [not] source-mac <mac-addr>**	<mac-addr> – defined as XX:XX:XX:XX:XX:XX where each part takes the values of [00..FF].
37	Set sender MAC address for which the rule should work (optional).	**esr(config-zone-rule)# match [not] destination-mac <mac-addr>**	
38	Set TCP/UDP ports profile for which the rule should work (if the protocol is specified).	**esr(config-zone-rule)# match [not] source-port <PORT-SET-NAME>**	<PORT-SET-NAME> – set by the string of up to 31 characters. When specifying the 'any' value, the rule will work for any sender/recipient TCP/UDP port.
39	Set the destination TCP/UDP ports profile for which the rule should work (if the protocol is specified).	**esr(config-zone-rule)# match [not] destination-port <PORT-SET-NAME>**	
40	Specify the type and code of ICMP messages for which the rule should work (if ICMP is selected as protocol) (optional).	**esr(config-zone-rule)# match [not] icmp <ICMP_TYPE> <ICMP_CODE>**	<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.
41	Set the limitation under which the rule will only work for traffic modified by the IP address and destination ports translation service.	**esr(config-zone-rule)# match [not] destination-nat**	
42	Set the maximum packet rate (optional, available only for zone-pair any self and zone-pair <zone-name> any).	**esr(config-zone-pair-rule)# rate-limit pps <rate-pps>**	<rate-pps> – maximum amount of packets that can be transmitted. Takes values in the range of [1..10000].
43	Set the filtration only for fragmented IP packets (optional, available only for zone-pair any self and zone-pair <zone-name> any).	**esr(config-zone-pair-rule)# match [not] fragment**	

Step	Description	Command	Keys
44	Set the filtration only for IP packets including ip-option (optional, available only for zone-pair any self and zone-pair <zone-name> any).	**esr(config-zone-pair-rule)# match [not] ip-option**	
45	Create an interzone interaction rule.	**esr(config-zone-rule)# enable**	
46	Enable filtering and session tracking mode while packets are transmitted between one Bridge group participants (optional, available only for ESR-1000/1200/1500/1511/1700 /3100).	**esr(config-bridge)# ports firewall enable**	

---

[1] When using the not key, the rule will work for values which are not included in a specified profile.

The order of traffic processing by firewall rules

The order of processing transit traffic by firewall rules

1. If traffic is transmitted from one interface to another within the same zone (src-zone-name), then it is checked by the zone-pair src-zone-name src-zone-name rules. If the traffic does not fall under any of the rules of the current zone-pair, go to the next step.
2. If traffic is transmitted from one interface to another in different zones, then it is checked by the zone-pair src-zone-name dst-zone-name rules. If the traffic does not fall under any of the rules of the current zone-pair, go to the next step.
3. Traffic is checked by the zone-pair src-zone-name any rules. If the traffic does not fall under any of the rules of the current zone-pair, go to the next step.
4. Traffic is checked by the zone-pair any any rules. If the traffic does not fall under any of the rules of the current zone-pair, it is dropped.

The order of processing traffic terminated on the router

1. Traffic is checked by the zone-pair any self rules. If the traffic does not fall under any of the rules of the current zone-pair, go to the next step.
2. Traffic is checked by the zone-pair src-zone-name self rules. If the traffic does not fall under any of the rules of the current zone-pair, go to the next step.

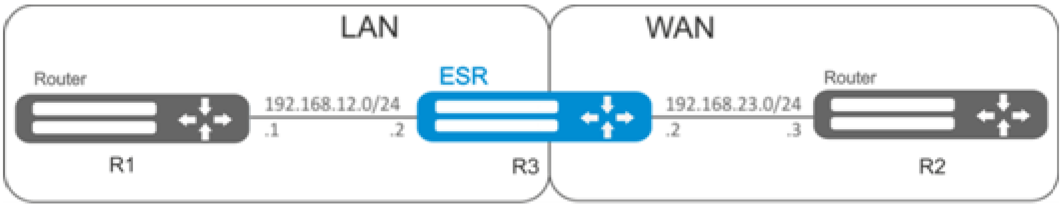
Each 'match' command may contain the 'not' key. When using the key, packets that do not meet the given requirement will fall under the rule.

You can obtain more detail information about firewall configuration in 'CLI command reference guide'.

13.4.2 Firewall configuration example

Objective:

Enable message passage via ICMP between R1, R2 and ESR router.



Solution:

Create a security zone for each ESR network:

```
esr# configure
esr(config)# security zone LAN
esr(config-zone)# exit
esr(config)# security zone WAN
esr(config-zone)# exit
```

Configure network interfaces and identify their inherence to security zones:

```
esr(config)# interface gi1/0/2
esr(config-if-gi)# ip address 192.168.12.2/24
esr(config-if-gi)# security-zone LAN
esr(config-if-gi)# exit
esr(config)# interface gi1/0/3
esr(config-if-gi)# ip address 192.168.23.2/24
esr(config-if-gi)# security-zone WAN
esr(config-if-gi)# exit
```

For definition of rules for security zones, create 'LAN' address profile that includes addresses which are allowed to access WAN network and 'WAN' network address profile.

```
esr(config)# object-group network WAN
esr(config-object-group-network)# ip address-range 192.168.23.2
esr(config-object-group-network)# exit
esr(config)# object-group network LAN
esr(config-object-group-network)# ip address-range 192.168.12.2
esr(config-object-group-network)# exit
esr(config)# object-group network LAN_GATEWAY
esr(config-object-group-network)# ip address-range 192.168.12.1
esr(config-object-group-network)# exit
esr(config)# object-group network WAN_GATEWAY
esr(config-object-group-network)# ip address-range 192.168.23.3
esr(config-object-group-network)# exit
```

To transfer traffic from 'LAN' zone into 'WAN' zone, create a pair of zones and add a rule allowing ICMP traffic transfer from R1 to R2. Rules are applied with the *enable* command:

```
esr(config)# security zone-pair LAN WAN
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol icmp
esr(config-zone-pair-rule)# match destination-address WAN_GATEWAY
esr(config-zone-pair-rule)# match source-address LAN_GATEWAY
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair-pair)# exit
```

To transfer traffic from 'WAN' zone into 'LAN' zone, create a pair of zones and add a rule allowing ICMP traffic transfer from R2 to R1. Rules are applied with the *enable* command:

```
esr(config)# security zone-pair WAN LAN
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol icmp
esr(config-zone-pair-rule)# match destination-address LAN_GATEWAY
esr(config-zone-pair-rule)# match source-address WAN_GATEWAY
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
```

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. Create a pair of zones for traffic coming from 'WAN' zone into 'self' zone. In order the router could response to the ICMP requests from 'WAN' zone, add a rule allowing ICMP traffic transfer from R2 to ESR router:

```
esr(config)# security zone-pair WAN self
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol icmp
esr(config-zone-pair-rule)# match destination-address WAN
esr(config-zone-pair-rule)# match source-address WAN_GATEWAY
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
```

Create a pair of zones for traffic coming from 'LAN' zone into 'self' zone. In order the router could response to the ICMP requests from 'LAN' zone, add a rule allowing ICMP traffic transfer from R1 to ESR:

```
esr(config)# security zone-pair LAN self
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol icmp
esr(config-zone-pair-rule)# match destination-address LAN
esr(config-zone-pair-rule)# match source-address LAN_GATEWAY
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
esr(config)# exit
```

To view port membership in zones, use the following command:

```
esr# show security zone
```

To view zone pairs and their configuration, use the following commands:

```
esr# show security zone-pair
esr# show security zone-pair configuration
```

To view active sessions, use the following commands:

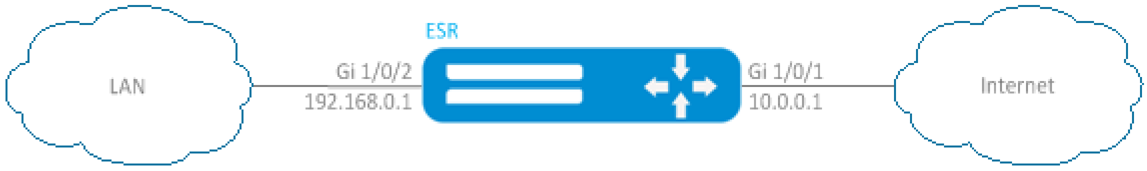
```
esr# show ip firewall sessions
```

13.4.3 Configuration example of application filtering (DPI)

The use of application filtering mechanism reduces by several times the router performance because of the need to check each packet. The performance decreases with an increase in amount of the selected for filtration applications.

Objective:

Block access to such resources as youtube, bittorrent and facebook.



Solution:

Create a security zone for each ESR network:

```
esr# configure
esr(config)# security zone LAN
esr(config-zone)# exit
esr(config)# security zone WAN
esr(config-zone)# exit
```

Configure network interfaces and identify their inherence to security zones:

```
esr(config)# interface gi1/0/1
esr(config-if-gi)# ip address 10.0.0.1/24
esr(config-if-gi)# security-zone WAN
esr(config-if-gi)# exit
esr(config)# interface gi1/0/2
esr(config-if-te)# ip address 192.168.0.1/24
esr(config-if-te)# security-zone LAN
esr(config-if-te)# exit
```

Switch the ESR firewall mode to stateless:

```
esr(config)# ip firewall mode stateless
```

To configure security zones rules, create profile of the applications that should be blocked.

```
esr(config)# object-group application APP
esr(config-object-group-application)# application youtube
esr(config-object-group-application)# application bittorrent
esr(config-object-group-application)# application facebook
esr(config-object-group-application)# exit
```

To set the rules of traffic passing from 'WAN' zone to 'LAN' zone, create a couple of zones and add a rule prohibiting the application traffic flow and a rule allowing all other traffic to pass. Rules are applied with the *enable* command:

```
esr(config)# security zone-pair WAN LAN
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# action deny
esr(config-zone-pair-rule)# match application APP
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# rule 2
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair-pair)# exit
```

To set the rules for passing traffic from the 'LAN' zone to the 'WAN' zone, create a pair of zones and add a rule rule prohibiting the application traffic flow and a rule allowing all other traffic to pass. Rules are applied with the *enable* command:

```
esr(config)# security zone-pair LAN WAN
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# action deny
esr(config-zone-pair-rule)# match application APP
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# rule 2
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair-pair)# exit
```

To view port membership in zones, use the following command:

```
esr# show security zone
```

To view zone pairs and their configuration, use the following commands:

```
esr# show security zone-pair
esr# show security zone-pair configuration
```

To view active sessions, use the following commands:

```
esr# show ip firewall sessions
```

13.5 Access list (ACL) configuration

Access Control List or ACL is a list that contains rules defining traffic transmission through the interface.

13.5.1 Configuration algorithm

Step	Description	Command	Keys
1	Create access control list and switch to its configuration mode.	**esr(config)# ip access-list extended <NAME>**	<NAME> – access control list name, set by the string of up to 31 characters.
2	Specify the description of a configurable access control list (optional).	**esr(config-acl)# description <DESCRIPTION>**	<DESCRIPTION> – access control list description, set by the string of up to 255 characters.
3	Create a rule and switch to its configuration mode. The rules are proceeded by the router in number ascending order.	**esr(config-acl)# rule <ORDER>**	<ORDER> – rule number, takes values of [1..4094].
4	Specify the action that should be applied for the traffic meeting the given requirements.	**esr(config-acl-rule)# action <ACT>**	<ACT> – allocated action: • permit – traffic transfer is permitted; • deny – traffic transfer is denied.
5	Set name of protocol for which the rule should work (optional).	**esr(config-acl-rule)# match protocol <TYPE>**	<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.
		esr(config-acl-rule)# match protocol-id <ID>	<ID> – IP identification number, takes values of [0x00-0xFF].

Step	Description	Command	Keys
6	Set sender IP addresses for which the rule should work (optional).	**esr(config-acl-rule)# match source-address { <ADDR> <MASK> | any }**	<ADDR> – sender 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 work for any sender/ recipient IP address.
7	Set destination IP addresses for which the rule should work (optional).	**esr(config-acl-rule)# match destination-address { <ADDR> <MASK> | any }**	
8	Set sender MAC addresses for which the rule should work (optional).	**esr(config-acl-rule)# match source-mac <ADDR><WILDCARD>**	<ADDR> – sender 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.
9	Set destination MAC addresses for which the rule should work (optional).	**esr(config-acl-rule)# match destination-mac <ADDR><WILDCARD>**	
10	Set the number of sender TCP/ UDP ports for which the rule should work (if the protocol is specified).	**esr(config-acl-rule)# match source-port { <PORT> | any }**	<PORT> – number of sender TCP/ UDP port, takes values of [1..65535]. When specifying the 'any' value, the rule will work for any sender TCP/ UDP port.
11	Set the destination TCP/UDP ports number for which the rule should work (if the protocol is specified).	**esr(config-acl-rule)# match destination-port { <PORT> | any }**	
12	Set priority 802.1p value for which the rule should work (optional).	**esr(config-acl-rule)# match c os <COS>**	<COS> – priority 802.1p value, takes values of [0..7].
13	Set DSCP code value for which the rule should work (optional). Can not be used with IP Precedence.	**esr(config-acl-rule)# match dscp <DSCP>**	<DSCP> – DSCP code value, takes values in the range of [0..63].
14	Set IP Precedence code for which the rule should work (optional). Can not be used with DSCP.	**esr(config-acl-rule)# match ip-precedence <IPP>**	<IPP> – IP Precedence code value, takes values in the range of [0..7].

Step	Description	Command	Keys
15	Set VLAN ID for which the rule should work (optional).	**esr(config-acl-rule)# match vlan <VID>**	<VID> – VLAN ID, takes values of [1..4094].
16	Activate a rule.	**esr(config-acl-rule)# enable**	
17	Specify access control list for the configured interface to filtrate incoming traffic.	**esr(config-if-gi)# service-acl input <NAME>**	<NAME> – access control list name, set by the string of up to 31 characters.

Also the access lists can be used to organize QoS policy.

13.5.2 Access list configuration example

Objective:

Allow traffic transmission from 192.168.20.0/24 subnet only.

Solution:

Configure access control list for filtering by a subnet:

```
esr# configure
esr(config)# ip access-list extended white
esr(config-acl)# rule 1
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match source-address 192.168.20.0 255.255.255.0
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# exit
```

Apply access list to Gi1/0/19 interface for inbound traffic:

```
esr(config)# interface gigabitethernet 1/0/19
esr(config-if-gi)# service-acl input white
```

To view the detailed information on access control list, use the following command:

```
esr# show ip access-list white
```

13.6 IPS/IDS configuration

The function is activated only under the license.

IPS/IDS (*Intrusion Prevention System/Intrusion Detection System*) – a network and computer security software system that detects intrusions or security breaches and automatically protecting from them.

The system is based on signature traffic analysis. Signatures for IPS/IDS systems are commonly called rules. ESR devices allow you to download current rules from open sources on the Internet or from a corporate server. Using the CLI, you can also create your own specific rules.

By default, ESR devices have a basic set of rules from EmergingThreats designed for testing and verifying system health.

13.6.1 Base configuration algorithm

Step	Description	Command	Keys
1	Create IPS/IDS security policy.	**esr(config)# security ips policy <NAME>**	<NAME> – security policy name, set by the string of up to 32 characters
2	Specify policy description (optional).	**esr(config-ips-policy)# description <DESCRIPTION>**	<DESCRIPTION> – description, set by the string of up to 255 characters.
3	Specify the IP address profile that IPS/IDS will protect.	**esr(config-ips-policy)# protect network-group <OBJ-GROUP-NETWORK_NAME>**	<OBJ-GROUP-NETWORK-NAME> – protected IP addresses profile name, set by the string of up to 32 characters.
4	Specify the profile of IP addresses that are external for IPS/IDS (optional).	**esr(config-ips-policy)# external network-group <OBJ-GROUP-NETWORK_NAME>**	<OBJ-GROUP-NETWORK-NAME> – external IP addresses profile name, set by the string of up to 32 characters.
5	Switch to the IPS/IDS configuration mode.	**esr(config)# security ips**	
6	Assign IPS/IDS security policy.	**esr(config-ips)# policy <NAME>**	<NAME> – security policy name, set by the string of up to 32 characters
7	Use all ESR resources for IPS/IDS (optional).	**esr(config-ips)# perfomance max**	By default, half of the available processor cores are allocated for IPS/IDS.

Step	Description	Command	Keys
8	Set remote server parameters for sending IPS/IDS service statistics in EVE format (elasticsearch) (optional).	**esr(config-ips)# logging remote-server { <ADDR> | <IPV6-ADDR> } [<TRANSPORT>] [<PORT>] [source-address { <SRC-ADDR> | <IPV6-SRC-ADDR> }]**	<ADDR> – sender 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]; <TRANSPORT> – data transfer protocol, by default is UDP, takes the following values: • TCP – data transfer via TCP; • UDP – data transfer via UDP. <PORT> – number of sender TCP/UDP port, takes values of [1..65535], by default is 514; <SRC-ADDR> – IPv4 address of the router that will be used as the source IP address in the sent syslog packets. By default – the IPv4 address of the interface from which the packets are sent; <IPV6-SRC-ADDR> – IPv6 address of the router that will be used as the source IP address in the sent syslog packets. By default – the IPv6 address of the interface from which the packets are sent.
9	Set the interval for sending IPS/IDS service statistics in EVE (elasticsearch) format (optional).	**esr(config-ips)# logging update-interval <INTERVAL>**	<INTERVAL> – IPS/IDS service statistics sending interval, set in minutes.
10	Activate IPS/IDS on the interface.	**esr(config-if-gi)# service-ips { inline | monitor }**	inline – this mode is set when ESR with IPS/IDS service is put into network break. monitor – this mode is set when ESR with IPS/IDS monitors mirrored traffic.

13.6.2 Configuration algorithm for IPS/IDS rules autoupdate from external sources

Step	Description	Command	Keys
1	Switch to the autoupdate configuration mode	**esr(config-ips)# auto-upgrade**	

Step	Description	Command	Keys
2	Specify a name and enter the configuration mode of the user update server.	**esr(config-ips-auto-upgrade)# user-server <WORD>**	<WORD> – server name, set by the string of up to 32 characters.
3	Specify the description of the user update server (optional).	**esr(config-ips-upgrade-user-server)# description <DESCRIPTION>**	<DESCRIPTION> – description, set by the string of up to 255 characters.
4	Specify URL.	**esr(config-ips-upgrade-user-server)# url <URL>**	<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.
5	Set the frequency for update checking (optional).	**esr(config-ips-upgrade-user-server)# upgrade interval <HOURS>**	<HOURS> – update interval in hours, from 1 to 240. Default value: 24 hours

For IPS/IDS rules loaded from external sources, a separate area of volatile memory is allocated on ESR routers.
The size of this area depends on the ESR model:

- **ESR-1X – 25 MB;**
- **ESR-2X – 50 MB.**

For all other models - 100 MB.
If you configure too many rule sources or load rules that exceed the specified limits, the router will generate error messages: %STORAGE_IPS_MGR-I-ERR: There is no free space in rules directory error.

13.6.3 Recommended open rule update source

https://sslbl.abuse.ch/	SSL Blacklist contains lists of 'bad' SSL certificates, i.e. certificates in respect of which the fact of their use by malware and botnets has been established. The lists contain SHA1 fingerprints of public keys from SSL certificates.
https://feodotracker.abuse.ch/	Feodo Tracker – list of management servers for the Feodo Trojan. Feodo (also known as Cridex or Bugat) is used by cybercriminals to steal sensitive information in the field of electronic banking (credit card information, logins/passwords) from users' computers. Currently, there are four versions of the Trojan (versions A, B, C and D), mainly distinguished by the infrastructure of control servers.

https://rules.emergingthreats.net/open/suricata/rules/botcc.rules	These rules describe well-known botnets and control servers. Sources: Shadowserver.org, Zeus Tracker, Palevo Tracker, Feodo Tracker, Ransomware Tracker.
https://rules.emergingthreats.net/open/suricata/rules/ciarmy.rules	These rules describe malicious hosts by the classification of the www.cinsarmy.com project.
https://rules.emergingthreats.net/open/suricata/rules/compromised.rules	These rules describe well-known compromised and malicious hosts. Sources: Daniel Gerzo's BruteForceBlocker, The OpenBL, Emerging Threats Sandnet, SidReporter Projects.
https://rules.emergingthreats.net/open/suricata/rules/drop.rules	These rules describe spammer hosts/networks by the classification of the www.spamhaus.org project.
https://rules.emergingthreats.net/open/suricata/rules/dshield.rules	These rules describe malicious hosts by the classification of the www.dshield.org project.
https://rules.emergingthreats.net/open/suricata/rules/emerging-activex.rules	These rules contain signatures for using ActiveX content.
https://rules.emergingthreats.net/open/suricata/rules/emerging-attack_response.rules	Rules that detect host behavior after successful attacks.
https://rules.emergingthreats.net/open/suricata/rules/emerging-chat.rules	These rules describe signs of accessing popular chat rooms.
https://rules.emergingthreats.net/open/suricata/rules/emerging-current_events.rules	Temporary rules awaiting possible inclusion in permanent rule lists.
https://rules.emergingthreats.net/open/suricata/rules/emerging-dns.rules	These rules contain signatures of vulnerabilities in the DNS protocol, signs of the use of DNS by malware, and incorrect use of the DNS protocol.

https://rules.emergingthreats.net/open/suricata/rules/emerging-dos.rules	These rules contain DOS attack signatures.
https://rules.emergingthreats.net/open/suricata/rules/emerging-exploit.rules	These rules contain exploit signatures.
https://rules.emergingthreats.net/open/suricata/rules/emerging-ftp.rules	These rules contain signatures of vulnerabilities in the FTP protocol, signs of incorrect use of the FTP protocol.
https://rules.emergingthreats.net/open/suricata/rules/emerging-games.rules	These rules describe the signs of reference to popular game sites: World of Warcraft, Starcraft, etc.
https://rules.emergingthreats.net/open/suricata/rules/emerging-icmp.rules	These rules contain signatures of incorrect use of the ICMP protocol.
https://rules.emergingthreats.net/open/suricata/rules/emerging-icmp_info.rules	These rules contain signatures of ICMP information messages.
https://rules.emergingthreats.net/open/suricata/rules/emerging-imap.rules	These rules contain signatures of vulnerabilities in the IMAP protocol, signs of incorrect use of the IMAP protocol.
https://rules.emergingthreats.net/open/suricata/rules/emerging-inappropriate.rules	These rules describe signs of accessing unwanted resources.
https://rules.emergingthreats.net/open/suricata/rules/emerging-info.rules	These rules contain different vulnerabilities signatures.
https://rules.emergingthreats.net/open/suricata/rules/emerging-malware.rules	These rules contain signatures of malware that uses the HTTP protocol in their work.

https://rules.emergingthreats.net/open/suricata/rules/emerging-misc.rules	These rules contain different vulnerabilities signatures.
https://rules.emergingthreats.net/open/suricata/rules/emerging-mobile_malware.rules	These rules contain malware signatures for mobile platforms.
https://rules.emergingthreats.net/open/suricata/rules/emerging-netbios.rules	These rules contain signatures of vulnerabilities in the NetBIOS protocol, signs of incorrect use of the NetBIOS protocol.
https://rules.emergingthreats.net/open/suricata/rules/emerging-p2p.rules	These rules describe signs of access to P2P networks (Bittorrent, Gnutella, Limewire).
https://rules.emergingthreats.net/open/suricata/rules/emerging-policy.rules	These rules describe unwanted network activity (access to MySpace, Ebay).
https://rules.emergingthreats.net/open/suricata/rules/emerging-poprules	These rules contain signatures of vulnerabilities in the POP3 protocol, signs of incorrect use of the POP3 protocol.
https://rules.emergingthreats.net/open/suricata/rules/emerging-rpc.rules	These rules contain signatures of vulnerabilities in the RPC protocol, signs of incorrect use of the RPC protocol.
https://rules.emergingthreats.net/open/suricata/rules/emerging-scada.rules	These rules contain vulnerability signatures for SCADA systems.
https://rules.emergingthreats.net/open/suricata/rules/emerging-scan.rules	These rules describe signs of activity associated with network scanning (Nessus, Nikto, portscanning).
https://rules.emergingthreats.net/open/suricata/rules/emerging-shellcode.rules	These rules describe signs of activity associated with attempts to gain shell access as a result of exploits.

https://rules.emergingthreats.net/open/suricata/rules/emerging-smtp.rules	These rules contain signatures of vulnerabilities in the SMTP protocol, signs of incorrect use of the SMTP protocol.
https://rules.emergingthreats.net/open/suricata/rules/emerging-sql.rules	These rules contain vulnerability signatures for SQL DBMS.
https://rules.emergingthreats.net/open/suricata/rules/emerging-telnet.rules	These rules contain signatures of vulnerabilities in the Telnet protocol, signs of incorrect use of the Telnet protocol.
https://rules.emergingthreats.net/open/suricata/rules/emerging-tftp.rules	These rules contain signatures of vulnerabilities in the TFTP protocol, signs of incorrect use of the TFTP protocol.
https://rules.emergingthreats.net/open/suricata/rules/emerging-trojan.rules	These rules contain signs of network activity of Trojans.
https://rules.emergingthreats.net/open/suricata/rules/emerging-user_agents.rules	These rules contain signs of suspicious and potentially dangerous HTTP clients (identified by the values in the User-Agent HTTP header).
https://rules.emergingthreats.net/open/suricata/rules/emerging-l.rules	These rules contain vulnerability signatures for VoIP protocol.
https://rules.emergingthreats.net/open/suricata/rules/emerging-web_client.rules	These rules contain vulnerability signatures for web clients.
https://rules.emergingthreats.net/open/suricata/rules/emerging-web_server.rules	These rules contain vulnerability signatures for web servers.
https://rules.emergingthreats.net/open/suricata/rules/emerging-web_specific_apps.rules	These rules contain vulnerability exploitation signatures for WEB applications.

https://rules.emergingthreats.net/open/suricata/rules/emerging-worm.rules	These rules describe signs of network worm activity.

13.6.4 IPS/IDS configuration example with rules autoupdate

Objective:

Organize LAN protection with autoupdate rules from open sources.

192.168.1.0/24 – LAN

Solution:

Create a profile of protected LAN addresses:

```
esr(config)# object-group network LAN
esr(config-object-group-network)# ip prefix 192.168.1.0/24
esr(config-object-group-network)# exit
```

Configure the DNS client on the ESR to allow the names of the IPS/IDS rule update sources:

```
esr(config)# domain lookup enable
esr(config)# domain name-server 8.8.8.8
```

Create IPS/IDS security policy:

```
esr(config)# security ips policy OFFICE
esr(config-ips-policy)# description "My Policy"
esr(config-ips-policy)# protect network-group LAN
```

Allow IPS/IDS operation on the bridge 1 LAN interface:

```
esr(config)# bridge 1
esr(config-bridge)# service-ips inline
```

Configure IPS/IDS parameters:

```
esr(config)# security ips
esr(config-ips)# logging remote-server 192.168.10.1
esr(config-ips)# logging update-interval 15
esr(config-ips)# policy OFFICE
esr(config-ips)# enable
```

The device will be used only as a security gateway, for this allocate the IPS/IDS service all available resources:

```
esr(config-ips)# perfomance max
```

Configure autoupdate rules from EmergingThreats.net, etnetera.cz and Abuse.ch sites

```
esr(config-ips)# auto-upgrade
esr(config-auto-upgrade)# user-server ET-Open
esr(config-ips-upgrade-user-server)# description "emerging threats open rules"
esr(config-ips-upgrade-user-server)# url https://rules.emergingthreats.net/open/suricata-4.0/
emerging-all.rules
esr(config-ips-upgrade-user-server)# exit
esr(config-auto-upgrade)# user-server Aggressive
esr(config-ips-upgrade-user-server)# description "Etnetera aggressive IP blacklist"
esr(config-ips-upgrade-user-server)# url https://security.etnetera.cz/feeds/
etn_aggressive.rules
esr(config-ips-upgrade-user-server)# upgrade interval 4
esr(config-ips-upgrade-user-server)# exit
esr(config-auto-upgrade)# user-server SSL-BlackList
esr(config-ips-upgrade-user-server)# description "Abuse.ch SSL Blacklist"
esr(config-ips-upgrade-user-server)# url https://sslbl.abuse.ch/blacklist/sslblacklist.rules
esr(config-ips-upgrade-user-server)# upgrade interval 4
esr(config-ips-upgrade-user-server)# exit
esr(config-auto-upgrade)# user-server C2-Botnet
esr(config-ips-upgrade-user-server)# description "Abuse.ch Botnet C2 IP Blacklist"
esr(config-ips-upgrade-user-server)# url https://sslbl.abuse.ch/blacklist/sslipblacklist.rules
esr(config-ips-upgrade-user-server)# upgrade interval 4
esr(config-ips-upgrade-user-server)# exit
```

13.6.5 Basic user rules configuration algorithm

Step	Description	Command	Keys
1	Specify a name and enter the configuration mode of the set of user rules.	**esr(config)# security ips-category user-defined <WORD>**	<WORD> – user rule set name, set by the string of up to 32 characters.
2	Define a description of a set of user rules (optional).	**esr(config-ips-category)# description <DESCRIPTION>**	<DESCRIPTION> – description, set by the string of up to 255 characters.
3	Create a rule and switch to its configuration mode.	**esr(config-ips-category)# rule <ORDER>**	<ORDER> – rule number, takes values of [1..512].
4	Specify rule description (optional).	**esr(config-ips-category-rule)# description <DESCRIPTION>**	<DESCRIPTION> – description, set by the string of up to 255 characters.

Step	Description	Command	Keys
5	Specify the given rule force.	**esr(config-ips-category-rule)# action { alert | reject | pass | drop }**	• 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 recipient, 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.
6	Set name of IP protocol for which the rule should work.	**esr(config-ips-category-rule)# protocol <PROTOCOL>**	<PROTOCOL> – take values: any/ip/icmp/http/tcp/udp When specifying the 'any' value, the rule will work for any protocols
7	Set sender IP addresses for which the rule should work.	**esr(config-ips-category-rule)# source-address {ip <ADDR> | ip-prefix <ADDR/LEN> | object-group <OBJ_GR_NAME> | policy-object-group { protect | external } | any }**	<ADDR> – sender IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]; <ADDR/LEN> – sender 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. • protect – sets sender addresses, protect addresses defined in IPS/IDS policy; • external – sets external addresses defined in IPS/IDS policy as sender addresses. When specifying the 'any' value, the rule will be triggered for any source IP address.

Step	Description	Command	Keys
8	Set the profile of sender TCP/UDP ports for which the rule should work. For protocol icmp value, source-port can only be any.	**esr(config-ips-category-rule)# source-port {any | <PORT> | object-group <OBJ-GR-NAME> }**	<PORT> – number of sender 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 work for any sender TCP/UDP port.
9	Set destination IP addresses for which the rule should trigger.	**esr(config-ips-category-rule)# destination-address {ip <ADDR> | ip-prefix <ADDR/LEN> | object-group <OBJ_GR_NAME> | policy-object-group { protect | external } | any }**	<<ADDR> – recipient IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]; <ADDR/LEN> – recipient 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 recipient IP address, set by the string of up to 31 characters. • protect – sets recipient addresses, protect addresses defined in IPS/IDS policy; • external – sets external addresses defined in IPS/IDS policy as recipient addresses. When specifying the 'any' value, the rule will work for any sender IP address.
10	Set the profile of destination TCP/UDP ports for which the rule should work. For protocol icmp value, destination-port can only be any.	**esr(config-ips-category-rule)# destination-port {any | <PORT> | object-group <OBJ-GR-NAME> }**	<PORT> – number of destination TCP/UDP port, takes values of [1..65535]. <OBJ_GR_NAME> – recipient 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.

Step	Description	Command	Keys
11	Set traffic direction for which the rule should trigger.	**esr(config-ips-category-rule)# direction { one-way | round-trip }**	• one-way – traffic is transmitted in one direction. • round-trip – traffic is transmitted in both directions.
12	Define the message that IPS/IDS will record to the log when this rule will trigger.	**esr(config-ips-category-rule)# meta log-message <MESSAGE>**	<MESSAGE> – text message, specified by a string of up to 129 characters.

Step	Description	Command	Keys
13	Define the traffic classification which will record to the log when this rule will trigger (optional).	**esr(config-ips-category-rule)# 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 }**	• 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.

Step	Description	Command	Keys
			• 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.
14	Set DSCP code value for which the rule should work (optional).	**esr(config-ips-category-rule)# ip dscp <DSCP>**	<DSCP> – DSCP code value, takes values in the range of [0..63].
15	Set the packet lifetime (TTL) value for which the rule will trigger (optional).	**esr(config-ips-category-rule)# ip ttl <TTL>**	<TTL> – TTL value, takes values in the range of [1..255].
16	Set number of IP for which the rule should work (optional). Applicable only for protocol any value.	**esr(config-ips-category-rule)# ip protocol-id <ID>**	<ID> – IP identification number, takes values of [1..255].
17	Set ICMP CODE value for which the rule should trigger (optional). Applicable only for protocol icmp value.	**esr(config-ips-category-rule)# ip icmp code <CODE>**	<CODE> – ICMP CODE value, takes a value in the range [0..255].

Step	Description	Command	Keys
		esr(config-ips-category-rule)# ip icmp code comparison-operator { greater-than | less-than }	Comparison operator for ip icmp code value: • greater-than – greater than.. • less-than – less than..
18	Set ICMP ID value for which the rule should trigger (optional). Applicable only for protocol icmp value.	**esr(config-ips-category-rule)# ip icmp id <ID>**	<ID> – ICMP ID value, takes a value in the range [0..65535].
19	Set ICMP Sequence-ID value for which the rule should trigger (optional). Applicable only for protocol icmp value.	**esr(config-ips-category-rule)# ip icmp sequence-id <SEQ-ID>**	<SEQ-ID> – ICMP Sequence-ID value, takes a value in the range [0..4294967295].
20	Set ICMP TYPE value for which the rule should trigger (optional). Applicable only for protocol icmp value.	**esr(config-ips-category-rule)# ip icmp type <TYPE>**	<TYPE> – ICMP TYPE value, takes a value in the range [0..255].
		esr(config-ips-category-rule)# ip icmp type comparison-operator { greater-than | less-than }	Comparison operator for ip icmp type value: • greater-than – greater than.. • less-than – less than..
21	Set TCP Acknowledgement-Number value for which the rule should trigger (optional). Applicable only for protocol tcp value.	**esr(config-ips-category-rule)# ip tcp acknowledgment-number <ACK-NUM>**	<ACK-NUM> – TCP Acknowledgement-Number value, takes a value in the range [0..4294967295].
22	Set TCP Sequence-ID value for which the rule should trigger (optional). Applicable only for protocol tcp value.	**esr(config-ips-category-rule)# ip tcp sequence-id <SEQ-ID>**	<SEQ-ID> – TCP Sequence-ID value, takes a value in the range [0..4294967295].
23	Set TCP Window-Size value for which the rule should trigger (optional). Applicable only for protocol tcp value.	**esr(config-ips-category-rule)# ip tcp window-size <SIZE>**	<SIZE> – TCP Window-Size value, takes a value in the range [0..65535].

Step	Description	Command	Keys
24	Set HTTP protocol keywords for which the rule will trigger (optional). Applicable only for protocol http value.	**esr(config-ips-category-rule)# ip http { accept | accept-enc | accept-lang | client-body | connection | content-type | cookie | file-data | header | header-names | host | method | protocol | referer | request-line | response-line | server-body | start | start-code | start-msg | uri | user-agent }**	See the Suricata 4.X documentation for the meaning of the keywords. https://suricata.readthedocs.io/en/suricata-4.1.4/rules/http-keywords.html
25	Set HTTP protocol URI LEN keyword value for which the rule will trigger0 (optional). Applicable only for protocol http value.	**esr(config-ips-category-rule)# ip http urilen <LEN>**	<LEN> – takes values in the range of [0.. 65535].
		esr(config-ips-category-rule)# ip http urilen comparison-operator { greater-than | less-than }	Comparison operator for ip http urilen value: • greater-than – greater than.. • less-than – less than..
26	Set the value of the content of packages (Payload content) for which the rule will trigger (optional).	**esr(config-ips-category-rule)# payload content <CONTENT>**	<CONTENT> – text message specified by a string of up to 1024 characters.
27	Do not distinguish between uppercase and lowercase letters in the description of package contents (optional). Only applicable in conjunction with the payload content command.	**esr(config-ips-category-rule)# payload no-case**	
28	Set how many bytes from the beginning of the contents of the packet will be checked (optional). Only applicable in conjunction with the payload content command.	**esr(config-ips-category-rule)# payload depth <DEPTH>**	<DEPTH> – the number of bytes from the beginning of the packet contents, takes a value in the range [1 .. 65535]. By default, the entire contents of the package are checked.
29	Set the number of offset bytes from the beginning of the contents of the packet to check (optional). Only applicable in conjunction with the payload content command.	**esr(config-ips-category-rule)# payload offset <OFFSET>**	<OFFSET> – the number of offset bytes from the beginning of the packet contents, takes a value in the range [1 .. 65535]. By default, it is checked from the beginning of the content.
30	Set the size of the contents of packets for which the rule will trigger (optional).	**esr(config-ips-category-rule)# payload data-size <SIZE>**	<SIZE> – packet content size, takes values in the range of [0.. 65535].

Step	Description	Command	Keys
		esr(config-ips-category-rule)# payload data-size comparison-operator { greater-than | less-than }	Comparison operator for payload data-size value: • greater-than – greater than.. • less-than – less than.
31	Specify the threshold number of packets at which the rule will trigger (optional).	**esr(config-ips-category-rule)# threshold count <COUNT>**	<COUNT> – number of packets, takes values in the range of [1.. 65535].
32	Specify the time interval for which the threshold number of packets is considered (Mandatory if threshold count is enabled).	**esr(config-ips-category-rule)# threshold second <SECOND>**	<SECOND> – time interval in seconds, takes values in the range of [1.. 65535].
33	Specify at the sender or recipient address thresholds will be considered. (Mandatory if threshold count is enabled).	**esr(config-ips-category-rule)# threshold track { by-src | by-dst }**	• by-src – read threshold value for packets with the same IP sender. • by-dst – read threshold value for packets with the same IP recipient.
34	Specify threshold handling method.	**esr(config-ips-category-rule)# threshold type {threshold | limit | both }**	• 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 <SECOND> time interval.
35	Activate rule.	**esr(config-ips-category-rule)# enable**	

13.6.6 Basic user rules configuration example

Objective:

Write a rule to protect a server with IP 192.168.1.10 from a DOS attack by large ICMP packets.

Solution:

Create a set of user rules:

```
esr(config)# security ips-category user-defined USER
```

Create a rule to protect against attack:

```
esr(config-ips-category)# rule 10
esr(config-ips-category-rule)# description "Big ICMP DoS"
```

We will drop packets:

```
esr(config-ips-category-rule)# action drop
```

Configure attack message:

```
esr(config-ips-category-rule)# meta log-message "Big ICMP DoS"
esr(config-ips-category-rule)# meta classification-type successful-dos
```

Specify protocol type for the rule:

```
esr(config-ips-category-rule)# protocol icmp
```

Since icmp protocol was specified, 'any' should be specified as the port of the sender and recipient:

```
esr(config-ips-category-rule)# source-port any
esr(config-ips-category-rule)# destination-port any
```

We will indicate our server as the recipient address:

```
esr(config-ips-category-rule)# destination-address ip 192.168.1.10
```

Attacker can send packets from any address:

```
esr(config-ips-category-rule)# source-address any
```

Set traffic direction:

```
esr(config-ips-category-rule)# direction one-way
```

The rule will trigger on packets larger than 1024 bytes:

```
esr(config-ips-category-rule)# payload data-size 1024
esr(config-ips-category-rule)# payload data-size comparison-operator greate r-than
```

The rule will trigger if the load on the server exceeds 3 Mbps, while an attack message will be generated not more than once a minute:

```
3 Mbps = 3145728 bps
1KB packet = 8192 bits
3145728/8192 = 384 packet per second
384 * 60 = 23040 packets per minute
```

```
esr(config-ips-category-rule)# threshold count 23040
esr(config-ips-category-rule)# threshold second 60
esr(config-ips-category-rule)# threshold track by-dst
esr(config-ips-category-rule)# threshold type both
```

13.6.7 Extended user rules configuration algorithm

Step	Description	Command	Keys
1	Specify a name and enter the configuration mode of the set of user rules.	**esr(config)# security ips-category user-defined <WORD>**	<WORD> – user rule set name, set by the string of up to 32 characters.
2	Define a description of a set of user rules (optional).	**esr(config-ips-category)# description <DESCRIPTION>**	<DESCRIPTION> – description, set by the string of up to 255 characters.
3	Create extended rule and switch to its configuration mode.	**esr(config-ips-category)# rule-advanced <SID>**	<SID> – rule number, takes values of [1..4294967295].
4	Specify rule description (optional).	**esr(config-ips-category-rule-advanced)# description <DESCRIPTION>**	<DESCRIPTION> – description, set by the string of up to 255 characters.

Step	Description	Command	Keys
5	Specify the given rule force.	**esr(config-ips-category-rule-advanced)# rule-text <LINE>**	<CONTENT> – text message in SNORT 2.X/ Suricata 4.X format, specified by a string of up to 1024 characters. **When writing rules,only double quotes (symbol ") must be used in the text of the rule, and the rule itself must be enclosed in single quotes (symbol ').**
6	Activate a rule.	**esr(config-ips-category-rule-advanced)# enable**	

13.6.8 Extended user rules configuration example

Objective:

Write a rule detecting attack like Slowloris.

Solution:

Create a set of user rules:

```
esr(config)# security ips-category user-defined ADV
```

Create an extended rule:

```
esr(config-ips-category)# rule-advanced 1
esr(config-ips-category-rule-advanced)# description "Slow Loris rule 1"
esr(config-ips-category-rule-advanced)# rule-text 'alert tcp any any -> any 80 (msg:"Possible Slowloris Attack Detected"; flow:to_server,established; content:"X-a|3a|"; distance:0; pcre:"/\d\d\d\d/"; distance:0; content:"|0d 0a|"; sid:10000001;)'
```

Create another extended rule that works on a similar algorithm to determine which rule will be more effective:

```
esr(config-ips-category)# rule-advanced 2
esr(config-ips-category-rule-advanced)# description "Slow Loris rule 2"
esr(config-ips-category-rule-advanced)# rule-text 'alert tcp $EXTERNAL_NET any -> $HOME_NET $HTTP_PORTS (msg:"SlowLoris.py DoS attempt"; flow:established,to_server,no_stream; content:"X-a:"; dsize:<15; detection_filter:track by_dst, count 3, seconds 30; classtype:denial-of-service; sid: 10000002; rev:1; )'
```

13.7 Eltex Distribution Manager interaction configuration

EDM (Eltex Distribution Manager) is a service for distributing licensed content to devices via commercial subscription.

Using Kaspersky Lab's security infrastructure, including the Kaspersky Security Network cloud-based 'collective intelligence' with Kaspersky SafeStream II support, the ESR service router is able to detect malware in all types of traffic (web, email, P2P, instant messaging services, etc.). As a result, users are protected from the most dangerous cyber threats, including zero-day threats, encryption programs, infected sites and other types.

IPS on ESR devices can use the following sets of rules provided by Kaspersky SafeStream II:

- IP address Reputation Data – set of IP addresses with contextual information that reports suspicious and malicious hosts;
- URLs of malicious links – set of URLs corresponding to dangerous links and websites;
- URLs of phishing links – set of URLs recognized by Kaspersky Lab as phishing. Masked and unmasked entries are available;
- URLs of botnet command servers – set of URLs of botnet command servers and associated malicious objects;
- URLs of encryptors – set of encryptor URLs;
- Hashes of malicious objects – set of file hashes that covers the most dangerous and common, as well as the newest malicious programs;
- Hashes of malicious objects for mobile devices – a set of file hashes to detect malicious objects infecting mobile devices;
- URLs of botnet command servers for mobile devices – a set of URLs with contextual information to identify botnet command servers using mobile devices;
- URLs of websites used to host malicious programs that infect Internet of Things (IoT) devices.

EDM Server software is provided to operate under a group license, allowing the new ESR service router to be automatically enabled under an existing license. Thus, the user of the system can manage the allocation of licenses to ESR devices within his organization. EDM Server software can be installed on multiple hosts to provide scalability and fault tolerance.

13.7.1 Basic configuration algorithm

Step	Description	Command	Keys
1	Go to the content provider configuration.	**esr (config)# content-provider**	
2	Specify edm server IP address.	**esr (config-content-provider)# host address <A.B.C.D | WORD | X:X:X:X::X>**	<IP-ADDR> – 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]. WORD(1-31) - DNS name of the server.
3	Set the port to connect to the edm server.	**esr (config-content-provider)# host port <PORT>**	<PORT> – number of sender TCP/ UDP port, takes values of [1..65535].
4	Set the type and partition of the external device to create a crypto store.	**esr (config-content-provider)# storage-device <DEVICE>**	<DEVICE> – label and partition name on the external storage in the format of usb://Partion_name:/ mmc://Partion_name:/

Step	Description	Command	Keys
5	Set the time to reboot the device after receiving the certificate.	**esr (config-content-provider)# reboot immediately | [time <HH:MM:SS>]**	Restart the device after receiving the certificate. time <HH:MM:SS> – The time at which ESR will reboot <hours:minutes:seconds>.
6	Enable content provider.	**enable**	
7	Set the interval for accessing the edm server in hours.	**esr (config-content-provider)# upgrade interval <1-240>**	
8	Specify description (optional).	**esr (config-content-provider)# description <LINE>**	<LINE> (1-255) – string describing server.
9	Set device text name that is transmitted to the EDM-Issue server (optional).	**esr (config-content-provider)# system-name <WORD>**	<WORD> – name, set by the string of up to 255 characters.
10	Set text description that is transmitted to the EDM-Issue server (optional).	**esr (config-content-provider)# location <WORD>**	<WORD> – description, set by the string of up to 255 characters.
11	Create IP addresses lists which will be used during filtration.	**esr (config)# object-group network <WORD>esr (config-object-group-network)# ip prefix <ADDR/LEN>**	<WORD> – server name, set by the string of up to 32 characters. <ADDR/LEN> – 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].
12	Enable service-ips on interface.	**esr (config)# interface gigabitethernet 1/0/Xesr (config-if-gi)# service-ips enable**	
13	Create IPS/IDS security policy.	**esr (config)# security ips policy WORD(1-31)**	WORD(1-31)
14	Specify the IP address profile that IPS/IDS will protect.	**esr(config-ips-policy)# protect network-group <OBJ-GROUP-NETWORK_NAME>**	<OBJ-GROUP-NETWORK-NAME> – protected IP addresses profile name, set by the string of up to 32 characters.
15	Enter the vendor configuration section.	**esr (config-ips-policy)# vendor kaspersky**	

Step	Description	Command	Keys
16	Connect the required category.	**esr (config-ips-vendor)# category WORD(1-64)**	Phishing URL Data Feed – Phishing URL data streams Malicious URL Data Feed – Malicious URL data streams Botnet C&C URL Data Feed – Botnet C&C URL data streams Malicious Hash Data Feed – Malicious Hashes data streams Mobile Malicious Hash Data Feed – mobile Malicious Hashes data streams IP Reputation Data Feed – IP address data streams Mobile Botnet Data Feed – mobile Botnet data streams Ransomware URL Data Feed – Ransomware URL data stream Botnet C&C URL Exact Data Feed – Botnet C&C URL Exact data stream Phishing URL Exact Data Feed – Phishing URL Exact data stream Malicious URL Exact Data Feed – Malicious URL Exact data stream Iot URL Data Feed – IoT URL data stream
17	Specify rule type.	**esr (config-ips-vendor-category)# rules action <ACTION>**	<ACTION> - drop | reject | alert | pass – actions to be applied to packages. • 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 recipient, 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.

Step	Description	Command	Keys
18	Set the number of downloadable rules.	**esr (config-ips-vendor-category)# rules count <number>**	<number>
19	Enable category.	**enable**	
20	Switch to the IPS/IDS configuration mode.	**esr (config)# security ips**	
21	Assign IPS/IDS security policy.	**esr(config-ips)# policy <NAME>**	<NAME> – security policy name, set by the string of up to 32 characters.
22	Use all ESR resources for IPS/IDS (optional).	**esr(config-ips)# perfomance max**	
23	Set remote server parameters for sending IPS/IDS service statistics in EVE format (elasticsearch) (optional).	**esr(config-ips)# logging remote-server { <ADDR> | <IPV6-ADDR> } [<TRANSPORT>] [<PORT>] [source-address { <SRC-ADDR> | <IPV6-SRC-ADDR> }]**	<ADDR> – sender 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]; <TRANSPORT> – data transfer protocol, by default is UDP, takes the following values: • TCP – data transfer via TCP; • UDP – data transfer via UDP. <PORT> – number of sender TCP/UDP port, takes values of [1..65535], by default is 514; <SRC-ADDR> – IPv4 address of the router that will be used as the source IP address in the sent syslog packets. By default – the IPv4 address of the interface from which the packets are sent; <IPV6-SRC-ADDR> – IPv6 address of the router that will be used as the source IP address in the sent syslog packets. By default – the IPv6 address of the interface from which the packets are sent.
24	Set the interval for sending IPS/IDS service statistics in EVE (elasticsearch) format (optional).	**esr(config-ips)# logging update-interval <INTERVAL>**	

Step	Description	Command	Keys
25	Enable IPS/IDS.	**esr(config- ips)# enable**	

13.7.2 Configuration example

Set the content-provider parameters – this is the address of the Eltex server. There must be network reachability between the content-provider server and the router.

```
content-provider
  host address edm.eltex-co.ru
  host port 8098
  upgrade interval 1
  storage-path mmc://TEST:/
  reboot immediately
  enable
exit
```

After rebooting the device, you can start configuring the IPS service.

Specify the IP address profile that IPS/IDS will protect:

```
object-group network objectgroup0
  ip prefix 192.168.30.0/24
exit
```

Enable IPS on the interface:

```
interface gigabitethernet 1/0/1
  service-ips enable
exit
```

Configure security policy:

```
security ips policy policy0
  protect network-group objectgroup0
  vendor kaspersky
    category MaliciousURLsDF
      rules action alert
      rules count 100
      enable
    exit
    category MobileBotnetCAndCDF
      rules action alert
      rules count 1000
      enable
    exit
    category APTIPDF
      rules action alert
      rules count 1000
      enable
    exit
```

```
    category APTURLsDF
      rules action alert
      rules count 1000
      enable
    exit
    category BotnetCAndCURLsDF
      rules action alert
      rules count 1000
      enable
    exit
    category IPReputationDF
      rules action alert
      rules count 1000
      enable
    exit
    category IoTURLsDF
      rules action alert
      rules count 1000
      enable
    exit
    category MaliciousHashDF
      rules action alert
      rules count 1
      enable
    exit
    category MobileMaliciousHashDF
      rules action alert
      rules count 1
      enable
    exit
    category PSMSTrojanDF
      rules action alert
      rules count 1
      enable
    exit
    category PhishingURLsDF
      rules action alert
      rules count 1000
      enable
    exit
    category RansomwareURLsDF
      rules action alert
      rules count 1000
      enable
    exit
  exit
exit
```

Assign an IPS policy to the service and enable it:

```
security ips
  performance max
  policy policy0
  enable
exit
```

The following commands can be used to view information about downloaded content for IPS/IDS:

show security ips content-provider

```
esr-20# show security ips content-provider
 Server: content-provider
               Last MD5 of received files:        c60bd0f10716d3f48e18f24828337135
               Next update: 30 October 2020 00:37:06
```

With this command you can find out if the content provider has downloaded rules from the EDM server (based on the presence of the md5 checksum) and when the next update is scheduled for the device.

show security ips counters

```
esr-20# show security ips counters
TCP flows processed :      191
Alerts generated :         0
Blocked by ips engine :    7
Accepted by ips engine :   51483
```

It shows the traffic that passed through IPS/IDS and the actions that were applied to the traffic, as well as the number of times IPS/IDS rules were fired.

13.8 Content filtering service configuration

The function is activated only under the license.

The content filtering service is designed to restrict access to HTTP sites based on their content. For each site is determined by its belonging to a particular category. Kaspersky Lab database is used as a database of site categories. ESR sends HTTPS requests to Kaspersky Lab's server at https://ksn-vt.kaspersky-labs.com to determine the category of sites.

The operation of the content filtering service is based on the Intrusion Prevention System (IPS) and is configured as user IPS rules.

13.8.1 Basic configuration algorithm

Step	Description	Command	Keys
1	Define DNS server IP address used for DNS names resolution.	**esr(config)# domain name-server <IP>**	<IP> – IP address of DNS server being used, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
2	Enable DNS name resolution on the device.	**esr(config)# domain lookup enable**	
3	Create IPS/IDS security policy.	**esr(config)# security ips policy <NAME>**	<NAME> – security policy name, set by the string of up to 32 characters.

Step	Description	Command	Keys
4	Specify policy description (optional).	**esr(config-ips-policy)# description <DESCRIPTION>**	<DESCRIPTION> – description, set by the string of up to 255 characters.
5	Create IP addresses lists which will be used during filtration.	**esr (config)# object-group network <WORD>** **esr (config-object-group-network)# ip prefix <ADDR/LEN>**	<WORD> – server name, set by the string of up to 32 characters. <ADDR/LEN> – 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].
6	Specify the IP address profile that IPS/IDS will protect.	**esr(config-ips-policy)# protect network-group <OBJ-GROUP-NETWORK_NAME>**	<OBJ-GROUP-NETWORK-NAME> – protected IP addresses profile name, set by the string of up to 32 characters.
7	Specify the profile of IP addresses that are external for IPS/IDS (optional).	**esr(config-ips-policy)# external network-group <OBJ-GROUP-NETWORK_NAME>**	<OBJ-GROUP-NETWORK-NAME> – external IP addresses profile name, set by the string of up to 32 characters.
8	Create a content filter category profile.	**esr(config)# object-group content-filter <NAME>**	<NAME> – name of the content filtering profile, specified as a string of up to 31 characters.
9	Set the description of the content filter categories profile (optional).	**esr(config-object-group-content-filter)# description <DESCRIPTION>**	<DESCRIPTION> – description, set by the string of up to 255 characters.
10	Set the content filtering category provider.	**esr(config-object-group-content-filter)# vendor <CONTENT-FILTER-VENDOR>**	<CONTENT-FILTER-VENDOR> – name of the content filtering category provider. In the current version of the software, only Kaspersky Lab can act as a content filtering category provider.
11	Set the necessary categories of content filtering.	**esr(config-object-group-cf-kaspersky)# category <CATEGORY>**	<CATEGORY> – category name. A description of the available categories can be found in the CLI command reference.
12	Switch to the IPS/IDS configuration mode.	**esr(config)# security ips**	
13	Assign IPS/IDS security policy.	**esr(config-ips)# policy <NAME>**	<NAME> – security policy name, set by the string of up to 32 characters.

Step	Description	Command	Keys
14	Use all ESR resources for IPS/IDS (optional).	**esr(config-ips)# perfomance max**	By default, half of the available processor cores are allocated for IPS/IDS.
15	Set remote server parameters for sending IPS/IDS service statistics in EVE format (elasticsearch) (optional).	**esr(config-ips)# logging remote-server { <ADDR> | <IPV6-ADDR> } [<TRANSPORT>] [<PORT>] [source-address { <SRC-ADDR> | <IPV6-SRC-ADDR> }]**	<ADDR> – sender 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]; <TRANSPORT> – data transfer protocol, by default is UDP, takes the following values: • TCP – data transfer via TCP; • UDP – data transfer via UDP. <PORT> – number of sender TCP/UDP port, takes values of [1..65535], by default is 514; <SRC-ADDR> – IPv4 address of the router that will be used as the source IP address in the sent syslog packets. By default – the IPv4 address of the interface from which the packets are sent; <IPV6-SRC-ADDR> – IPv6 address of the router that will be used as the source IP address in the sent syslog packets. By default – the IPv6 address of the interface from which the packets are sent.
16	Set the interval for sending IPS/IDS service statistics in EVE (elasticsearch) format (optional).	**esr(config-ips)# logging update-interval <INTERVAL>**	<INTERVAL> – IPS/IDS service statistics sending interval, set in minutes.
17	Enable IPS/IDS.	**esr(config-ips)# enable**	
18	Enable IPS/IDS on the interface.	**esr(config-if-gi)# service-ips enable**	
19	Specify a name and enter the configuration mode of the set of user rules.	**esr(config)# security ips-category user-defined <WORD>**	<WORD> – user rule set name, set by the string of up to 32 characters.

Step	Description	Command	Keys
20	Define a description of a set of user rules (optional).	**esr(config-ips-category)# description <DESCRIPTION>**	<DESCRIPTION> – description, set by the string of up to 255 characters.
21	Create a rule and switch to its configuration mode.	**esr(config-ips-category)# rule <ORDER>**	<ORDER> – rule number, takes values of [1..512].
22	Specify rule description (optional).	**esr(config-ips-category-rule)# description <DESCRIPTION>**	<DESCRIPTION> – description, set by the string of up to 255 characters.
23	Specify the given rule force.	**esr(config-ips-category-rule)# action { alert | reject | pass | drop }**	• 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 recipient, 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.
24	Set the IP protocol to HTTP.	**esr(config-ips-category-rule)# protocol http**	
25	Set sender IP addresses for which the rule should work.	**esr(config-ips-category-rule)# source-address {ip <ADDR> | ip-prefix <ADDR/LEN> | object-group <OBJ_GR_NAME> | policy-object-group { protect | external } | any }**	<ADDR> – sender IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]; <ADDR/LEN> – sender 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. • protect – sets sender addresses, protect addresses defined in IPS/IDS policy; • external – sets external addresses defined in IPS/IDS policy as sender addresses. When specifying the 'any' value, the rule will be triggered for any source IP address.

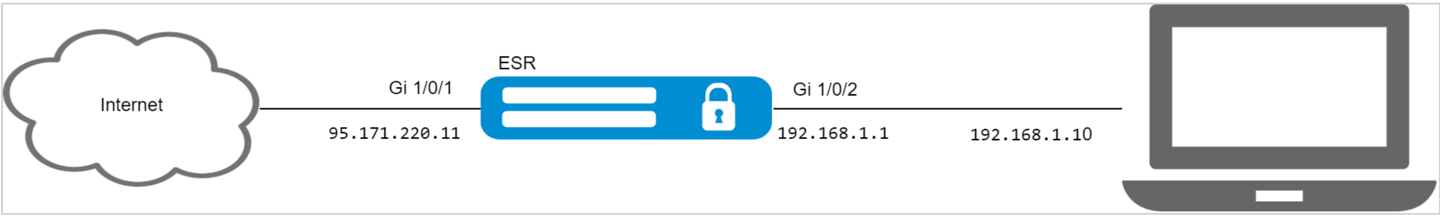
Step	Description	Command	Keys
26	Set the profile of source TCP ports for which the rule should work.	**esr(config-ips-category-rule)# source-port {any | <PORT> | object-group <OBJ-GR-NAME> }**	<PORT> – number of sender TCP/UDP port, takes values of [1..65535]. <OBJ_GR_NAME> – sender TCP/UDP ports profile name, set by the string of up to 31 characters. When specifying the 'any' value, the rule will work for any sender TCP/UDP port.
27	Set destination IP addresses for which the rule should trigger.	**esr(config-ips-category-rule)# destination-address {ip <ADDR> | ip-prefix <ADDR/LEN> | object-group <OBJ_GR_NAME> | policy-object-group { protect | external } | any }**	<<ADDR> – recipient IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]; <ADDR/LEN> – recipient 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 recipient IP address, set by the string of up to 31 characters. • protect – sets recipient addresses, protect addresses defined in IPS/IDS policy; • external – sets external addresses defined in IPS/IDS policy as recipient addresses. When specifying the 'any' value, the rule will work for any sender IP address.
28	Set the profile of destination TCP ports for which the rule should trigger. Normally TCP port 80 is used for the http protocol. In cases where web servers are used on non-standard ports need to write these ports too.	**esr(config-ips-category-rule)# destination-port {any | <PORT> | object-group <OBJ-GR-NAME> }**	<PORT> – number of destination TCP/UDP port, takes values of [1..65535]. <OBJ_GR_NAME> – recipient 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.
29	Set traffic direction for which the rule should trigger.	**esr(config-ips-category-rule)# direction { one-way | round-trip }**	• one-way – traffic is transmitted in one direction. • round-trip – traffic is transmitted in both directions.

Step	Description	Command	Keys
30	Define the message that IPS/IDS will record to the log when this rule will trigger.	**esr(config-ips-category-rule)# meta log-message <MESSAGE>**	<MESSAGE> – text message specified by a string of up to 129 characters.
31	Assign a content filter category profile	**esr(config-ips-category-rule)# ip http content-filter <NAME>**	<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.
32	Activate a rule.	**esr(config-ips-category-rule)# enable**	

13.8.2 Content filtering rules configuration example

Objective:

Deny access to http sites related to the categories of adult-content, casino, online-betting, online-lotteries from the local network 192.168.1.0/24



Solution:

Interfaces and firewall rules must be configured on the device beforehand.

Create a profile of protected LAN addresses:

```
esr(config)# object-group network LAN
esr(config-object-group-network)# ip prefix 192.168.1.0/24
esr(config-object-group-network)# exit
```

Configure the DNS client on the ESR to allow the names of the IPS/IDS rule update sources:

```
esr(config)# domain lookup enable
esr(config)# domain name-server 8.8.8.8
```

Create IPS/IDS security policy:

```
esr(config)# security ips policy OFFICE
esr(config-ips-policy)# description "My Policy"
esr(config-ips-policy)# protect network-group LAN
```

Allow IPS/IDS operation on the bridge gigabitethernet 1/0/2 interface:

```
esr(config)# interface gigabitethernet 1/0/2
esr(config-if-gi)# service-ips inline
```

Configure IPS/IDS parameters:

```
esr(config)# security ips
esr(config-ips)# policy OFFICE
esr(config-ips)# enable
```

The device will be used only as a security gateway, for this allocate the IPS/IDS service all available resources:

```
esr(config-ips)# perfomance max
```

Create a content filtering profile for the selected categories:

```
esr(config)# object-group content-filter Black
esr(config-object-group-content-filter)# vendor kaspersky-lab
esr(config-object-group-cf-kaspersky)# category adult-content
esr(config-object-group-cf-kaspersky)# category casino
esr(config-object-group-cf-kaspersky)# category online-betting
esr(config-object-group-cf-kaspersky)# category online-lotteries
```

Create a set of user rules:

```
esr(config)# security ips-category user-defined USER
```

Create the rule:

```
esr(config-ips-category)# rule 10
esr(config-ips-category-rule)# description "Content-Filter Block"
```

The packets will be dropped:

```
esr(config-ips-category-rule)# action drop
```

Configure attack message:

```
esr(config-ips-category-rule)# meta log-message «Corporate policy violation»
```

Specify protocol type for the rule:

```
esr(config-ips-category-rule)# protocol http
```

For http requests, the operating system uses a random value as the TCP sender port, so you must specify any:

```
esr(config-ips-category-rule)# source-port any
```

As the TCP destination port for the protocol http is usually used port 80, but Internet sites can also work on non-standard ports, so we specify any:

```
esr(config-ips-category-rule)# destination-port any
```

As the recipient's address can be any site on the Internet:

```
esr(config-ips-category-rule)# destination-address any
```

Requests to the sites are sent from our local network:

```
esr(config-ips-category-rule)# source-address policy-object-group protect
```

Set traffic direction:

```
esr(config-ips-category-rule)# direction one-way
```

The rule will trigger for categories of sites listed in the Black profile:

```
esr(config-ips-category-rule)# ip http content-filter Black
```

Activate the rule:

```
esr(config-ips-category-rule)# enable
esr(config-ips-category-rule)# exit
esr(config-ips-category-rule)# threshold type both
```

13.9 Antispam service configuration

> **The function is activated only under the license.**

Mail antispam or spam filter is a program for detecting and filtering unwanted e-mail messages that can come through corporate mail servers and public e-mail services (spam, mail phishing, etc.).

The main task of the Antispam service is to detect such unwanted emails while they are still being delivered to the recipient's mailbox. For this purpose, the ESR service router with configured Antispam service is installed in the network in front of the protected mail server and redirects e-mail between this server and other mail servers on the Internet through itself, actually performing the function of Mail Proxy.

Emails from external email domains will be analyzed in the Antispam service in the following ways:

- authentication of the sending domain via SPF;
- verification of the signature of an email signed with a domain key using DKIM technology;
- identification of e-mails according to DMARC technology;
- checking if there is a correct MX record for the domain from which the email is sent;
- searching for the sender of an email in a list of known broadcast services;
- searching for the sender of a letter in the RBL;
- analysis of SMTP commands correctness during SMTP session pickup;
- analysis of Unicode encodings present in the text of the letter;
- analysis of references in the text of the email to phishing.

Emails that do not pass most of the checks will be discarded and will not reach the protected mail server.

> **When using the Antispam service to protect the mail server, make a number of additional settings not directly related to the ESR router configuration.**
> 1. **Change the MX record for the domain in use so that it does not point to a protected mail server, but to the ESR IP address with the Antispam service configured.**
> 2. **Configure the mail server to use SMTP Proxy, where the Proxy will be ESR with the Antispam service configured.**

13.9.1 Basic configuration algorithm

Step	Description	Command	Keys
1	Configure router network name.	**esr(config)# hostname <NAME>**	<NAME> – up to 64 characters.
2	Assign domain name for the router.	**esr(config)# domain name <NAME>**	<NAME> – up to 255 characters.
3	Assign DNS server IP address used for DNS names resolution.	**esr(config)# domain name-server <IP>**	<IP> – in the format of AAA.BBB.CCC.DDD, where each part takes values from 0 to 255.
4	Enable DNS name resolution.	**esr(config)# domain lookup enable**	

Step	Description	Command	Keys
5	Create Antispam service profile	**esr(config)# security antispam profile <NAME>**	<NAME> - up to 31 characters.
6	Set the description of the Antispam service profile (optional).	**esr(config-antispam-profile)# description <DESCRIPTION>**	<DESCRIPTION> – up to 255 characters.
7	Set the marking type for e-mails that the Antispam service classifies as 'Spam'.	**esr(config-antispam-profile)# mark-type <MARK-TYPE>**	<MARK-TYPE> – the marking type of an email categorized as 'Spam'. Possible values: • header – add the X-Spam header to email headers; • subject – add the [SPAM] tag before the subject of the email.
8	Create the mail domain profile and mailbox addresses (optional).	**esr(config)# object-group email <NAME>**	<NAME> – up to 31 characters.
9	Set the description of the mail domain profile and mailbox addresses (optional).	**esr(config-object-group-email)# description <DESCRIPTION>**	<DESCRIPTION> – up to 255 characters.
10	Enter the mail domain or mailbox address into the profile (optional).	**esr(config-object-group-email)# email <NAME>**	<NAME> – up to 63 characters.
11	Create a rule in the Antispam service profile (optional).	**esr(config-antispam-profile)# rule <ORDER>**	<ORDER> – rule number, may take values from 1..100.
12	Set the description of the Antispam service profile rule (optional).	**esr(config-antispam-profile)# description <DESCRIPTION>**	<DESCRIPTION> – up to 255 characters.
13	Specify the profile of transmitter IP addresses for which the rule should work (optional).	**esr(config-antispam-profile-rule)# sender ip <NAME>**	<NAME> – up to 31 characters.
14	Set the profile of mail domains and mailbox addresses for which the rule should trigger (optional).	**esr(config-antispam-profile-rule)# sender email <NAME>**	<NAME> – up to 31 characters.
15	Specify the action for the rule.	**esr(config-antispam-profile-rule)# action <ACTION>**	<ACTION> – assigned action. Possible values: • reject – further delivery of the letter is prohibited, the sender of the letter is sent a reply about the error.

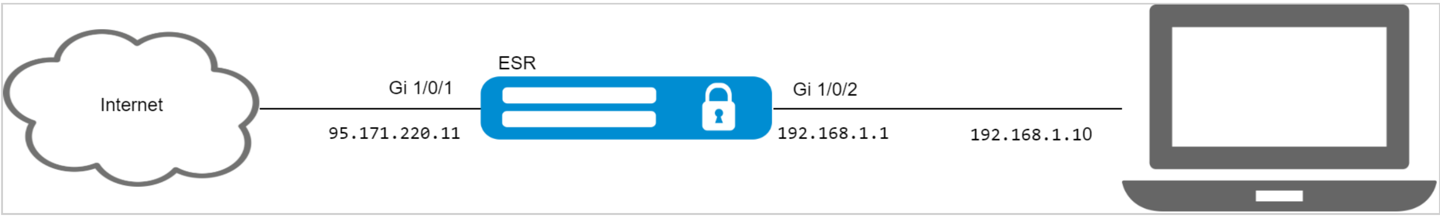
Step	Description	Command	Keys
16	Enable a rule in the Antispam service profile (optional).	**esr(config-antispam-profile-rule)# enable**	
17	Create mail domain.	**esr(config)# mailserver domain <DOMAIN-NAME>**	<DOMAIN-NAME> - up to 31 characters.
18	Set the description of the email domain (optional).	**esr(config-mailserver-domain)# description <DESCRIPTION>**	<DESCRIPTION> – up to 255 characters.
19	Set the name of the email domain to be served.	**esr(config-mailserver-domain)# mail domain <NAME>**	<NAME> - up to 63 characters.
20	Set the IP address of the mail server for which the Antispam service on ESR acts as SMTP Proxy.	**esr(config-mailserver-domain)# mail server ip <ADDR>**	<ADDR> – in the format of AAA.BBB.CCC.DDD, where each part takes values from 0 to 255.
21	Set the Antispam service profile, the settings of which will be applied to the current email domain.	**esr(config-mailserver-domain)# profile antispam <NAME>**	<NAME> - up to 63 characters.
22	Enable mail domain.	**esr(config-mailserver-domain)# enable**	
23	Proceed to the mail server configuration.	**esr(config)# mailserver**	
24	Set the name of the email domain.	**esr(config-mailserver)# domain <NAME>**	<NAME> – up to 63 characters.
25	Specify certificates and keys for the TLS protocol (optional).	**esr(config-mailserver)# tls keyfile <TYPE> <NAME>**	<TYPE> – type of certificate or key file. Possible 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.
26	Enable TLS support on the mail server (optional). If TLS is enabled, the configuration must include a preset CA certificate, server private key and public server certificate.	**esr(config-mailserver)# tls enable**	

Step	Description	Command	Keys
27	Set the maximum size of email headers in KB (optional).	**esr(config-mailserver)# headers max-size <SIZE>**	<SIZE> - the maximum size of email headers in KB, takes values from 50 to 200.
28	Set the maximum email size in KB (optional).	**esr(config-mailserver)# mail max-size <SIZE>**	<SIZE> - the maximum email size in KB, takes values from 5120 to 51200.
29	Enable mandatory HELO or EHLO SMTP command when establishing an SMTP session (optional).	**esr(config-mailserver)# smtp helo-required**	
30	Enable the VRFY SMTP command on the mail server during an SMTP session (optional).	**esr(config-mailserver)# smtp vrfy-enable**	
31	Enable mail server.	**esr(config-mailserver)# enable**	

13.9.2 Configuration example

Objective:

Configure the Antispam service on ESR to work as an SMTP Proxy to analyze e-mail addressed to the mail server located in the enterprise network and serving the eltex-co.ru domain.



Solution:

Ensure that the MX record for the domain eltex-co.ru points to the ESR IP address:

```
esr@eltex:~$ dig +noall +answer eltex-co.ru MX
eltex-co.ru.     3548     IN   MX  10 mail-gate.eltex-co.ru.
esr@eltex:~$ dig +noall +answer mail-gate.eltex-co.ru A
mail-gate.eltex-co.ru. 3453 IN   A    95.171.220.11
```

Configure the network interfaces:

```
esr# config
esr(config)# interface gi1/0/1
esr(config-if-gi)# ip address 95.171.220.11/18
esr(config-if-gi)# ip firewall disable
esr(config-if-gi)# exit
esr(config)# interface gi1/0/2
esr(config-if-te)# ip address 192.168.1.1/24
esr(config-if-te)# ip firewall disable
esr(config-if-te)# exit
```

Configure the network name, domain name and DNS settings. The network name and domain name must form the FQDN (Fully Qualified Domain Name), prescribed in the MX record for the domain eltex-co.ru:

```
esr(config)# hostname mail-gate
esr(config)# domain name eltex-co.ru
esr(config)# domain name-server 1.1.1.1
esr(config)# domain lookup enable
```

Create a profile for the Antispam service that will add an X-Spam header to emails identified as spam:

```
esr(config)# security antispam profile SimpleProfile
esr(config-antispam-profile)# description "Basic Antispam profile without rules"
esr(config-antispam-profile)# mark-type header
esr(config-antispam-profile)# exit
```

Create a mail domain, which will be configured to process emails for the eltex-co.ru domain and retransmit such emails to the local mail server. Add the Antispam service profile created above to the configuration of the mail domain so that the mail in transit will be analyzed for spam:

```
esr(config)# mailserver domain MainDomain
esr(config-mailserver-domain)# mail domain eltex-co.ru
esr(config-mailserver-domain)# description "Mail domain eltex-co.ru"
esr(config-mailserver-domain)# mail server ip 192.168.1.10
esr(config-mailserver-domain)# profile antispam SimpleProfile
esr(config-mailserver-domain)# enable
esr(config-mailserver-domain)# exit
```

Add the domain we created to the mail server configuration and write the settings for TLS:

```
esr(config)# mailserver
esr(config-mailserver)# domain MainDomain
esr(config-mailserver)# tls keyfile ca ca.crt
esr(config-mailserver)# tls keyfile server-crt server.crt
esr(config-mailserver)# tls keyfile server-key server.key
esr(config-mailserver)# tls enable
esr(config-mailserver)# enable
esr(config-mailserver)# exit
```

Applying the current configuration will start the service.

⚠ **The SMTP protocol (TCP port 25) must be enabled on the firewall.**

14 Redundancy management

14.1 VRRP configuration

VRRP (*Virtual Router Redundancy Protocol*) is a network protocol designed for increased availability of routers, acting as a default gateway. This is performed by aggregation of a router group into a single virtual router and assigning a shared IP address, that will be used as a default gateway for computers in the network.

14.1.1 Configuration algorithm

Step	Description	Command	Keys
1	Switch to the interface/tunnel/ network bridge configuration mode for which it is necessary to configure VRRP.	**esr(config)# interface <IF-TYPE><IF-NUM>**	<IF-TYPE> – interface type; <IF-NUM> – F/S/P – F frame (1), S – slot (0), P – port.
		esr(config)# tunnel <TUN-TYPE><TUN-NUM>	<TUN-TYPE> – tunnel type; <TUN-NUM> – tunnel number.
		esr(config)# bridge <BR-NUM>	<BR-NUM> – bridge number.
2	Configure the required parameters on the interface/network bridge, including IP address.		
3	Enable VRRP process on IP interface.	**esr(config-if-gi)# vrrp**	
		esr(config-if-gi)# ipv6 vrrp	

Step	Description	Command	Keys
4	Set virtual IP address of VRRP router.	**esr(config-if-gi)# vrrp ip <ADDR/LEN>**	<ADDR/LEN> – virtual IP 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]. You can specify several IP addresses separated by commas. Up to 4 IP addresses can be assigned to the interface.
		esr(config-if-gi)# ipv6 vrrp ip <IPV6-ADDR>	<IPV6-ADDR> – virtual IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF]. Up to 8 IPv6 addresses separated by commas can be specified.
5	Set the VRRP router identifier.	**esr(config-if-gi)# vrrp id <VRID>**	<VRID> – VRRP router identifier, takes values in the range of [1..255].
		esr(config-if-gi)# ipv6 vrrp id <VRID>	
6	Set the VRRP router priority (optional).	**esr(config-if-gi)# vrrp priority <PR>**	<PR> – VRRP router priority, takes values in the range of [1..254]. Default value: 100.
		esr(config-if-gi)# ipv6 vrrp priority <PR>	
7	Identify the VRRP router's inherence to a 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 (optional).	**esr(config-if-gi)# vrrp group <GRID>**	<GRID> – VRRP router group identifier, takes values in the range of [1..32].
		esr(config-if-gi)# ipv6 vrrp group <GRID>	
8	Set the IP address that will be used as a source IP address for VRRP messages (optional).	**esr(config-if-gi)# vrrp source-ip <IP>**	<ADDR> – sender IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
		esr(config-if-gi)# ipv6 vrrp source-ip <IPV6>	<IPV6> – source IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].
9	Set the interval between sending VRRP messages (optional).	**esr(config-if-gi)# vrrp timers advertise <TIME>**	<TIME> – time in seconds, takes values of [1..40]. Default value: 1 second.

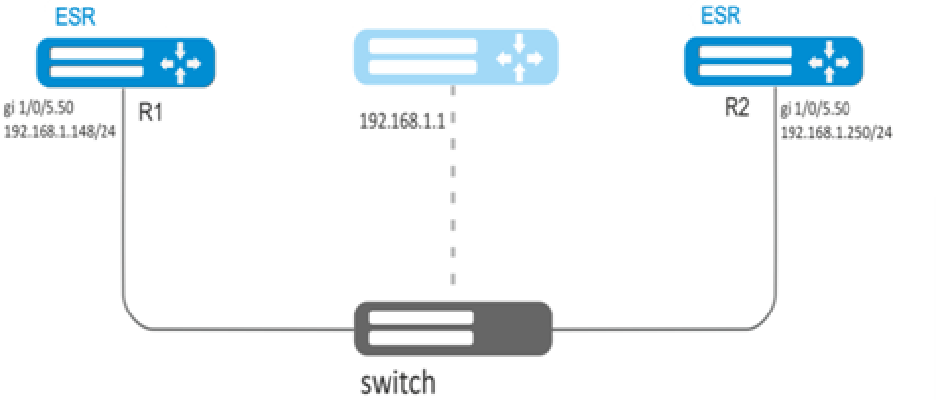
Step	Description	Command	Keys
		esr(config-if-gi)# ipv6 vrrp timers advertise <TIME>	
10	Set the interval after which GratuituousARP messages are sent when switching the router to the Master status (optional).	**esr(config-if-gi)# vrrp timers garp delay <TIME>**	<TIME> – time in seconds, takes values of [1..60]. Default value: 5 seconds.
11	Set the amount of GratuituousARP messages that will be sent when switching the router to the Master status (optional).	**esr(config-if-gi)# vrrp timers garp repeat <COUNT>**	<COUNT> – amount of messages, takes values of [1..60]. Default value: 5.
12	Set the interval after which GratuituousARP messages will be sent periodically while the router is in the Master status (optional).	**esr(config-if-gi)# vrrp timers garp refresh <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: Periodic sending is disabled.
13	Set the amount of GratuituousARP messages that will be sent with the garprefresh period while the router is in the Master status (optional).	**esr(config-if-gi)# vrrp timers garp refresh-repeat <COUNT>**	<COUNT> – amount of messages, takes values of [1..60]. Default value: 1.
14	Specify whether the higher priority Backup router would try to take the Master role from the current lower priority Master router (optional).	**esr(config-if-gi)# vrrp preemption disable**	
		esr(config-if-gi)# ipv6 vrrp preemption disable	
15	Set the time interval after which the higher priority Backup route will try to take the Master role from the current lower priority Master router (optional).	**esr(config-if-gi)# vrrp preemption delay <TIME>**	<TIME> – timeout, takes value in seconds [1..1000]. Default value: 0
		esr(config-if-gi)# ipv6 vrrp preemption delay <TIME>	
16	Set the password for neighbour authentication (optional).	**esr(config-if-gi)# vrrp authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }**	<CLEAR-TEXT> – password, set by the string of 8 to 16 characters; <ENCRYPTED-TEXT> – encrypted password of 8 to 16 bytes (from 16 to 32 characters) in hexadecimal format (0xYYYY ...) or (YYYY ...).

Step	Description	Command	Keys
17	Specify authentication algorithm (optional).	**esr(config-if-gi)# vrrp authentication algorithm <ALGORITHM>**	<ALGORITHM> – authentication algorithm: • **cleartext** – password, transmitted in clear text; • **md5** – password is hashed by md5 algorithm.
18	Specify VRRP version (optional).	**esr(config-if-gi)# vrrp version <VERSION>**	<VERSION> – VRRP version: 2, 3.
19	Set the mode when vrrp IP address remains in the UP status regardless of the status of the interface itself (optional).	**esr(config-if-gi)# vrrp force-up**	
20	Specify the delay between the assignment of MASTER status to ipv6 vrrp and the start of ND messages distribution (optional).	**esr(config-if-gi)# ipv6 vrrp timers nd delay <TIME>**	<TIME> – time in seconds, takes values of [1..60]. Default value: 5.
21	Specify the period of ND protocol information update for ipv6 vrrp in MASTER status (optional).	**esr(config-if-gi)# ipv6 vrrp timers nd refresh <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 5.
22	Specify the amount of ND messages sent in the update period for ipv6 vrrp in MASTER status (optional).	**esr(config-if-gi)# ipv6 vrrp timers nd refresh-repeat <NUM>**	<NUM> – amount, takes values of [1..60]. Default value: 0.
23	Specify the amount of ND packets sendings after setting ipv6 vrrp to the MASTER status (optional).	**esr(config-if-gi)# ipv6 vrrp timers nd repeat <NUM>**	<NUM> – amount, takes values of [1..60]. Default value: 1.

14.1.2 Configuration example 1

Objective:

Establish LAN virtual gateway in VLAN 50 using VRRP. IP address 192.168.1.1 is used as a local virtual gateway.



Solution:

First, do the following:

- create a correspond sub interface;
- configure a zone for the sub-interface;
- specify IP address for the sub-interface.

Main configuration step:

Configure R1 router.

Configure VRRP in the created sub-interface. Specify unique VRRP identifier:

```
R1(config)#interface gi 1/0/5.50
R1(config-subif)# vrrp id 10
```

Specify virtual gateway IP address 192.168.1.1/24:

```
R1(config-subif)# vrrp ip 192.168.1.1
```

Enable VRRP:

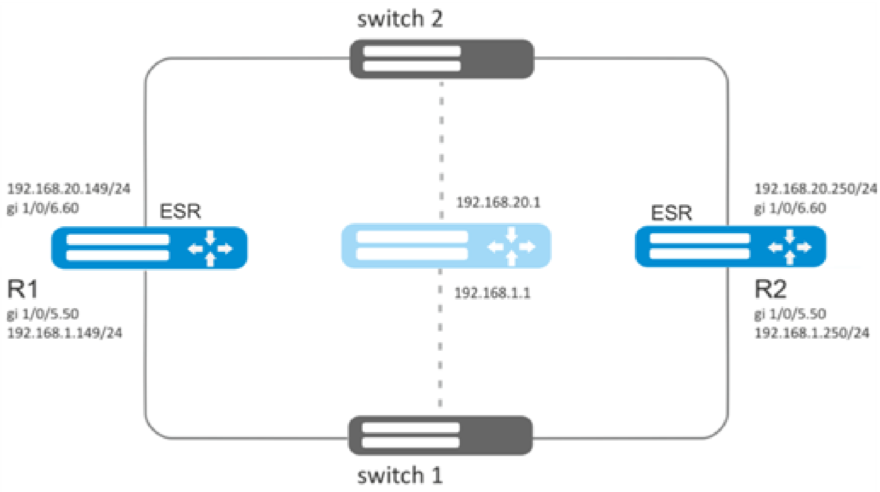
```
R1(config-subif)# vrrp
R1(config-subif)# exit
```

After that it is necessary to make the same settings on R2.

14.1.3 Configuration example 2

Objective:

Establish virtual gateways for 192.168.20.0/24 subnet in VLAN 50 and 192.168.1.0/24 in VLAN 60 using VRRP with Master sync feature. To do this, group VRRP processes. IP addresses 192.168.1.1 and 192.168.20.1 are used as virtual gateways.



Solution:

First, do the following:

- create correspond sub interfaces;
- configure a zone for the sub-interfaces;
- specify IP addresses for the sub-interfaces.

Main configuration step:

Configure R1 router.

Configure VRRP for 192.168.1.0/24 subnet in the created sub-interface.

Specify unique VRRP identifier:

```
R1(config-sub)#interface gi 1/0/5.50
R1(config-subif)# vrrp id 10
```

Specify virtual gateway IP address 192.168.1.1:

```
R1(config-subif)# vrrp ip 192.168.1.1
```

Specify VRRP group identifier:

```
R1(config-subif)# vrrp group 5
```

Enable VRRP:

```
R1(config-subif)# vrrp
R1(config-subif)# exit
```

Configure VRRP for 192.168.20.0/24 subnet in the created sub-interface.

Specify unique VRRP identifier:

```
R1(config-sub)#interface gi 1/0/6.60
R1(config-subif)# vrrp id 20
```

Specify virtual gateway IP address 192.168.1.20:

```
R1(config-subif)# vrrp ip 192.168.20.1
```

Specify VRRP group identifier:

```
R1(config-subif)# vrrp group 5
```

Enable VRRP:

```
R1(config-subif)# vrrp
R1(config-subif)# exit
```

Configure R2 in the same manner.

In addition to tunnel creation, enable VRRP protocol (112) in the firewall.

When using IPsec with VRRP, it is recommended to configure DPD to speed up IPsec tunnel rebuilding.

14.2 Tracking configuration

Tracking is a mechanism, which allows activating entities, depending on VRRP/SLA state.

14.2.1 Configuration algorithm

Step	Description	Command	Keys
1	Configure VRRP according to the section VRRP configuration algorithm or configure SLA.		
2	Add Tracking object to the system and switch to the Tracking object parameters configuration mode.	**esr(config)#tracking <ID>**	<ID> – Tracking object number, takes values of [1..100].

Step	Description	Command	Keys
3	Set a rule for tracking VRRP/SLA processes, based on which Tracking object will switch to active state.	**esr(config-track)# track vrrp id <VRID> state [not] { master | backup | fault } [vrf <VRF>]**	<VRID> – trackable VRRP router identifier, takes values in the range of [1..255]; <VRF> – VRF name, set by the string of up to 31 characters.
		esr(config-track)# track sla test <NUM> [mode <MODE>]	<NUM> – SLA test name, set in range of [1..10000]; <MODE> – SLA test tracking mode, may take the following values: • state – state of the SLA test is monitored; • reachability – state of the communication channel is monitored, which is provided by the SLA test.
4	Enable Tracking object.	**esr(config-tracking)#enable**	
5	Set delay for changing state of the monitored object (optional).	**esr(config-track)# delay { down | up } <TIME>**	<TIME> – delay in seconds, set in range of [1..300].
6	Set tracking operation mode (optional).	**esr(config-track)# mode <MODE>**	<MODE> – condition for the Tracking object to be in the active state, takes the following values: • and – Tracking object is in active state, if all tracked conditions are in active state; • or – Tracking object is in active state, if at least one tracked condition is in active state.
7	Create an entity on the ESR that will change depending on the state of the Tracking object.		

Step	Description	Command	Keys
7.1	Add the ability to manage a static IP route to the specified subnet (optional).	**esr(config)# ip route [vrf <VRF>] <SUBNET> { <NEXTHOP> [resolve] | interface <IF> | tunnel <TUN> | wan load-balance rule <RULE> | blackhole | unreachable | prohibit } [<METRIC>] [track <TRACK-ID>]**	<VRF> – VRF name, set by the string of up to 31 characters. <SUBNET> – destination address, can be specified in the following formats: AAA.BBB.CCC.DDD – host IP address, where each part takes values of [0..255]. AAA.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);

Step	Description	Command	Keys
			• **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.
7.2	Add the ability to manage the logical state of the interface (optional).	**esr(config-if-gi)# shutdown track <ID>**	<ID> – Tracking object number, takes values of [1..100].
7.3	Add the ability to control the priority of the VRRP process (optional).	**esr(config-if-gi)# vrrp priority track <ID> { <PRIO> | increment <INC> | decrement <DEC> }**	<ID> – Tracking object number, takes values of [1..100]; <PRIO> – priority of the VRRP process, which will be set if the Tracking object is in the active state, takes values of [1..254]; <INC> – value by which the priority of the VRRP process will increase if the Tracking object is in the active state, takes values of [1..254]; <DEC> – value by which the priority of the VRRP process will decrease if the Tracking object is in the active state, takes values of [1..254].
7.4	Add the ability to control Next-Hop for packets that match criteria in the specified access list (ACL) (optional).	**esr(config-route-map-rule)# action set ip next-hop verify-availability <NEXTHOP><METRIC> track <ID>**	<NEXTHOP> – gateway IP address in the AAA.BBB.CCC.DDD format, where each part takes values of [0..255]; <METRIC> – route metric, takes values of [0..255]; <ID> – Tracking object number, takes values of [1..100].

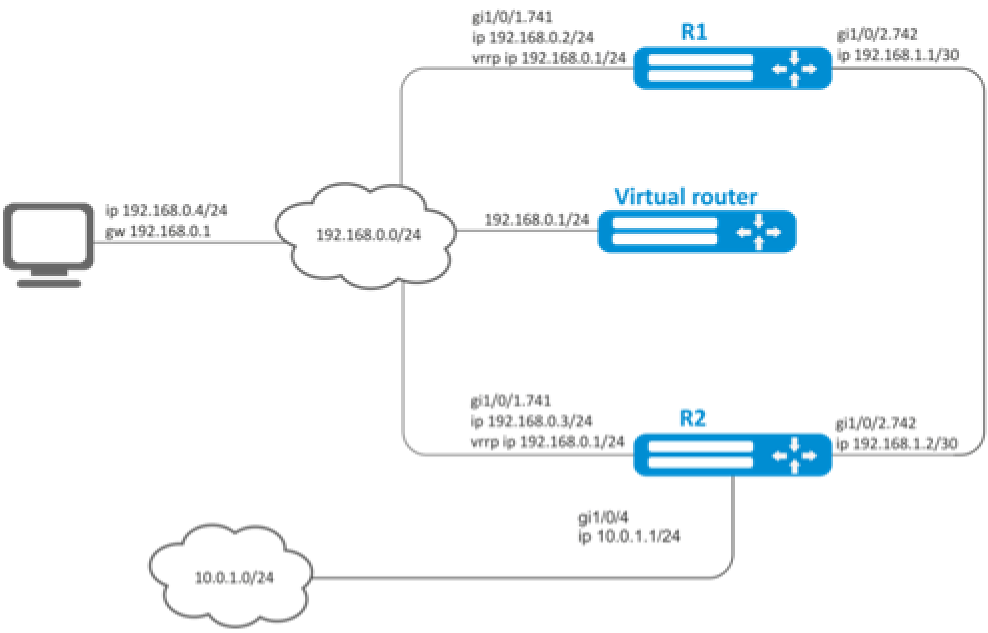
Step	Description	Command	Keys
7.5	Add the ability to control the BGP AS-Path attribute that will be added to the front of the AS-Path list (optional).	**esr(config-route-map-rule)# action set as-path prepend <AS-PATH> track <ID>**	<AS-PATH> – list of autonomous system numbers to be added to the current value in the route. Specified as AS,AS,AS, takes values of [1..4294967295]; <ID> – Tracking object number, takes values of [1..100].
7.6	Add the ability to control the BGP MED attribute in the route for which the rule should fire (optional).	**esr(config-route-map-rule)# action set metric bgp <METRIC> track <ID>**	<METRIC> – BGPMED attribute value, takes values of [0..4294967295]; <ID> – Tracking object number, takes values of [1..100].

14.2.2 Configuration example

Objective:

Virtual gateway 192.168.0.1/24 is organized for 192.168.0.0/24 subnet, using VRRP protocol and routers R1 and R2. There is a link with a singular subnet 192.168.1.0/30 between R1 and R2 routers. Subnet 10.0.1.0/24 is terminated only on R2 router. PC has IP address - 192.168.0.4/24 and default gateway 192.168.1.1.

When router R1 is in vrrp backup state, traffic from PC will be transmitted without any additional settings. When router R1 is in vrrp master state, additional route is necessary for subnet 10.0.1.0/24 through interface 192.168.1.2.



Initial configurations of the routers:

R1 router

```
hostname R1
interface gigabitethernet 1/0/1
  switchport forbidden default-vlan
exit
interface gigabitethernet 1/0/1.741
  ip firewall disable
  ip address 192.168.0.2/24
  vrrp id 10
  vrrp ip 192.168.0.1/24
  vrrp
exit
interface gigabitethernet 1/0/2
  switchport forbidden default-vlan
exit
interface gigabitethernet 1/0/2.742
  ip firewall disable
  ip address 192.168.1.1/30
exit
```

R2 router

```
hostname R2
interface gigabitethernet 1/0/1
  switchport forbidden default-vlan
exit
interface gigabitethernet 1/0/1.741
  ip firewall disable
  ip address 192.168.0.3/24
  vrrp id 10
  vrrp ip 192.168.0.1/24
  vrrp
exit
interface gigabitethernet 1/0/2
  switchport forbidden default-vlan
exit
interface gigabitethernet 1/0/2.742
  ip firewall disable
  ip address 192.168.1.2/30
exit
interface gigabitethernet 1/0/4
  ip firewall disable
  ip address 10.0.1.1/24
exit
```

Solution:

There is no need in any changes in router R2, since subnet 10.0.1.0/24 is terminated on it and as soon as router R2 is vrrp master, packets will be transmitted to corresponding interface. As soon as R1 becomes vrrp master, route for packets must be created with destination IP address from network 10.0.1.0/24.

Create track-object with corresponding condition:

```
R1(config)# track 1
R1(config-track)# track vrrp id 10 state master
R1(config-track)# enable
R1(config-track)# exit
```

Create static route to subnet 10.0.1.0/24 through 192.168.1.2, which will work in case of satisfying of track 1 condition:

```
R1(config)# ip route 10.0.1.0/24 192.168.1.2 track 1
```

14.3 Firewall/NAT failover configuration

Firewall failover is required to reserve firewall sessions.

14.3.1 Configuration algorithm

Step	Description	Command	Key
1	Select the routers communication mode.	**ip firewall failover sync-type <MODE>**	<MODE> – communication mode: • unicast – unicast mode; • multicast – multicast mode.
2	Select the IP address of the network interface from which messages will be sent when the Firewall is running in session reservation mode.	**ip firewall failover source-address <ADDR>**	<ADDR> – IP address of the sender network interface, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
3	Set neighbor's IP address when reserving Firewall sessions in unicast mode.	**ip firewall failover destination-address <ADDR>**	<ADDR> – neighbor IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
	Configuring multicast IP address that will be used to exchange information when the Firewall session backup is in multicast mode.	**ip firewall failover multicast-address <ADDR>**	<ADDR> – multicast IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
4	If Firewall session reservation works in multicast mode, then it is necessary to configure the multicast group ID.	**ip firewall failover multicast-group <GROUP>**	<GROUP> – multicast group, specified in range [1000..9999].

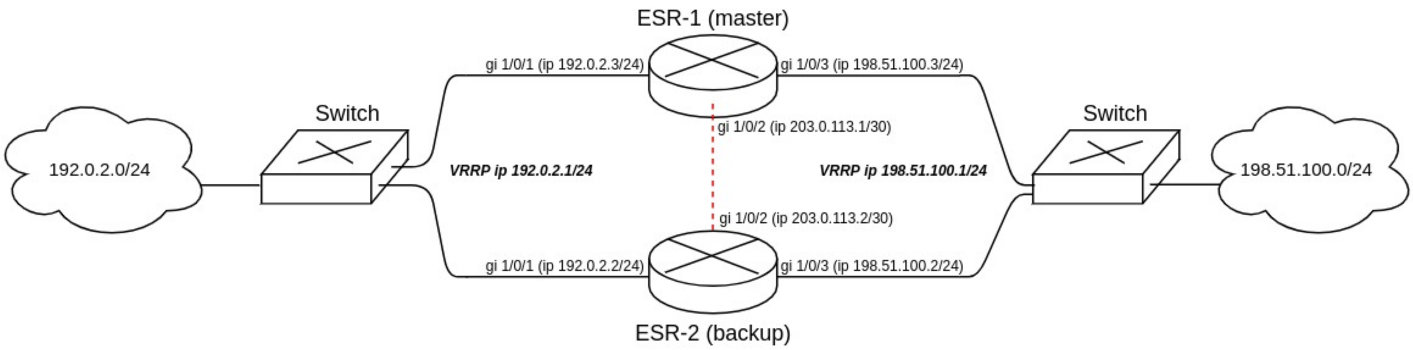
Step	Description	Command	Key
5	Setting the UDP port number of the Firewall session reservation service through which information is exchanged when working in unicast mode (optional).	**ip firewall failover port <PORT>**	<PORT> – port number of Firewall session reservation service, specified in range [1..65535].
6	Binding of a VRRP group, based on which the state (main/backup) of the router is determined when reserving Firewall sessions (optional).	**ip firewall failover vrrp-group <GRID>**	<GRID> – VRRP router group ID, takes values of [1..32].
7	Enabling Firewall session reservation.	**ip firewall failover**	

⚠ **When configuring firewall failover, NAT sessions between devices will also be synchronized.**

14.3.2 Configuration example

Objective:

Configure firewall session reservation for VRRP group in unicast mode. It is necessary to organize redundancy for two subnets using the VRRP protocol, synchronize VRRP processes on routers.



Main configuration steps:

1. Configure VRRP processes on routers. Use vrrp priority 20 for master, and vrrp priority 10 for backup.
2. Configure firewall failover in unicast mode with udp port number 3333 for VRRP group.
3. Configure security zone for VRRP and UDP protocols.

Solution:

Configure ESR-1 router (master).

First, configure IP address on interfaces and define belonging to the security zone.

```
master(config)# interface gigabitethernet 1/0/1
master(config-if-gi)# security-zone trusted
master(config-if-gi)# ip address 192.0.2.3/24
master(config-if-gi)# exit
master(config)# interface gigabitethernet 1/0/2
master(config-if-gi)# security-zone trusted
master(config-if-gi)# ip address 203.0.113.1/30
master(config-if-gi)# exit
master(config)# interface gigabitethernet 1/0/3
master(config-if-gi)# security-zone trusted
master(config-if-gi)# ip address 198.51.100.3/24
master(config-if-gi)# exit
```

Configure VRRP processes on interfaces. The following parameters on the router interfaces should be configured: VRRP ID, VRRP IP address, VRRP priority, VRRP router belonging to a group.

Additionally, vrrp preempt delay should be configured on the master, as a result of which there will be time to establish firewall synchronization before the backup router transfers master role.

After that, enable the vrrp process using the 'vrrp' command.

> **It is possible to select vrrp preempt disable operation mode instead of configuring vrrp preempt delay. As a result, router with higher VRRP priority will not take away the master role from the router with lower VRRP priority after returning to operation.**

To ensure that the VRRP processes states on a router are synchronized (master, backup), as well as to synchronize their sessions using firewall failover, it is necessary to configure them to belong to the same VRRP group.

```
master(config)# interface gigabitethernet 1/0/1
master(config-if-gi)# vrrp id 1
master(config-if-gi)# vrrp ip 192.0.2.1/24
master(config-if-gi)# vrrp priority 20
master(config-if-gi)# vrrp group 1
master(config-if-gi)# vrrp preempt delay 60
master(config-if-gi)# vrrp
master(config-if-gi)# exit

master(config)# interface gigabitethernet 1/0/3
master(config-if-gi)# vrrp id 3
master(config-if-gi)# vrrp ip 198.51.100.1/24
master(config-if-gi)# vrrp priority 20
master(config-if-gi)# vrrp group 1
master(config-if-gi)# vrrp preempt delay 60
master(config-if-gi)# vrrp
master(config-if-gi)# exit
```

Configure firewall failover.

Select the unicast session reservation mode:

```
master(config)# ip firewall failover sync-type unicast
```

Select the IP addresses of the network interface from which messages will be sent when the Firewall is running in session reservation mode:

```
master(config)# ip firewall failover source-address 203.0.113.1
```

Configure the neighbor's IP addresses when reserving Firewall sessions in unicast mode:

```
master(config)# ip firewall failover destination-address 203.0.113.2
```

Configure the UDP port number of the Firewall session reservation service:

```
master(config)# ip firewall failover port 3333
```

Enable Firewall session reservations:

```
master(config)# ip firewall failover
```

To configure security zone rules, create a profile for the firewall failover port:

```
master(config)# object-group service failover
master(config-object-group-service)# port-range 3333
master(config-object-group-service)# exit
```

Additionally, the following protocols must be allowed in the security zone-pair trusted self:

```
master(config)# security zone-pair trusted self
master(config-zone-pair)# rule 66
master(config-zone-pair-rule)# action permit
master(config-zone-pair-rule)# match protocol vrrp
master(config-zone-pair-rule)# enable
master(config-zone-pair-rule)# exit
master(config-zone-pair)# rule 67
master(config-zone-pair-rule)# action permit
master(config-zone-pair-rule)# match protocol udp
master(config-zone-pair-rule)# match destination-port failover
master(config-zone-pair-rule)# enable
master(config-zone-pair-rule)# exit
master(config-zone-pair)# exit
```

View the status of VRRP processes using the following command:

```
master# show vrrp
Virtual router   Virtual IP                         Priority   Preemption   State
--------------   ---------------------------------  --------   ----------   ------
1                192.0.2.1/24                       20         Enabled      Master
3                198.51.100.1/24                    20         Enabled      Master
```

View the status of Firewall session reservations using the following command

```
master# show ip firewall failover
Communication interface:                     gigabitethernet 1/0/2
Status:                                      Running
Bytes sent:                                  2496
Bytes received:                              640
Packets sent:                                271
Packets received:                            40
Send errors:                                 0
Receive errors:                              0
```

View the status of device redundancy systems using the following command:

```
master# show high-availability state
AP Tunnels:
    State:                          Disabled
    Last state change:              --
DHCP server:
    State:                          Disabled
    Last state change:              --
Firewall sessions:
    State:                          successful synchronization
    Last synchronization:           09:38:00 05.08.2021
```

Configure ESR-2 router (backup).
Configure interfaces:

```
backup(config)# interface gigabitethernet 1/0/1
backup(config-if-gi)# security-zone trusted
backup(config-if-gi)# ip address 192.0.2.2/24
backup(config-if-gi)# vrrp id 1
backup(config-if-gi)# vrrp ip 192.0.2.1/24
backup(config-if-gi)# vrrp priority 10
backup(config-if-gi)# vrrp group 1
backup(config-if-gi)# vrrp
backup(config-if-gi)# exit
```

```
backup(config)# interface gigabitethernet 1/0/2
backup(config-if-gi)# security-zone trusted
backup(config-if-gi)# ip address 203.0.113.2/30
backup(config-if-gi)# exi
```

```
backup(config)# interface gigabitethernet 1/0/3
backup(config-if-gi)# security-zone trusted
backup(config-if-gi)# ip address 198.51.100.2/24
backup(config-if-gi)# vrrp id 3
backup(config-if-gi)# vrrp ip 198.51.100.1/24
backup(config-if-gi)# vrrp priority 10
backup(config-if-gi)# vrrp group 1
backup(config-if-gi)# vrrp
backup(config-if-gi)# exit
```

Configure firewall failover:

```
backup(config)# ip firewall failover sync-type unicast
backup(config)# ip firewall failover source-address 203.0.113.2
backup(config)# ip firewall failover destination-address 203.0.113.1
backup(config)# ip firewall failover port 3333
backup(config)# ip firewall failover vrrp-group 1
backup(config)# ip firewall failover
```

Configuration of a security zone is similar to the configuration of security zone for the ESR-1 (master) router.

14.4 DHCP failover configuration

DHCP failover is used to reserve a database of IP addresses that were dynamically issued during the operation of the DHCP server.

14.4.1 Configuration algorithm

Step	Description	Command	Keys
1	To configure DHCP failover, switch to its configuration menu.	**ip dhcp-server failover [vrf <VRF>]**	<VRF> – VRF name, set by the string of up to 31 characters.

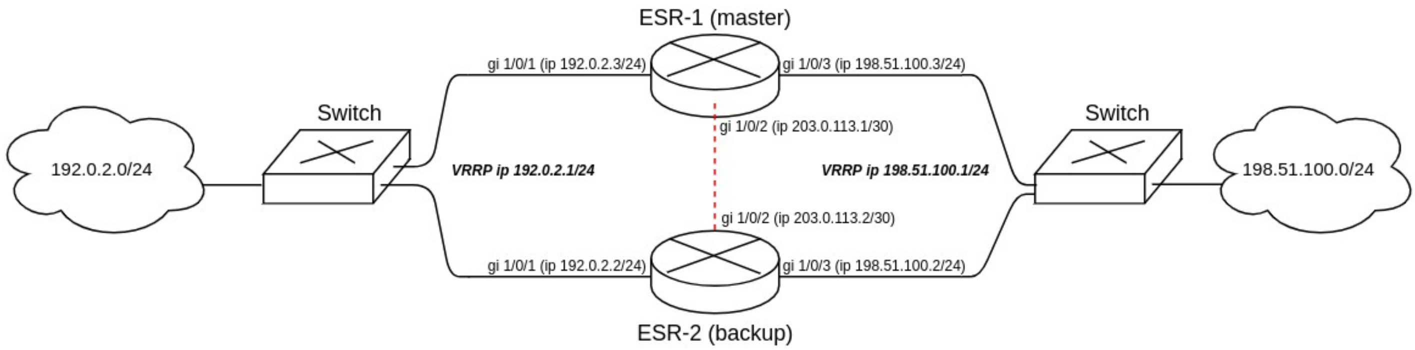
Step	Description	Command	Keys
2	Select DHCP failover operation mode.	**mode { active-active | active-standby }**	active-active – operating mode with two active routers; active-standby – operating mode with one active router and one standby router.
3	Configure the IP address from which DHCP failover will work.	**local-address <ADDR>**	<ADDR> – neighbor IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
4	Configure the remote IP address of the neighbor that DHCP failover will work with.	**remote-address <ADDR>**	<ADDR> – neighbor IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
5	Configure the DHCP failover role when redundancy works in Active-Active mode.	**role <ROLE>**	<ROLE> – DHCP server role when operating in standby mode: • primary – active DHCP server mode; • secondary – standby DHCP server mode.
6	Bind the VRRP group, on the basis of which the state (primary/backup) of the router is determined when reserving sessions in the Active-Standby mode.	**vrrp-group <GRID>**	<GRID> – VRRP router group identifier, takes values in the range of [1..32].
7	Enable DHCP failover redundancy.	**enable**	

The active-standby mode in VRF is not supported.

14.4.2 Configuration example

Objective:

Configure DHCP server reservation in Active-Standby mode. It is necessary to organize redundancy for two subnets using the VRRP protocol, synchronize VRRP processes on routers.



Main configuration steps:

1. Configure VRRP processes on routers. Use vrrp priority 20 for master, and vrrp priority 10 for backup.
2. Configure DHCP failover in Active-Standby mode.
3. Configure security zone for VRRP, UDP and TCP protocols.

Solution:

Configure ESR-1 router (master).

First, configure IP address on interfaces and define belonging to the security zone.

```
master(config)# interface gigabitethernet 1/0/1
master(config-if-gi)# security-zone trusted
master(config-if-gi)# ip address 192.0.2.3/24
master(config-if-gi)# exit
master(config)# interface gigabitethernet 1/0/2
master(config-if-gi)# security-zone trusted
master(config-if-gi)# ip address 203.0.113.1/30
master(config-if-gi)# exit
master(config)# interface gigabitethernet 1/0/3
master(config-if-gi)# security-zone trusted
master(config-if-gi)# ip address 198.51.100.3/24
master(config-if-gi)# exit
```

Configure VRRP processes on interfaces. The following parameters on the router interfaces should be configured: VRRP ID, VRRP IP address, VRRP priority, VRRP router belonging to a group.

After that, enable the vrrp process using the 'vrrp' command.

It is possible to select vrrp preempt disable operation mode instead of configuring vrrp preempt delay. As a result, router with higher VRRP priority will not take away the master role from the router with lower VRRP priority after returning to operation.

To ensure that the VRRP processes states on a router are synchronized (master, backup), it is necessary to configure them to belong to the same VRRP group.

```
master(config)# interface gigabitethernet 1/0/1
master(config-if-gi)# vrrp id 1
master(config-if-gi)# vrrp ip 192.0.2.1/24
master(config-if-gi)# vrrp priority 20
master(config-if-gi)# vrrp group 1
master(config-if-gi)# vrrp
master(config-if-gi)# exit
master(config)# interface gigabitethernet 1/0/3
master(config-if-gi)# vrrp id 3
master(config-if-gi)# vrrp ip 198.51.100.1/24
master(config-if-gi)# vrrp priority 20
master(config-if-gi)# vrrp group 1
master(config-if-gi)# vrrp
master(config-if-gi)# exit
```

Configure DHCP failover. For DHCP failover it is necessary to configure the following parameters: mode, local-address, remote-address, VRRP router belonging to a group.

```
master(config)# ip dhcp-server pool LAN
master(config-dhcp-server)# network 192.0.2.0/24
master(config-dhcp-server)# address-range 192.0.2.10-192.0.2.20
master(config-dhcp-server)# exit
master(config)# ip dhcp-server
master(config)# ip dhcp-server failover
master(config-dhcp-server-failover)# mode active-standby
master(config-dhcp-server-failover)# local-address 203.0.113.1
master(config-dhcp-server-failover)# remote-address 203.0.113.2
master(config-dhcp-server-failover)# vrrp-group 1
master(config-dhcp-server-failover)# enable
master(config-dhcp-server-failover)# exit
```

To start DHCP failover, first configure and enable the DHCP server that will be backed up.

To configure security zone rules, create a profile for the DHCP failover port:

```
master(config)# object-group service dhcp_failover
master(config-object-group-service)# port-range 873
master(config-object-group-service)# exit
```

DHCP failover uses TCP port 873 for synchronization, it must be enabled when configuring firewall.

Additionally, the following protocols must be allowed in the security zone-pair trusted self:

```
master(config)# security zone-pair trusted self
master(config-zone-pair)# rule 66
master(config-zone-pair-rule)# action permit
master(config-zone-pair-rule)# match protocol vrrp
master(config-zone-pair-rule)# enable
master(config-zone-pair-rule)# exit
master(config-zone-pair)# rule 67
master(config-zone-pair-rule)# action permit
master(config-zone-pair-rule)# match protocol tcp
master(config-zone-pair-rule)# match destination-port dhcp_failover
master(config-zone-pair-rule)# enable
master(config-zone-pair-rule)# exit
master(config-zone-pair)# rule 68
master(config-zone-pair-rule)# action permit
master(config-zone-pair-rule)# match protocol udp
master(config-zone-pair-rule)# enable
master(config-zone-pair-rule)# exit
```

View the status of VRRP processes using the following command:

```
master# show vrrp
Virtual router   Virtual IP                          Priority   Preemption   State
--------------   ---------------------------------   --------   ----------   ------
1                192.0.2.1/24                        20         Enabled      Master
3                198.51.100.1/24                     20         Enabled      Master
```

View the status of Firewall session reservations using the following command:

```
master# show ip dhcp server failover
VRF:       --
    State: Successful
```

View the status of device redundancy systems using the following command:

```
master# show high-availability state
AP Tunnels:
    State:                      Disabled
    Last state change:          --
DHCP option 82 table:
    State:                      Disabled
    Last state change:          --
DHCP server:
VRF:                            --
    State:                      Successful synchronization
    State:                      Disabled
    Last synchronization:       --
```

For successful synchronization of the DHCP failover service, identical time must be set on the devices.

ESR-2 router (backup) configuration.

Configure interfaces:

```
backup(config)# interface gigabitethernet 1/0/1
backup(config-if-gi)#   security-zone trusted
backup(config-if-gi)#   ip address 192.0.2.2/24
backup(config-if-gi)#   vrrp id 1
backup(config-if-gi)#   vrrp ip 192.0.2.1/24
backup(config-if-gi)#   vrrp priority 20
backup(config-if-gi)#   vrrp group 1
backup(config-if-gi)#   vrrp
backup(config-if-gi)# exit
backup(config)# interface gigabitethernet 1/0/2
backup(config-if-gi)#   security-zone trusted
backup(config-if-gi)#   ip address 203.0.113.2/30
backup(config-if-gi)# exit
backup(config)# interface gigabitethernet 1/0/3
backup(config-if-gi)#   security-zone trusted
backup(config-if-gi)#   ip address 198.51.100.2/24
backup(config-if-gi)#   vrrp id 3
backup(config-if-gi)#   vrrp ip 198.51.100.1/24
backup(config-if-gi)#   vrrp priority 10
backup(config-if-gi)#   vrrp group 1
backup(config-if-gi)#   vrrp
backup(config-if-gi)# exit
```

Configure DHCP failover:

```
backup(config)# ip dhcp-server pool LAN
backup(config-dhcp-server)#   network 192.0.2.0/24
backup(config-dhcp-server)#   address-range 192.0.2.10-192.0.2.20
backup(config-dhcp-server)# exit
backup(config)# ip dhcp-server
backup(config)# ip dhcp-server failover
backup(config-dhcp-server-failover)#   mode active-standby
backup(config-dhcp-server-failover)#   local-address 203.0.113.2
backup(config-dhcp-server-failover)#   remote-address 203.0.113.1
backup(config-dhcp-server-failover)#   vrrp-group 1
backup(config-dhcp-server-failover)#   enable
backup(config-dhcp-server-failover)# exit
```

Configuration of a security zone is similar to the configuration of security zone for the ESR-1 (master) router.

15 Remote access configuration

15.1 Configuring server for remote access to corporate network via PPTP protocol

PPTP (Point-to-Point Tunneling Protocol) is a point-to-point tunneling protocol that allows a computer to establish secure connection with a server by creating a special tunnel in a common unsecured network. PPTP encapsulates PPP frames into IP packets for transmission via global IP network, e.g. the Internet. PPTP may be used for tunnel establishment between two local area networks. PPTP uses an additional TCP connection for tunnel handling.

15.1.1 Configuration algorithm

Step	Description	Command	Keys
1	Create PPTP server profile.	**esr(config)# remote-access pptp <NAME>**	<NAME> – PPTP server profile name, set by the string of up to 31 characters.
2	Specify the description of the configured server (optional).	**esr(config-pptp-server)# description <DESCRIPTION>**	<DESCRIPTION> – PPTP server description, set by the string of up to 255 characters.

Step	Description	Command	Keys
3	Specify IP address that should be proceeded by PPTP server.	**esr(config-pptp-server)# outside-address { object-group <OBJ-GROUP-NETWORK-NAME> | ip-address <ADDR> | interface { <IF> | <TUN> } }**	<OBJ-GROUP-NETWORK-NAME> – name of the profile having IP address that should listened by PPTP server, set by the string of up to 31 characters; <ADDR> – range starting IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]; <IF> – router interface type and identifier; <TUN> – router tunnel type and number.
4	IP address of a local gateway.	**esr(config-pptp-server)# local-address { object-group <OBJ-GROUP-NETWORK-NAME> | ip-address <ADDR> }**	<OBJ-GROUP-NETWORK-NAME> – name of the IP addresses profile that includes local gateway IP address, set by the string of up to 31 characters; <ADDR> – range starting IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
5	Specify IP addresses list from which dynamic IP addresses are leased to remote users by PPTP.	**esr(config-pptp-server)# remote-address { object-group <OBJ-GROUP-NETWORK-NAME> | address-range <FROM-ADDR>-<TO-ADDR> }**	<OBJ-GROUP-NETWORK-NAME> – name of the IP addresses profile that includes remote users 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].
6	Select PPTP clients authentication mode.	**esr(config-pptp-server)# authentication mode { local | radius }**	• local – user authentication by local base. • radius – user authentication by RADIUS server base. The router must be configured to interact with a RADIUS-server, see section AAA RADIUS configuration algorithm.

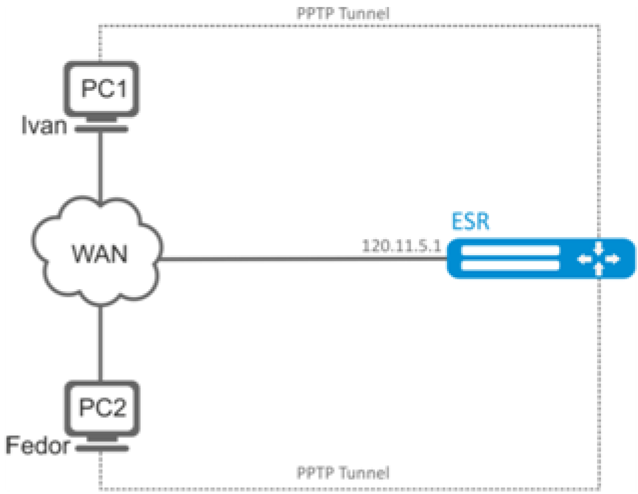
Step	Description	Command	Keys
7	Allow necessary authentication methods for remote users	**esr(config-pptp-server)# authentication method <METHOD>**	<METHOD> – authentication method, possible values: [chap, mschap, mschap-v2, eap, pap]. By default only chap is allowed.
8	Specify user name (when using local user authentication).	**esr(config-pptp-server) username <NAME>**	<NAME> – user name, set by the string of up to 12 characters.
9	Specify password (when using local user authentication).	**esr(config-pptp-user) password ascii-text { <PASSWORD> | encrypted <PASSWORD> }**	<PASSWORD> – user password, set by the string of up to 32 characters.
10	Activate user (when using local user authentication).	**esr(config-pptp-user) enable**	
11	Include the PPTP server in a security zone and configure interaction rules between zones or disable firewall (see section Firewall configuration).	**esr(config-pptp-server)# security-zone <NAME>**	<NAME> – security zone name, set by the string of up to 31 characters.
12	Enable server.	**esr(config-pptp-server)# enable**	
13	Specify outgoing packets DSCP priority (optional).	**esr(config-pptp-server)# dscp <DSCP>**	<DSCP> – outgoing packets dscp priority [0..63].
14	Enable MPPE encryption for PPTP connections (optional).	**esr(config-pptp-server)# encryption mppe**	
15	Specify MTU size (MaximumTransmissionUnit) for the server (optional). MTU above 1500 will be active only when using the 'system jumbo-frames' command.	**esr(config-pptp-server) mtu <MTU>**	<MTU> – MTU value, takes values in the range of [1280..1500]. Default value: 1500.
16	Define the list of DNS servers that will be used by remote users (optional).	**esr(config-pptp-server)# dns-servers object-group <OBJ-GROUP-NETWORK -NAME>**	<OBJ-GROUP-NETWORK-NAME> – name of the IP addresses profile that includes required DNS servers addresses, set by the string of up to 31 characters.
17	Define the list of WINS servers that will be used by remote users (optional).	**esr(config-pptp-server)# wins-servers object-group <OBJ-GROUP-NETWORK-NAME >**	<OBJ-GROUP-NETWORK-NAME> – name of the IP addresses profile that includes required WINS servers addresses, set by the string of up to 31 characters.

15.1.2 Configuration example

Objective:

Configure PPTP server on a router.

- PPTP server address: 120.11.5.1;
- Gateway inside the tunnel for connecting clients: 10.10.10.1;
- IP address pool for lease: 10.10.10.5-10.10.10.25;
- DNS servers: 8.8.8.8, 8.8.8.4;
- Accounts for connection: Fedor, Ivan.



Solution:

Create an address profile that contains an address to be listened by the server:

```
esr# configure
esr(config)# object-group network pptp_outside
esr(config-object-group-network)# ip address-range 120.11.5.1
esr(config-object-group-network)# exit
```

Create address profile that contains local gateway address:

```
esr(config)# object-group network pptp_local
esr(config-object-group-network)# ip address-range 10.10.10.1
esr(config-object-group-network)# exit
```

Create address profile that contains client addresses:

```
esr(config)# object-group network pptp_remote
esr(config-object-group-network)# ip address-range 10.10.10.5-10.10.10.25
esr(config-object-group-network)# exit
```

Create PPTP server and map profiles listed above:

```
esr(config)# remote-access pptp remote-workers
esr(config-pptp)# local-address object-group pptp_local
esr(config-pptp)# remote-address object-group pptp_remote
esr(config-pptp)# outside-address object-group pptp_outside
esr(config-pptp)# dns-servers object-group pptp_dns
```

Select authentication method for PPTP server users:

```
esr(config-pptp)# authentication mode local
```

Specify security zone that user sessions will be related to:

```
esr(config-pptp)# security-zone VPN
```

Create PPTP users *Ivan* and *Fedor* for PPTP server:

```
esr(config-pptp)# username ivan
esr(config-pptp-user)# password ascii-text password1
esr(config-pptp-user)# enable
esr(config-pptp-user)# exit
esr(config-pptp)# username fedor
esr(config-pptp-user)# password ascii-text password2
esr(config-pptp-user)# enable
esr(config-pptp-user)# exit
esr(config-pptp)# exit
```

Enable PPTP server:

```
esr(config-pptp)# enable
```

When a new configuration is applied, the router will listen to 120.11.5.1:1723. To view PPTP server session status, use the following command:

```
esr# show remote-access status pptp server remote-workers
```

To view PPTP server session counters, use the following command:

```
esr# show remote-access counters pptp server remote-workers
```

To clear PPTP server session counters, use the following command:

```
esr# clear remote-access counters pptp server remote-workers
```

To end PPTP server session for user 'fedor', use one of the following commands:

```
esr# clear remote-access session pptp username fedor
esr# clear remote-access session pptp server remote-workers username fedor
```

To view PPTP server configuration, use the following command:

```
esr# show remote-access configuration pptp remote-workers
```

> **In addition to PPTP server creation, open TCP port 1723 designed for connection handling and enable GRE protocol (47) for the tunnel traffic in the firewall.**

15.2 Configuring server for remote access to corporate network via L2TP protocol

L2TP (Layer 2 Tunneling Protocol) is a sophisticated tunneling protocol used to support virtual private networks. L2TP encapsulates PPP frames into IP packets for transmission via global IP network, e.g. the Internet. L2TP may be used for tunnel establishment between two local area networks. L2TP uses an additional UDP connection for tunnel handling. L2TP protocol does not provide data encryption, therefore it is usually combined with an IPsec protocol group that provides security on a packet level.

15.2.1 Configuration algorithm

Step	Description	Command	Keys
1	Create L2TP server profile.	**esr(config)# remote-access l2tp <NAME>**	<NAME> – L2TP server profile name, set by the string of up to 31 characters.
2	Specify the description of the configured server (optional).	**esr(config-l2tp-server)# description <DESCRIPTION>**	<DESCRIPTION> – L2TP server description, set by the string of up to 255 characters.
3	Specify IP address that should be listened by L2TP server.	**esr(config-l2tp-server)# outside-address { object-group <NAME> | ip-address <ADDR> | interface { <IF> | <TUN> } }**	<OBJ-GROUP-NETWORK-NAME> – name of the profile having IP address that should be listened by L2TP server, set by the string of up to 31 characters; <ADDR> – range starting IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]; <IF> – router interface type and identifier; <TUN> – router tunnel type and number.

Step	Description	Command	Keys
4	Specify the IP address of the local gateway or disable firewall for the PPTP server	**esr(config-l2tp-server)# local-address { object-group <OBJ-GROUP-NETWORK -NAME> | ip-address <ADDR> }**	<OBJ-GROUP-NETWORK-NAME> – name of the IP addresses profile that includes local gateway IP address, set by the string of up to 31 characters; <ADDR> – range starting IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
5	Specify IP addresses list from which dynamic IP addresses are leased to remote users by L2TP.	**esr(config-l2tp-server)# remote-address { object-group <OBJ-GROUP-NETWORK -NAME> | address-range <FROM-ADDR>-<TO-ADDR> }**	<OBJ-GROUP-NETWORK-NAME> – name of the IP addresses profile that includes remote users 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].
6	Select L2TP clients authentication mode.	**esr(config-l2tp-server)# authentication mode { local | radius }**	• local – user authentication by local base. • radius – user authentication by RADIUS server base. The router must be configured to interact with a RADIUS-server, see section AAA RADIUS configuration algorithm
7	Allow necessary authentication methods for remote users	**esr(config-l2tp-server)# authentication method <METHOD>**	<METHOD> – authentication method, possible values: [chap, mschap, mschap-v2, eap, pap]. By default only chap is allowed.
8	Include the L2TP server in a security zone and configure interaction rules between zones (see section Firewall configuration).	**esr(config-l2tp-server)# security-zone <NAME>**	<NAME> – security zone name, set by the string of up to 31 characters.
9	Specify user name (when using local authentication base).	**esr(config-l2tp-server) username < NAME >**	<NAME> – user name, set by the string of up to 12 characters.

<table>
<tr><th>Step</th><th>Description</th><th>Command</th><th>Keys</th></tr>
<tr><td>10</td><td>Specify user password (when using local authentication base).</td><td>esr(config-l2tp-user) password ascii-text { <PASSWORD> | encrypted <PASSWORD> }</td><td><PASSWORD> – user password, set by the string of up to 32 characters.</td></tr>
<tr><td>11</td><td>Enable user (when using local authentication base).</td><td>esr(config-l2tp-user) enable</td><td></td></tr>
<tr><td>12</td><td>Select a key authentication method for IKE connection (optional).</td><td>esr(config-l2tp-server)# ipsec authentication method pre-shared-key</td><td></td></tr>
<tr><td>13</td><td>Specify a shared secret authentication key that should be the same for both parties of the tunnel.</td><td>esr(config-l2tp-server)# ipsec authentication pre-shared-key { ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> } | hexadecimal {<HEX> | encrypted <ENCRYPTED-HEX> } }</td><td><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.</td></tr>
<tr><td>14</td><td>Restrict the authentication and encryption methods used for the IKE protocol (optional).</td><td>esr(config-l2tp-server)# ipsec ike proposal <NAME></td><td><NAME> – name of the previously created IKE profile, set by the string of up to 31 characters.</td></tr>
<tr><td>15</td><td>Restrict the authentication and encryption methods used for the IPsec protocol (optional).</td><td>esr(config-l2tp-server)# ipsec proposal <NAME></td><td><NAME> – name of the previously created IPsec profile, set by the string of up to 31 characters.</td></tr>
<tr><td>16</td><td>Enable server.</td><td>esr(config-l2tp-server)# enable</td><td></td></tr>
<tr><td>17</td><td>Specify outgoing packets DSCP priority.</td><td>esr(config-l2tp-server)# dscp <DSCP></td><td><DSCP> – outgoing packets dscp priority [0..63].</td></tr>
<tr><td>18</td><td>Specify MTU size (MaximumTransmissionUnit) for the server (optional).
MTU above 1500 will be active only when using the 'system jumbo-frames' command.</td><td>esr(config-l2tp-server) mtu <MTU></td><td><MTU> – MTU value, takes values in the range of [1280..1500].

Default value: 1500.</td></tr>
</table>

Step	Description	Command	Keys
19	Define the list of DNS servers that will be used by remote users (optional).	**esr(config-l2tp-server)# dns-servers object-group <OBJ-GROUP-NETWORK -NAME >**	<OBJ-GROUP-NETWORK-NAME> – name of the IP addresses profile that includes required DNS servers addresses, set by the string of up to 31 characters.
20	Define the list of WINS servers that will be used by remote users (optional).	**esr(config-l2tp-server)# wins-servers object-group <OBJ-GROUP-NETWORK -NAME >**	<OBJ-GROUP-NETWORK-NAME> – name of the IP addresses profile that includes required WINS servers addresses, set by the string of up to 31 characters.

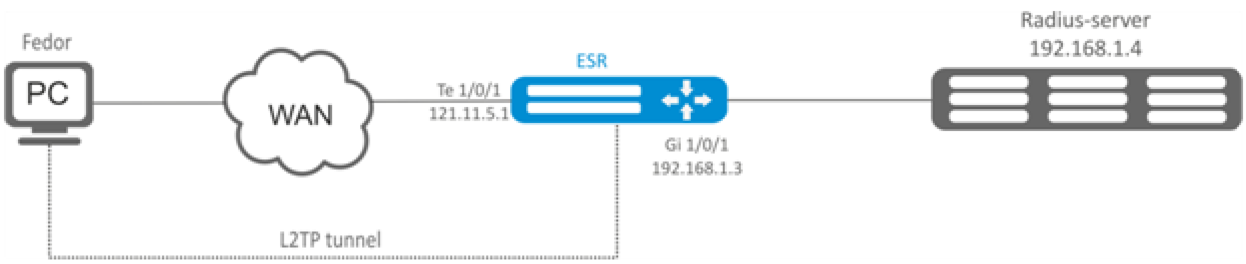
15.2.2 Configuration example

Objective:

Configure L2TP server on a router for remote user connection to LAN. Authentication is performed on RADIUS server.

- L2TP server address: 120.11.5.1;
- Gateway inside the tunnel: 10.10.10.1;
- Radius server address: 192.168.1.4.

For IPsec, key authentication method is used: key – 'password'.



Solution:

First, do the following:

- Configure RADIUS server connection;
- Configure zones for te1/0/1 and gi1/0/1 interfaces.
- Specify IP addresses for te1/0/1 and te1/0/1 interfaces.

Create address profile that contains local gateway address:

```
esr(config)# object-group network l2tp_local
esr(config-object-group-network)# ip address-range 10.10.10.1
esr(config-object-group-network)# exit
```

Create address profile that contains DNS servers:

```
esr(config)# object-group network pptp_dns
esr(config-object-group-network)# ip address-range 8.8.8.8
esr(config-object-group-network)# ip address-range 8.8.4.4
esr(config-object-group-network)# exit
```

Create L2TP server and map profiles listed above:

```
esr(config)# remote-access l2tp remote-workers
esr(config-l2tp)# local-address ip-address 10.10.10.1
esr(config-l2tp)# remote-address address-range 10.10.10.5-10.10.10.15
esr(config-l2tp)# outside-address ip-address 120.11.5.1
esr(config-l2tp)# dns-server object-group l2tp_dns
```

Select authentication method for L2TP server users:

```
esr(config-l2tp)# authentication mode radius
```

Specify security zone that user sessions will be related to:

```
esr(config-l2tp)# security-zone VPN
```

Specify authentication method for IKE phase 1 and define an authentication key.

```
esr(config-l2tp)# ipsec authentication method psk
esr(config-l2tp)# ipsec authentication pre-shared-key ascii-text password
```

Enable L2TP server:

```
esr(config-l2tp)# enable
```

When a new configuration is applied, the router will listen to IP address 120.11.5.1 and port 1701. To view L2TP server session status, use the following command:

```
esr# show remote-access status l2tp server remote-workers
```

To view L2TP server session counters, use the following command:

```
esr# show remote-access counters l2tp server remote-workers
```

To clear L2TP server session counters, use the following command:

```
esr# clear remote-access counters l2tp server remote-workers
```

To end L2TP server session for user 'fedor', use one of the following commands:

```
esr# clear remote-access session l2tp username fedor
esr# clear remote-access session l2tp server remote-workers username fedor
```

To view L2TP server configuration, use the following command:

```
esr# show remote-access configuration l2tp remote-workers
```

In addition to creating L2TP server, open UDP port 500, 1701, 4500 designed for connection handling and enable ESP (50) and GRE protocol (47) for the tunnel traffic in the firewall.

15.3 Configuring server for remote access to corporate network via OpenVPN protocol

OpenVPN is a sophisticated tool based on SSL that implements Virtual Private Networks (VPN), enables remote access and solves many different tasks related to data transmission security.

15.3.1 Configuration algorithm

Step	Description	Command	Keys
1	Create OpenVPN server profile.	**esr(config)# remote-access openvpn <NAME>**	<NAME> – OpenVPN server profile name, set by the string of up to 31 characters.
2	Specify the description of the configured server (optional).	**esr(config-openvpn-server)# description <DESCRIPTION>**	<DESCRIPTION> – OpenVPN server description, set by the string of up to 255 characters.
3	Define the subnet from which IP addresses are leased to users (only for tunnel ip).	**esr(config-openvpn-server)# network <ADDR/LEN>**	<ADDR/LEN> – subnet 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 [16..29].
4	Specify an encapsulated protocol.	**esr(config-openvpn-server)# protocol <PROTOCOL>**	<PROTOCOL> – encapsulation type, possible values: • TCP encapsulation in TCP segments; • UDP encapsulation in UDP datagrams.
5	Define type of connection with a private network via OpenVPN server.	**esr(config-openvpn-server)# tunnel <TYPE>**	<TYPE> – encapsulation protocol, takes the following values: • ip – point-to-point connection; • ethernet – L2 domain connection.

Step	Description	Command	Keys
6	Specify IP addresses list from which dynamic IP addresses are leased to remote users in L2 mode by OpenVPN server (only for tunnel ethernet).	**esr(config-openvpn-server)# address-range <FROM-ADDR>-<TO-ADDR>**	<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].
7	Include client connections via OpenVPN in L2 domain (only for tunnel ethernet).	**esr(config-openvpn-server)# bridge-group <BRIDGE-ID>**	<BRIDGE-ID> – bridge identifying number.
8	Specify certificates and keys.	**esr(config-openvpn-server)# certificate <CERTIFICATE-TYPE> <NAME>**	<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. <NAME> – certificate or key name, set by the string of up to 31 characters.
9	Select encryption algorithm used when data transmission.	**esr(config-openvpn-server)# encryption algorithm <ALGORITHM>**	<ALGORITHM> – encryption protocol identifier, may take values: 3des,blowfish128, aes128.
10	Include the OpenVPN server in a security zone and configure interaction rules between zones (see section Firewall configuration).	**esr(config-openvpn-server)# security-zone <NAME>**	<NAME> – security zone name, set by the string of up to 31 characters.
11	Define the additional parameters for a specified OpenVPN server user (when using a local base for user authentication).	**esr(config-openvpn-server)# username < NAME >**	<NAME> – user name, set by the string of up to 31 characters.
12	Define a subnet for the specified user of the OpenVPN server.	**esr(config-openvpn-user)# subnet <ADDR/LEN>**	<ADDR/LEN> – subnet 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].

Step	Description	Command	Keys
13	Define a static ip address for the specified OpenVPN server user.	**esr(config-openvpn-user)# ip address <ADDR>**	<ADDR> – address set in the following format: AAA.BBB.CCC.DDD – IP address of the subnet where AAA-DDD are set to [0..255].
14	Enable OpenVPN server profile.	**esr(config-openvpn-server)# enable**	
15	Enable data transmission blocking between clients (optional).	**esr(config-openvpn-server)# client-isolation**	
16	Set the maximum amount of simultaneous user sessions (optional).	**esr(config-openvpn-server)# client-max <VALUE>**	<VALUE> – maximum amount of users, takes values of [1..65535].
17	The mechanism of transmitted data compression between clients and the OpenVPN server is enabled (optional).	**esr(config-openvpn-server)# compression**	
18	Define the list of DNS servers that will be used by remote users (optional).	**esr(config-openvpn-server)# dns-server <ADDR>**	<ADDR> – DNS server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
19	Specify TCP/UDP port that will be listened by OpenVPN server (optional).	**esr(config-openvpn-server)# port <PORT>**	<PORT> – TCP/UDP port, takes values of [1..65535]. Default value: 1194.
20	Enable the default route advertising for OpenVPN connections, which leads to the replacement of the default route on the client side (optional).	**esr(config-openvpn-server)# redirect-gateway**	
21	Enable the advertising of specified subnets, the gateway is OpenVPN server IP address (optional).	**esr(config-openvpn-server)# route <ADDR/LEN>**	<ADDR/LEN> – subnet 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].
22	Set time interval after which the opposing party is considered to be unavailable (optional).	**esr(config-openvpn-server)# timers holdtime <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 120.

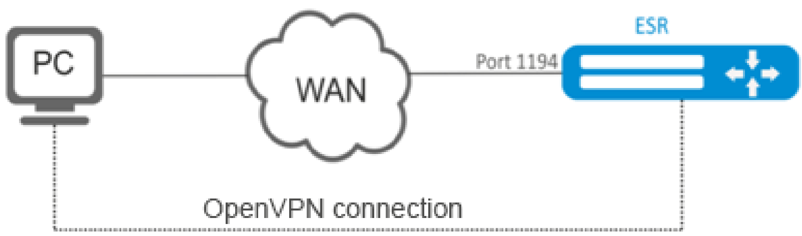
Step	Description	Command	Keys
23	Set the time interval after which the connection with the opposing party is checked (optional).	**esr(config-openvpn-server)# timers keepalive <TIME>**	<TIME> – time in seconds, takes values of [1..65535]. Default value: 10.
24	Allow multiple users with the same certificate to connect to the OpenVPN server.	**esr(config-openvpn-server)# duplicate-cn**	
25	Define the list of WINS servers that will be used by remote users (optional).	**esr(config-openvpn-server)# wins-server <ADDR>**	<ADDR> – WINS server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
26	Change the authentication algorithm for OpenVPN clients (optional).	**esr(config-openvpn-server)# authentication algorithm <ALGORITHM>**	<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.

15.3.2 Configuration example

Objective:

Configure Open VPN server in L3 mode on a router for remote user connection to LAN.

- OpenVPN server subnet: 10.10.100.0/24;
- Mode: L3;
- Authentication based on certificates.



Solution:

First, do the following:

- Prepare certificates and keys:
  - CA certificate;
  - OpenVPN server key and certificate;
  - Diffie-Hellman and HMAC key for TLS.
- Configure zone for te1/0/1 interface;
- Specify IP address for te1/0/1 interface.

Import certificates and keys via TFTP:

```
esr# copy tftp://192.168.16.10:/ca.crt certificate:ca/ca.crt
esr# copy tftp://192.168.16.10:/dh.pem certificate:dh/dh.pem
esr# copy tftp://192.168.16.10:/server.key certificate:server-key/server.key
esr# copy tftp://192.168.16.10:/server.crt certificate:server-crt/server.crt
esr# copy tftp://192.168.16.10:/ta.key certificate:ta/ta.key
```

Create OpenVPN server and a subnet for its operation:

```
esr(config)# remote-access openvpn AP
esr(config-openvpn)# network 10.10.100.0/24
```

Specify L3 connection type and encapsulation protocol:

```
esr(config-openvpn)# tunnel ip
esr(config-openvpn)# protocol tcp
```

Advert LAN subnets that will be available via OpenVPN connection and define DNS server:

```
esr(config-)# route 10.10.0.0/20
esr(config-openvpn)# dns-server 10.10.1.1
```

Specify previously imported certificates and keys that will be used with OpenVPN server:

```
esr(config-openvpn)# certificate ca ca.crt
esr(config-openvpn)# certificate dh dh.pem
esr(config-openvpn)# certificate server-key server.key
esr(config-openvpn)# certificate server-crt server.crt
esr(config-openvpn)# certificate ta ta.key
```

Specify security zone that user sessions will be related to:

```
esr(config-openvpn)# security-zone VPN
```

Select aes128 encryption algorithm:

```
esr(config-openvpn)# encryption algorithm aes128
```

Enable OpenVPN server:

```
esr(config-openvpn)# enable
```

When a new configuration is applied, the router will listen to port 1194 (used by default).

To view OpenVPN server session status, use the following command:

```
esr# show remote-access status openvpn server AP
```

To view OpenVPN server session counters, use the following command:

```
esr# show remote-access counters openvpn server AP
```

To clear OpenVPN server session counters, use the following command:

```
esr# clear remote-access counters openvpn server AP
```

To end OpenVPN server session for user 'fedor', use one of the following commands:

```
esr# clear remote-access session openvpn username fedor
esr# clear remote-access session openvpn server AP username fedor
```

To view OpenVPN server configuration, use the following command:

```
esr# show remote-access configuration openvpn AP
```

In addition to creating OpenVPN server, open TCP port 1194 in the firewall.

15.4 Configuring remote access client via PPPoE

PPPoE is a tunneling protocol that allows encapsulating IP PPP over Ethernet connections and has PPP connection software capabilities, which allows using it to establish virtual connections to a neighboring Ethernet device or a point-to-point connection that is used to transmit IP packets, and also works with PPP features. This allows applying conventional PPP-oriented software to configure the connection that uses not serial communication link but packet-oriented network (for example, Ethernet) to organize a classical connection with login and password for Internet connections. In addition, IP address on the opposite side of connection is assigned only when PPPoE connection is open, allowing the dynamic reuse of IP addresses.

15.4.1 Configuration algorithm

Step	Description	Command	Keys
1	Create a PPPoE tunnel and switch to its configuration mode.	**esr(config)# tunnel pppoe <PPPoE>**	<PPPoE> – tunnel sequence number from 1 to 10.
2	Specify the description of the configured client (optional).	**esr(config-pppoe)# description <DESCRIPTION>**	<DESCRIPTION> – PPPoE server description, set by the string of up to 255 characters.

Step	Description	Command	Keys
3	Specify the name of the VRF instance that will use the PPPoE client (optional).	**esr(config-pppoe)# ip vrf forwarding <VRF>**	<VRF> – VRF name, set by the string of up to 31 characters.
4	Specify the interface through which the PPPoE connection will be established.	**esr(config-pppoe)# interface <IF>**	<IF> – interface or interface group.
5	Specify user name and password for connection to PPPoE server.	**esr(config-pppoe)# username <NAME> password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }**	<NAME> – user name, set by the string of up to 31 characters; <CLEAR-TEXT> – password, set by the string of 8 to 16 characters; <ENCRYPTED-TEXT> – encrypted password, set by the string of [16..128] characters.
6	Include the PPPoE tunnel in a security zone and configure interaction rules between zones (see section Firewall configuration).	**esr(config-pppoe)# security-zone <NAME>**	<NAME> – security zone name, set by the string of up to 31 characters.
7	Enable a configured profile.	**esr(config-pppoe)# enable**	
8	Specify authentication method (optional).	**esr(config-pppoe)# authentication method <METHOD>**	<METHOD> – authentication method, possible values: chap, mschap, mschap-v2, eap, pap Default value: chap.
9	Enable the opt-out of receiving the default route from PPPoE server (optional).	**esr(config-pppoe)# ignore-default-route**	
10	Specify the time interval during which the statistics on the load is averaged (optional).	**esr(config-pppoe)# load-average <TIME>**	<TIME> – time interval in seconds from 5 to 150 (5 seconds by default).
11	Specify MTU size (MaximumTransmissionUnit) for PPPoE tunnel. MTU above 1500 will be active only when using the 'system jumbo-frames' command (optional).	**esr(config-pppoe)# mtu <MTU>**	<MTU> – MTU value, takes values in the range of: • for ESR-10/12V(F)/14VF – [1280..9600]; • for ESR-20/21 – [1280..9500]; • for ESR-100/200/1000/1200/1500/1511/1700 – [1280..10000] • for ESR-3100 – [1280..9190]. Default value: 1500.

Step	Description	Command	Keys
12	Change the number of failed data-link tests before breaking the session (optional).	**esr(config-pppoe)# ppp failure-count <NUM>**	<NUM> – the number of failed data-link tests, specified in the range [1..100]. Default value: 10.
13	Change the time interval in seconds after which the router sends a keepalive message (optional).	**esr(config-pppoe)# ppp timeout keepalive <TIME >**	<TIME> – time in seconds, takes values of [1..32767]. Default value: 10.
14	Override the MSS (Maximum segment size) field in incoming TCP packets (optional).	**esr(config-pppoe)# ip tcp adjust-mss <MSS>**	<MSS> – MSS value, takes values in the range of [500..1460]. Default value: 1460.
15	Enable recording of the current tunnel usage statistics (optional).	**esr(config-pppoe)# history statistics**	

It is also possible to configure the PPPoE client:

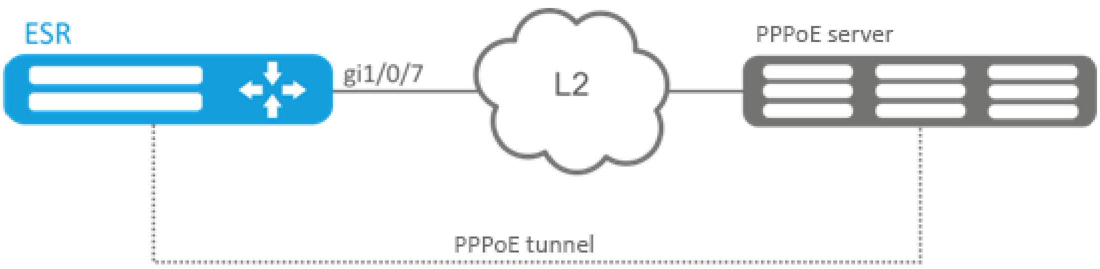
- QoS in basic or advanced mode (see section QoS management);
- proxy (see section HTTP/HTTPS traffic proxying);
- Traffic monitoring (see sections Netflow configuration and sFlow configuration);

15.4.2 Configuration example

Objective:

Configure PPPoE client on the router.

- Accounts for connection – tester;
- Account passwords – password;
- The connection should be established from the gigabitethernet 1/0/7 interface.



Solution:

Pre-configure PPPoE server with the accounts.

Enter the PPPoE client configuration mode and disable the firewall:

```
esr# configure
esr(config)# tunnel pppoe 1
esr(config-pppoe)# ip firewall disable
```

Specify user name and password for connection to PPPoE server:

```
esr(config-pppoe)# username tester password ascii-text password
```

Specify the interface through which the PPPoE connection will be established:

```
esr(config-pppoe)# interface gigabitethernet 1/0/7
esr(config- pppoe)# enable
```

To view the tunnel status, use the following command:

```
esr# show tunnels configuration pppoe 1
```

To view PPPoE client session counters, use the following command:

```
esr# show tunnels counters pppoe 1
```

15.5 Configuring remote access client via PPTP

PPTP (Point-to-Point Tunneling Protocol) is a point-to-point tunneling protocol that allows establishing secure connection with a server by creating a special tunnel in a common unsecured network. PPTP encapsulates PPP frames into IP packets for transmission via global IP network, e.g. the Internet. PPTP may be used for tunnel establishment between two local area networks. PPTP uses an additional TCP connection for tunnel handling.

15.5.1 Configuration algorithm

Step	Description	Command	Keys
1	Create a PPTP tunnel and switch to its configuration mode.	**esr(config)# tunnel pptp <INDEX>**	<INDEX> – tunnel identifier, set in the range of: [1..10].
2	Specify the description of the configured tunnel (optional).	**esr(config-pptp)# description <DESCRIPTION>**	<DESCRIPTION> – tunnel description, set by the string of up to 255 characters.
3	Specify VRF instance, in which the given PPTP tunnel will operate (optional).	**esr(config-pptp)# ip vrf forwarding <VRF>**	<VRF> – VRF name, set by the string of up to 31 characters.

Step	Description	Command	Keys
4	Include the PPTP tunnel in a security zone and configure interaction rules between zones or disable firewall (see section Firewall configuration).	**esr(config-pptp)# security-zone <NAME>**	<NAME> – security zone name, set by the string of up to 31 characters.
		esr(config-pptp)# ip firewall disable	
5	Set remote IP address for tunnel installation.	**esr(config-pptp)# remote address <ADDR>**	<ADDR> – local gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
6	Specify MTU size (MaximumTransmissionUnit) for the tunnel (optional).	**esr(config-pptp)# mtu <MTU>**	<MTU> – MTU value, takes values in the range of: • for ESR-10/12V(F)/14VF – [552..9600]; • for ESR-20/21 – [552..9500]; • for ESR-100/200/1000/1200/1500/1511/1700 – [552..10000] • for ESR-3100 – [552..9190]. Default value: 1500.
7	Specify the user and set an encrypted or unencrypted password to authenticate the remote party.	**esr(config-pptp)# username <NAME> password ascii-text { <WORD> | encrypted <HEX> }**	<NAME> – user name, set by the string of up to 31 characters. <WORD> – unencrypted password, set by the string of [8..64] characters, may include [0-9a-fA-F] characters. <HEX> – encrypted password, set by the string of [16..128] characters.
8	Enable the tunnel.	**esr(config-pptp)# enable**	
9	Override the MSS (Maximum segment size) field in incoming TCP packets (optional).	**esr(config-pptp)# ip tcp adjust-mss <MSS>**	<MSS> – MSS value, takes values in the range of [500..1460]. Default value: 1460.
10	Ignore the default route via the given PPTP tunnel (optional).	**esr(config-pptp)# ignore-default-route**	
11	Specify the time interval during which the statistics on the tunnel load is averaged (optional).	**esr(config-pptp)# load-average <TIME>**	<TIME> – interval in seconds, takes values of [5..150]. Default value: 5.

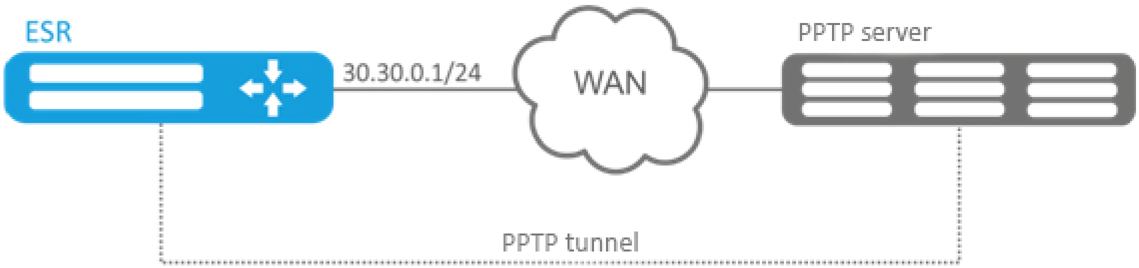
Step	Description	Command	Keys
12	Specify authentication method (optional).	**esr(config-pptp)# authentication method <METHOD>**	<METHOD> – authentication method, possible values: chap, mschap, mschap-v2, eap, pap Default value: chap.
13	Enable recording of the current tunnel usage statistics (optional).	**esr(config-pptp)# history statistics**	
14	Change the time interval in seconds after which the router sends a keepalive message (optional).	**esr(config-pptp)# ppp timeout keepalive <TIME >**	<TIME> – time in seconds, takes values of [1..32767]. Default value: 10.
15	Change the number of failed data-link tests before breaking the session (optional).	**esr(config-pptp)# ppp failure-count <NUM>**	<NUM> – the number of failed data-link tests, specified in the range [1..100]. Default value: 10.

15.5.2 Configuration example

Objective:

Configure PPTP tunnel on a router:

- PPTP server address: 20.20.0.1;
- account for connection – login: ivan, password: simplepass.



Solution:

Create PPTP tunnel:

```
esr(config)# tunnel pptp 1
```

Specify the account (Ivan user) to connect to the server:

```
esr(config-pptp)# username ivan password ascii-text simplepass
```

Specify the remote gateway:

```
esr(config-pptp)# remote address 20.20.0.1
```

Specify a security zone:

```
esr(config-pptp)# security-zone VPN
```

Enable PPTP tunnel:

```
esr(config-pptp)# enable
```

To view the tunnel status, use the following command:

```
esr# show tunnels status pptp
```

To view sent and received packet counters, use the following command:

```
esr# show tunnels counters pptp
```

To view the tunnel configuration, use the following command:

```
esr# show tunnels configuration pptp
```

15.6 Configuring remote access client via L2TP

L2TP (Layer 2 Tunneling Protocol) is a sophisticated tunneling protocol used to support virtual private networks. L2TP encapsulates PPP frames into IP packets for transmission via global IP network, e.g. the Internet. L2TP may be used for tunnel establishment between two local area networks. L2TP uses an additional UDP connection for tunnel handling. L2TP protocol does not provide data encryption, therefore it is usually combined with an IPsec protocol group that provides security on a packet level.

15.6.1 Configuration algorithm

Step	Description	Command	Keys
1	Create a L2TP tunnel and switch to its configuration mode.	**esr(config)# tunnel l2tp <INDEX>**	<INDEX> – tunnel identifier, set in the range of: [1..10].
2	Specify VRF instance, in which the given L2TP tunnel will operate (optional).	**esr(config-l2tp)# ip vrf forwarding <VRF>**	<VRF> – VRF name, set by the string of up to 31 characters.
3	Specify the description of the configured tunnel (optional).	**esr(config-l2tp)# description <DESCRIPTION>**	<DESCRIPTION> – tunnel description, set by the string of up to 255 characters.
4	Include the L2TP tunnel in a security zone and configure interaction rules between zones or disable firewall (see section Firewall configuration).	**esr(config-l2tp)# security-zone <NAME>**	<NAME> – security zone name, set by the string of up to 31 characters.
		esr(config-l2tp)# ip firewall disable	
5	Set remote IP address for tunnel installation.	**esr(config-l2tp)# remote address <ADDR>**	<ADDR> – local gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
6	Specify the user and set an encrypted or unencrypted password to authenticate the remote party.	**esr(config-l2tp)# username <NAME> password ascii-text { <WORD> | encrypted <HEX> }**	<NAME> – user name, set by the string of up to 31 characters. <WORD> – unencrypted password, set by the string of [8..64] characters, may include [0-9a-fA-F] characters. <HEX> – encrypted password, set by the string of [16..128] characters.
7	Select a key authentication method for IKE connection.	**esr(config-l2tp)# ipsec authentication method pre-shared-key**	

Step	Description	Command	Keys
8	Specify a shared secret authentication key that should be the same for both parties of the tunnel.	**esr(config-l2tp)# ipsec authentication pre-shared-key { ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> } | hexadecimal {<HEX> | encrypted <ENCRYPTED-HEX> } }**	<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.
9	Restrict the authentication and encryption methods used for the IKE protocol (optional).	**esr(config-l2tp)# ipsec ike proposal <NAME>**	<NAME> – name of the previously created IKE profile, set by the string of up to 31 characters.
10	Restrict the authentication and encryption methods used for the IPsec protocol (optional).	**esr(config-l2tp)# ipsec proposal <NAME>**	<NAME> – name of the previously created IPsec profile, set by the string of up to 31 characters.
11	Specify UDP port number on which the connection to the L2TP server is established (optional).	**esr(config-l2tp)# port <PORT>**	<PORT> – UDP port number, takes values of [1024..65535]. Default value: 1701.
12	Enable the tunnel.	**esr(config-l2tp)# enable**	
10	Specify MTU size (MaximumTransmissionUnit) for the tunnel (optional).	**esr(config-l2tp)# mtu <MTU>**	<MTU> – MTU value, takes values in the range of: • for ESR-10/12V(F)/14VF – [552..9600]; • for ESR-20/21 – [552..9500]; • for ESR-100/200/1000/1200/1500/1511/1700 – [552..10000] • for ESR-3100 – [552..9190]. Default value: 1500.
11	Ignore the default route via the given L2TP tunnel (optional).	**esr(config-l2tp)# ignore-default-route**	

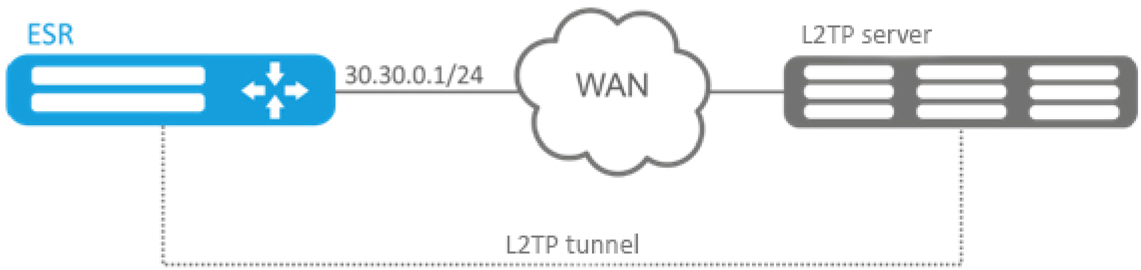
Step	Description	Command	Keys
12	Specify authentication method (optional).	**esr(config-l2tp)# authentication method <METHOD>**	<METHOD> – authentication method, possible values: chap, mschap, mschap-v2, eap, pap Default value: chap.
13	Specify the time interval during which the statistics on the tunnel load is averaged (optional).	**esr(config-l2tp)# load-average <TIME>**	<TIME> – interval in seconds, takes values of [5..150]. Default value: 5.
14	Change the time interval in seconds after which the router sends a keepalive message (optional).	**esr(config-l2tp)# ppp timeout keepalive <TIME >**	<TIME> – time in seconds, takes values of [1..32767]. Default value: 10.
15	Change the number of failed data-link tests before breaking the session (optional).	**esr(config-l2tp)# ppp failure-count <NUM>**	<NUM> – the number of failed data-link tests, specified in the range [1..100]. Default value: 10.
It is also possible to configure QoS in basic or advanced mode for the L2TP client (see section QoS management).			

15.6.2 Configuration example

Objective:

Configure PPTP tunnel on a router:

- PPTP server address: 20.20.0.1;
- account for connection – login: ivan, password: simplepass.



Solution:

Create L2TP tunnel:

```
esr(config)# tunnel l2tp 1
```

Specify the account (Ivan user) to connect to the server:

```
esr(config-l2tp)# username ivan password ascii-text simplepass
```

Specify the remote gateway:

```
esr(config-l2tp)# remote address 20.20.0.1
```

Specify a security zone:

```
esr(config-l2tp)# security-zone VPN
```

Specify IPsec authentication method:

```
esr(config-l2tp)# ipsec authentication method pre-shared-key
```

Specify IPsec security key:

```
esr(config-l2tp)# ipsec authentication pre-shared-key ascii-text password
```

Enable L2TP tunnel:

```
esr(config-l2tp)# enable
```

To view the tunnel status, use the following command:

```
esr# show tunnels status l2tp
```

To view sent and received packet counters, use the following command:

```
esr# show tunnels counters l2tp
```

To view the tunnel configuration, use the following command:

```
esr# show tunnels configuration l2tp
```

16 Service management

16.1 DHCP server configuration

Integrated DHCP server of the router allows configuring LAN device network settings. Router DHCP server is able to send additional options to network devices, for example:

- default-router – IP address of the router used as default gateway;
- domain-name – domain name which will be used by client while solving host names via domain name system (DNS);
- dns-server – list of domain name server addresses for the current network that should be known by the client. Server addresses are listed in descending order of their preference.

16.1.1 Configuration algorithm

Step	Description	Command	Keys
1	Enable IPv4/IPv6 DHCP server.	**esr(config)# ip dhcp-server [vrf <VRF>]**	<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.
		esr(config)# ipv6 dhcp-server [vrf <VRF>]	
2	Set the DSCP code value for the use in IP headers of DHCP server egress packets (optional).	**esr(config)# ip dhcp-server dscp <DSCP>**	<DSCP> – DSCP code value, takes values in the range of [0..63]. Default value: 61.

Step	Description	Command	Keys
3	Create pool of DHCP server IPv4/IPv6 addresses and switch to its configuration mode.	**esr(config)# ip dhcp-server pool <NAME> [vrf <VRF>]**	<NAME> – IPv4/IPv6 server profile name, set by the string of up to 31 characters. <VRF> – VRF instance name, within which the NTP server will operate. Set by the string of up to 31 characters.
		esr(config)# ipv6 dhcp-server pool <NAME> [vrf <VRF>]	
4	Specify IPv4/IPv6 address and mask for the subnet from which IPv4/IPv6 addresses pool will be allocated.	**esr(config-dhcp-server)# network <ADDR/LEN>**	<ADDR/LEN> – IP address and prefix of a 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].
		esr(config-ipv6-dhcp-server)# network <IPV6-ADDR/LEN>	<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].
5	Add IPv4/IPv6 addresses range to the address pool of configurable DHCP server.	**esr(config-dhcp-server)# address-range <FROM-ADDR>-<TO-ADDR>**	<FROM-ADDR> – range starting IP address; <TO-ADDR> – range ending IP address; The addresses are 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.
		esr(config-ipv6-dhcp-server)# address-range <FROM-ADDR>-<TO-ADDR>	<FROM-ADDR> – range starting IP address; <TO-ADDR> – range ending IP address; The addresses are defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].

Step	Description	Command	Keys
6	Add IPv4/IPv6 address for a specific physical address to the address pool of configurable DHCP server (optional).	**esr(config-dhcp-server)# address <ADDR> {mac-address <MAC> | client-identifier <CI>}**	<ADDR> – client 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, is defined as XX: XX: XX: XX: XX: XX where each part takes the values of [00..FF]. <CI> – client identifier according to DHCPOption61. Can be specified as follows: • HH:HH:HH:HH:HH:HH:HH: – client identifier in hexadecimal format and client MAC address; • STRING – text string from 1 to 64 characters.
		esr(config-ipv6-dhcp-server)# address <ADDR> mac-address <MAC>	<IPV6-ADDR> – client 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].
7	Specify the list of default gateway IPv4 addresses which will be transmitted by DHCP server to clients through DHCP option 3.	**esr(config-dhcp-server)# default-router <ADDR>**	<ADDR> – default gateway IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. Up to 8 IP addresses can be specified separated by commas.
8	Specify network domain DNS name. Domain name is transmitted to clients as part of DHCP option 15 (optional).	**esr(config-dhcp-server)# domain-name <NAME>** **esr(config-ipv6-dhcp-server)# domain-name <NAME>**	<NAME> – router domain name, set by the string from 1 to 255 characters.
9	Specify DNS server IPv4/IPv6 addresses list. The list is transmitted to clients as part of DHCP option 6 (optional).	**esr(config-dhcp-server)# dns-server <ADDR>**	<ADDR> – DNS server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. Up to 8 IP addresses can be specified separated by commas.

Step	Description	Command	Keys
		esr(config-ipv6-dhcp-server)# dns-server <IPV6-ADDR>	<IPV6-ADDR> – DNS server IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF]. Up to 8 IP addresses can be specified separated by commas.
10	Specify maximum IP addresses lease time (optional). If DHCP client requests the lease time that exceeds a maximum value, the time specified by the command will be set.	**esr(config-dhcp-server)# max-lease-time <TIME>** **esr(config-ipv6-dhcp-server)# max-lease-time <TIME>**	<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.
11	Specify the lease time for which a client will be given IP address (optional). This time will be used if a client did not request the certain lease time.	**esr(config-dhcp-server)# default-lease-time <TIME>** **esr(config-ipv6-dhcp-server)# default-lease-time <TIME>**	<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: 12 hours.
12	Create supplier class identifier (DHCP Option 60) (optional).	**esr(config)# ip dhcp-server vendor-class-id <NAME>** **esr(config)# ipv6 dhcp-server vendor-class-id <NAME>**	<NAME> – carrier class identifier, set by the string of up to 31 characters.
13	Specify specific supplier information (DHCP Option 43).	**esr(config-dhcp-vendor-id)# vendor-specific-options <HEX>** **esr(config-ipv6-dhcp-vendor-id)# vendor-specific-options <HEX>**	<HEX> – vendor-specific information, specified in hexadecimal format up to 128 symbols.
14	Specify NetBIOS server IP address (DHCP option 44) (optional).	**esr(config-dhcp-server)# netbios-name-server <ADDR>**	<ADDR> – NetBIOS server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. Up to 4 IP addresses can be specified.

Step	Description	Command	Keys
15	Specify TFTP server IP address (DHCP option 150) (optional).	**esr(config-dhcp-server)# tftp-server <ADDR>**	<ADDR> – DNS server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

16.1.2 Configuration example

Objective:

Configure DHCP server operation in a local network that belongs to the 'trusted' security zone. Specify IP address pool from 192.168.1.0/24 subnet for distribution to clients. Specify address lease time equal to 1 day. Configure transmission of the default route, domain name and DNS server addresses to clients using DHCP options.

Solution:

Create 'trusted' security zone and determine the inherence of the network interfaces being used to zones:

```
esr# configure
esr(config)# security zone trusted
esr(config-zone)# exit
```

Create address pool named 'Simple' and add IP address range intended for server clients lease into this pool. Specify parameters of the subnet that the pool belongs to, and the lease time for addresses:

```
esr# configure
esr(config)# ip dhcp-server pool Simple
esr(config-dhcp-server)# network 192.168.1.0/24
esr(config-dhcp-server)# address-range 192.168.1.100-192.168.1.125
esr(config-dhcp-server)# default-lease-time 1:00:00
```

Configure transfer of additional network parameters to clients:

- default route: 192.168.1.1;
- domain name: eltex.loc;
- DNS server list: DNS1: 172.16.0.1, DNS2: 8.8.8.8.

```
esr(config-dhcp-server)# domain-name "eltex.loc"
esr(config-dhcp-server)# default-router 192.168.1.1
esr(config-dhcp-server)# dns-server 172.16.0.1,8.8.8.8
esr(config-dhcp-server)# exit
```

To enable IP address distribution from the configurable pool by DHCP server, IP interface should be created on the router that belongs to the same subnet as the pool addresses.

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# security-zone trusted
esr(config-if-gi)# ip address 192.168.1.1/24
esr(config-if-gi)# exit
```

To enable DHCP message transmission to the server, create the respective port profiles including source port 68 and destination port 67 used by DHCP and create the allowing rule in the security policy for UDP packet transmission:

```
esr(config)# object-group service dhcp_server
esr(config-object-group-service)# port-range 67
esr(config-object-group-service)# exit
esr(config)# object-group service dhcp_client
esr(config-object-group-service)# port-range 68
esr(config-object-group-service)# exit
esr(config)# security zone-pair trusted self
esr(config-zone-pair)# rule 30
esr(config-zone-rule)# match protocol udp
esr(config-zone-rule)# match source-port dhcp_client
esr(config-zone-rule)# match destination-port dhcp_server
esr(config-zone-rule)# action permit
esr(config-zone-rule)# enable
esr(config-zone-rule)# exit
esr(config-zone-pair)# exit
```

Enable server operation:

```
esr(config)# ip dhcp-server
esr(config)# exit
```

To view the list of leased addresses, use the following command:

```
esr# show ip dhcp binding
```

To view the configured address pools, use the following commands:

```
esr# show ip dhcp server pool
esr# show ip dhcp server pool Simple
```

Configuration of settings for IPv6 is performed by analogy to IPv4.

16.2 Destination NAT configuration

Destination NAT (DNAT) function includes destination IP address translation for packets transferred through the network gateway.

DNAT is used for redirection of traffic, coming to a specific 'virtual' address in a public network, to a 'real' server in LAN located behind the network gateway. This function may be used for establishing a public access to servers located within the private network without any public network address.

16.2.1 Configuration algorithm

Step	Description	Command	Keys
1	Switch to the configuration mode of destination address translation service.	**esr(config)# nat destination**	
2	Create a pool of IP addresses and/or TCP/UDP ports with a specific name (optional).	**esr(config-dnat)# pool <NAME>**	<NAME> – NAT addresses pool name, set by the string of up to 31 characters.
3	Set the internal IP address which will replace a destination IP address.	**esr(config-dnat-pool)# ip address <ADDR>**	<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
4	Set the internal TCP/UDP port which will replace a destination TCP/UDP port.	**esr(config-dnat-pool)# ip port <PORT>**	<PORT> – TCP/UDP port, takes values of [1..65535].
5	Create a rule group with a specific name.	**esr(config-dnat)# ruleset <NAME>**	<NAME> – rule group name, set by the string of up to 31 characters.
6	Specify VRF instance, in which the given rule group will operate (optional).	**esr(config-dnat-ruleset)# ip vrf forwarding <VRF>**	<VRF> – VRF name, set by the string of up to 31 characters.
7	Set the rule group scope. The rules will be applied only to traffic coming from a certain zone or interface.	**esr(config-dnat-ruleset)# from { zone <NAME> | interface <IF> | tunnel <TUN> | default }**	<NAME> – isolation zone name; <IF> – device interface name; <TUN> – device tunnel name; default – denotes a group of rules for all traffic, the source of which did not fall under the criteria of other groups of rules.
8	Specify a rule with a certain number. The rules are proceeded in ascending order.	**esr(config-dnat-ruleset)# rule <ORDER>**	<ORDER> – rule number, takes values of [1..10000].

Step	Description	Command	Keys
9	Specify the profile of IP addresses {sender | recipient} for which the rule should work.	**esr(config-dnat-rule)# match [not] {source|destination}-address <OBJ-GROUP-NETWORK-NAME>**	<OBJ-GROUP-NETWORK-NAME> – IP addresses profile name, set by the string of up to 31 characters. 'Any' value points at any source IP address.
10	Specify the profile of services (TCP/UDP ports) {sender | recipient} for which the rule should work (optional).	**esr(config-dnat-rule)# match [not] {source|destination}-port <PORT-SET-NAME>**	<PORT-SET-NAME> – port profile name, set by the string of up to 31 characters. 'Any' value points at any source TCP/UDP port.
11	Set name or number of IP for which the rule should work (optional).	**esr(config-dnat-rule)# match [not] {protocol <TYPE> | protocol-id <ID> }**	<TYPE> – protocol type, takes the following values: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre. 'Any' value points at any protocol type. <ID> – IP identification number, takes values of [0x00-0xFF].
12	Specify the type and code of ICMP messages for which the rule should work (if ICMP is selected as protocol) (optional).	**esr(config-dnat-rule)# match [not] icmp {<ICMP_TYPE><ICMP_CODE> | <TYPE-NAME>}**	<ICMP_TYPE> – ICMP message type, takes values of [0..255]. <ICMP_CODE> – ICMP message code, takes values of [0..255]. 'Any' value points at any message code. <TYPE-NAME> – ICMP message type name.
13	Specify the action 'translation of source address and port' for the traffic meeting the requirements of 'match' commands.	**esr(config-dnat-rule)# action destination-nat { off | pool <NAME> | netmap <ADDR/LEN> }**	off – translation is disabled; pool<NAME> – name of the pool that contains IP addresses and/or TCP/UDP ports set; netmap <ADDR/LEN> – subnet IP address and mask used during translation. 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].
14	Activate a configured rule.	**esr(config-dnat-rule)# enable**	

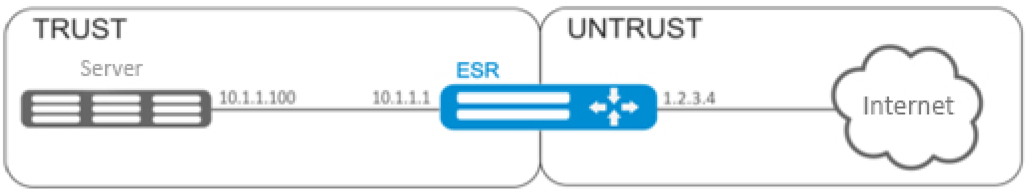
Step	Description	Command	Keys
15	Enable application layer session tracking for FTP, SIP, H323, netbios-ns, PPTP protocols (optional).	**esr(config)# ip firewall sessions tracking** **{<PROTOCOL> | sip [port <OBJECT-GROUP-SERVICE>] | all}**	all – enables application layer session tracking for all available protocols; <PROTOCOL> – application layer protocol whose sessions need to be monitored, takes values of [ftp, h323, pptp, netbios-ns]; <OBJECT-GROUP-SERVICE> – profile name of the TCP/UDP ports of the sip session, specified as a string of up to 31 characters. If the group is not specified, then sip sessions will be monitored for port 5060.
16	Enable IP address translation in application level headers (optional).	**esr(config)# nat alg {<PROTOCOL> | all}**	all – enables IP address translation in headers of all available protocols; <PROTOCOL> – application layer protocol in whose headers address translation should work, takes values of [ftp, h323, pptp, netbios-ns].

> ⚠ When using the *not* key, the rule will work for values which are not included in a specified profile.
> Each 'match' command may contain 'not' key. When using the key, packets that do not meet the given requirement will fall under the rule.
> For more information about router configuration, see 'CLI command reference guide'.

16.2.2 Destination NAT configuration example

Objective:

Establish access from the public network, that belongs to the 'UNTRUST' zone, to LAN server in 'TRUST' zone. Server address in LAN – 10.1.1.100. Server should be accessible from outside the network–address 1.2.3.4, access port 80.



Solution:

Create 'UNTRUST' and 'TRUST' security zones. Specify the inherence of the network interfaces being used to zones. Assign IP addresses to interfaces simultaneously.

```
esr# configure
esr(config)# security zone UNTRUST
esr(config-zone)# exit
esr(config)# security zone TRUST
esr(config-zone)# exit
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)#   security-zone TRUST
esr(config-if-gi)#   ip address 10.1.1.1/25
esr(config-if-gi)# exit
esr(config)# interface tengigabitethernet 1/0/1
esr(config-if-te)# ip address 1.2.3.4/29
esr(config-if-te)# security-zone UNTRUST
esr(config-if-te)# exit
```

Create IP address and port profiles required for configuration of the Firewall and DNAT rules.

- NET_UPLINK – public network address profile;
- SERVER_IP – local area network address profile;
- SRV_HTTP – port profile.

```
esr(config)# object-group network NET_UPLINK
esr(config-object-group-network)# ip address 1.2.3.4
esr(config-object-group-network)# exit
```

```
esr(config)# object-group service SRV_HTTP
esr(config-object-group-service)# port 80
esr(config-object-group-service)# exit
```

```
esr(config)# object-group network SERVER_IP
esr(config-object-group-network)# ip address 10.1.1.100
esr(config-object-group-network)# exit
```

Proceed to DNAT configuration mode and create destination address and port pool that will be used for translation of packet addresses coming to address 1.2.3.4 from the external network.

```
esr(config)# nat destination
esr(config-dnat)# pool SERVER_POOL
esr(config-dnat-pool)# ip address 10.1.1.100
esr(config-dnat-pool)# ip port 80
esr(config-dnat-pool)# exit
```

Create 'DNAT' rule set which will be used for address translation. In the set attributes, specify that the rules are applying only to packets coming from the 'UNTRUST' zone. Rule set includes data matching requirements for destination address and port (match destination-address, match destination-port) and for the protocol. Also, the set includes an action that applies to the data that satisfy all of the rules (action destination-nat). The rule set is applied with 'enable' command.

```
esr(config-dnat)# ruleset DNAT
esr(config-dnat-ruleset)# from zone UNTRUST
esr(config-dnat-ruleset)# rule 1
esr(config-dnat-rule)# match destination-address NET_UPLINK
esr(config-dnat-rule)# match protocol tcp
esr(config-dnat-rule)# match destination-port SRV_HTTP
esr(config-dnat-rule)# action destination-nat pool SERVER_POOL
esr(config-dnat-rule)# enable
esr(config-dnat-rule)# exit
esr(config-dnat-ruleset)# exit
esr(config-dnat)# exit
```

To transfer the traffic coming from 'UNTRUST' zone into 'TRUST' zone, create the respective pair of zones. Only DNAT-translated traffic with the destination address matching the 'SERVER_IP' specified in the profile should be transferred.

```
esr(config)# security zone-pair UNTRUST TRUST
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# match destination-address SERVER_IP
esr(config-zone-pair-rule)# match destination-nat
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
esr(config)# exit
```

Configuration changes will take effect when the configuration is applied:

```
esr# show ip nat destination pools
esr# show ip nat destination rulesets
esr# show ip nat proxy-arp
esr# show ip nat translations
```

16.3 Source NAT configuration

Source NAT (SNAT) function substitutes source address for packets transferred through the network gateway. When packets are transferred from LAN into public network, source address is substituted to one of the gateway public addresses. Additionally, source port substitution may be added to the source address. When packets are transferred back from public network to LAN, address and port are reverted to their original values.

SNAT function enables Internet access for computers located in LAN. At that, there is no need in assigning public IP addresses for these computers.

16.3.1 Configuration algorithm

Step	Description	Command	Keys
1	Switch to the configuration mode of source address translation service.	**esr(config)# nat source**	

Step	Description	Command	Keys
2	Create a pool of IP addresses and/or TCP/UDP ports with a specific name (optional).	**esr(config-snat)# pool <NAME>**	<NAME> – NAT addresses pool name, set by the string of up to 31 characters.
3	Set the range of IP addresses which will replace a source IP address.	**esr(config-snat-pool)# ip address-range <IP>[-<ENDIP>]**	<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.
4	Specify the range of external TCP/UDP ports which will replace a source TCP/UDP port.	**esr(config-snat-pool)# ip port-range <PORT>[-<ENDPORT>]**	<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.
5	Set external TCP/UDP port which will replace a source TCP/UDP port.	**esr(config-snat-pool)# ip port <PORT>**	<PORT> – TCP/UDP port, takes values of [1..65535].
6	Enable NAT persistent functions.	**esr(config-snat-pool)# persistent**	
7	Create a rule group with a specific name.	**esr(config-snat)# ruleset <NAME>**	<NAME> – rule group name, set by the string of up to 31 characters.
8	Specify VRF instance, in which the given rule group will operate (optionally).	**esr(config-snat-ruleset)# ip vrf forwarding <VRF>**	<VRF> – VRF name, set by the string of up to 31 characters.

Step	Description	Command	Keys
9	Set the rule group scope. The rules will be applied only to traffic coming to a certain zone or interface.	**esr(config-snat-ruleset)# to { zone <NAME> | interface <IF> tunnel <TUN> | | default }**	<NAME> – isolation zone name; <IF> – device interface name; <TUN> – device tunnel name default – denotes a group of rules for all traffic, the source of which did not fall under the criteria of other groups of rules.
10	Specify a rule with a certain number. The rules are proceeded in ascending order.	**esr(config-snat-ruleset)# rule <ORDER>**	<ORDER> – rule number, takes values of [1..10000].
11	Specify the profile of IP addresses {sender | recipient} for which the rule should work.	**esr(config-snat-rule)# match [not] {source|destination}-address <OBJ-GROUP-NETWORK-NAME>**	<OBJ-GROUP-NETWORK-NAME> – IP addresses profile name, set by the string of up to 31 characters. 'Any' value points at any source IP address.
12	Specify the profile of IP addresses {sender| recipient} for which the rule should work (optional).	**esr(config-snat-rule)# match [not] {source | destination}-port <PORT-SET-NAME>**	<PORT-SET-NAME> – port profile name, set by the string of up to 31 characters. 'Any' value points at any source TCP/UDP port.
13	Set name or number of IP for which the rule should work (optional).	**esr(config-snat-rule)# match [not] {protocol|protocol-id} <TYPE>**	<TYPE> – protocol type, takes the following values: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre. 'Any' value points at any protocol type. <ID> – IP identification number, takes values of [0x00-0xFF].
14	Specify the type and code of ICMP messages for which the rule should work (optional).	**esr(config-snat-rule)# match [not] icmp {<ICMP_TYPE><ICMP_CODE> | <TYPE-NAME>}**	<ICMP_TYPE> – ICMP message type, takes values of [0..255]. <ICMP_CODE> – ICMP message code, takes values of [0..255]. 'Any' value points at any message code. <TYPE-NAME> – ICMP message type name

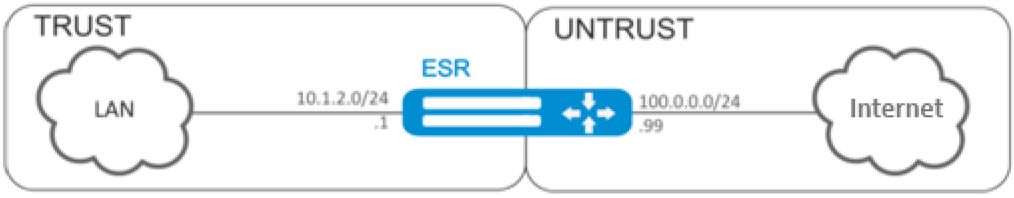
Step	Description	Command	Keys
15	Specify the action 'translation of source address and port' for the traffic meeting the requirements of 'match' command.	**esr(config-snat-rule)# action source-nat { off | pool <NAME> | netmap <ADDR/LEN> [static] | interface [FIRST_PORT – LAST_PORT] }**	off – translation is disabled; pool<NAME> – name of the pool that contains IP addresses and/or TCP/UDP ports set; netmap <ADDR/LEN> – subnet IP address and mask used during translation; static – option for static NAT organization. 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]. interface [FIRST_PORT – LAST_PORT] – specify the translation to the interface IP address. If the range of TCP/UDP ports is additionally specified, the translation will occur only for the sender TCP/UDP ports included in the specified range.
16	Activate a configured rule.	**esr(config-snat-rule)# enable**	
17	Enable application layer session tracking for FTP, SIP, H323, netbios-ns, PPTP protocols (optional).	**esr(config)# ip firewall sessions tracking** **{<PROTOCOL> | sip [port <OBJECT-GROUP-SERVICE>] | all}**	all – enables application layer session tracking for all available protocols; <PROTOCOL> – application layer protocol whose sessions need to be monitored, takes values of [ftp, h323, pptp, netbios-ns]; <OBJECT-GROUP-SERVICE> – profile name of the TCP/UDP ports of the sip session, specified as a string of up to 31 characters. If the group is not specified, then sip sessions will be monitored for port 5060.
18	Enable IP address translation in application level headers (optional).	**esr(config)# nat alg {<PROTOCOL> | all}**	all – enables IP address translation in headers of all available protocols; <PROTOCOL> – application layer protocol whose sessions need to be monitored, takes values of [ftp, h323, pptp, netbios-ns].

When using the *not* key, the rule will work for values which are not included in a specified profile.
Each 'match' command may contain 'not' key. When using the key, packets that do not meet the given requirement will fall under the rule.
For more information about router configuration, see 'CLI command reference guide'.

16.3.2 Configuration example 1

Objective:

Configure access for users in LAN 10.1.2.0/24 to public network using Source NAT function. Specify public network address range for SNAT 100.0.0.100-100.0.0.249.



Solution:

Begin configuration with creation of security zones, configuration of network interfaces and their inherence to security zones. Create 'TRUST' zone for LAN and 'UNTRUST' zone for public network.

```
esr# configure
esr(config)# security zone UNTRUST
esr(config-zone)# exit
esr(config)# security zone TRUST
esr(config-zone)# exit
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# ip address 10.1.2.1/24
esr(config-if-gi)# security-zone TRUST
esr(config-if-gi)# exit
esr(config)# interface tengigabitethernet 1/0/1
esr(config-if-te)# ip address 100.0.0.99/24
esr(config-if-te)# security-zone UNTRUST
esr(config-if-te)# exit
```

For SNAT function configuration and definition of rules for security zones, create 'LOCAL_NET' LAN address profile that includes addresses which are allowed to access the public network and 'PUBLIC_POOL' public network address profile.

```
esr(config)# object-group network LOCAL_NET
esr(config-object-group-network)# ip address-range 10.1.2.2-10.1.2.254
esr(config-object-group-network)# exit
esr(config)# object-group network PUBLIC_POOL
esr(config-object-group-network)# ip address-range 100.0.0.100-100.0.0.249
esr(config-object-group-network)# exit
```

To transfer traffic from 'TRUST' zone into 'UNTRUST' zone, create a pair of zones and add rules allowing traffic transfer in this direction. Additionally, there is a check in place to ensure that data source address belongs to 'LOCAL_NET' address range in order to limit the access to public network. Rules are applied with the *enable* command.

```
esr(config)# security zone-pair TRUST UNTRUST
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# match source-address LOCAL_NET
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
```

Configure SNAT service. First step is to create public network address pool for use with SNAT.

```
esr(config)# nat source
esr(config-snat)# pool TRANSLATE_ADDRESS
esr(config-snat-pool)# ip address-range 100.0.0.100-100.0.0.249
esr(config-snat-pool)# exit
```

Second step is to create SNAT rule set. In the set attributes, specify that the rules are applying only to packets transferred to public network–into the 'UNTRUST' zone. Rules include a check which ensures that data source address belongs to 'LOCAL_NET' pool.

```
esr(config-snat)# ruleset SNAT
esr(config-snat-ruleset)# to zone UNTRUST
esr(config-snat-ruleset)# rule 1
esr(config-snat-rule)# match source-address LOCAL_NET
esr(config-snat-rule)# action source-nat pool TRANSLATE_ADDRESS
esr(config-snat-rule)# enable
esr(config-snat-rule)# exit
esr(config-snat-ruleset)# exit
```

In order the router could response to the ARP requests for addresses from the public pool, you should launch ARP Proxy service. ARP Proxy service is configured on the interface that IP address from 'PUBLIC_POOL' public network address profile subnet belongs to.

```
esr(config)# interface tengigabitethernet 1/0/1
esr(config-if-te)# ip nat proxy-arp PUBLIC_POOL
```

To enable public network access for LAN devices, they should be configured for routing–10.1.2.1 should be defined as a gateway address.

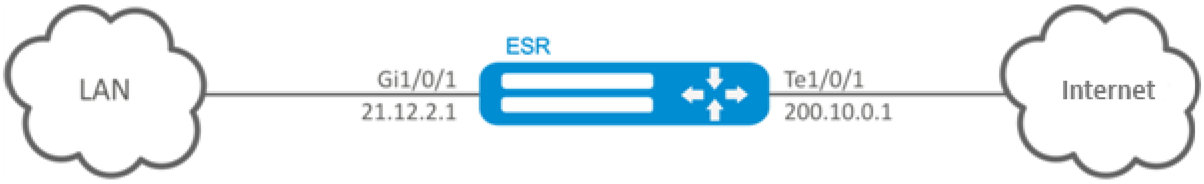
On the router, create the route for public network. Specify this route as a default using the following command.

```
esr(config)# ip route 0.0.0.0/0 100.0.0.1
esr(config)# exit
```

16.3.3 Configuration example 2

Objective:

Configure access for users in LAN 21.12.2.0/24 to public network using Source NAT function without the firewall. Public network address range for SNAT 200.10.0.100-200.10.0.249.



Solution:

Begin configuration with network interface configuration and disabling the firewall:

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# ip address 21.12.2.1/24
esr(config-if-gi)# ip firewall disable
esr(config-if-gi)# exit
```

```
esr(config)# interface tengigabitethernet 1/0/1
esr(config-if-te)# ip address 200.10.0.1/24
esr(config-if-te)# ip firewall disable
esr(config-if-te)# exit
```

For SNAT function configuration, create 'LOCAL_NET' LAN address profile that includes addresses which are allowed to access the public network and 'PUBLIC_POOL' public network address profile.

```
esr(config)# object-group network LOCAL_NET
esr(config-object-group-network)# ip address-range 21.12.2.2-21.12.2.254
esr(config-object-group-network)# exit

esr(config)# object-group network PUBLIC_POOL
esr(config-object-group-network)# ip address-range 200.10.0.100-200.10.0.249
esr(config-object-group-network)# exit
```

Configure SNAT service.

First step is to create public network address pool for use with SNAT:

```
esr(config)# nat source
esr(config-snat)# pool TRANSLATE_ADDRESS
esr(config-snat-pool)# ip address-range 200.10.0.100-200.10.0.249
esr(config-snat-pool)# exit
```

Second step is to create SNAT rule set. In the set attributes, specify that the rules are applying only to packets transferred to public network through te1/0/1 port. Rules include a check which ensures that data source address belongs to 'LOCAL_NET' pool:

```
esr(config-snat)# ruleset SNAT
esr(config-snat-ruleset)# to interface te1/0/1
esr(config-snat-ruleset)# rule 1
esr(config-snat-rule)# match source-address LOCAL_NET
esr(config-snat-rule)# action source-nat pool TRANSLATE_ADDRESS
esr(config-snat-rule)# enable
esr(config-snat-rule)# exit
esr(config-snat-ruleset)# exit
```

In order the router could response to the ARP requests for addresses from the public pool, launch ARP Proxy service. ARP Proxy service is configured on the interface that IP address from 'PUBLIC_POOL' public network address profile subnet belongs to:

```
esr(config)# interface tengigabitethernet 1/0/1
esr(config-if-te)# ip nat proxy-arp PUBLIC_POOL
```

To enable public network access for LAN devices, they should be configured for routing – 21.12.2.1 should be defined as a gateway address.

On the router, you should create the route for public network. Specify this route as a default using the following command:

```
esr(config)# ip route 0.0.0.0/0 200.10.0.254
esr(config)# exit
```

16.4 Static NAT configuration

Static NAT — static NAT sets a unique match between two addresses. In other words, when passing through the router the address is changed to another strictly specified one, one-to-one. The record about this translation is kept indefinitely until NAT reconfiguration is carried out on the router.

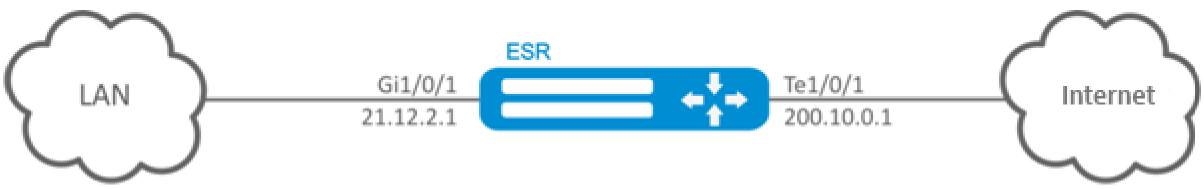
16.4.1 Configuration algorithm

Static NAT configuration is carried out by Source NAT means, the configuration algorithm is described in Section Source NAT configuration, configuration algorithm of the manual.

16.4.2 Static NAT configuration example

Objective:

Configure two-way and continuous translation from LAN for the addresses range of 21.12.2.100-21.12.2.150 to the public network 200.10.0.0/24. Public network address range for translation use – 200.10.0.100-200.10.0.150.



Solution:

Begin configuration with network interface configuration and disabling the firewall:

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# ip address 21.12.2.1/24
esr(config-if-gi)# ip firewall disable
esr(config-if-gi)# exit
```

```
esr(config)# interface tengigabitethernet 1/0/1
esr(config-if-te)# ip address 200.10.0.1/24
esr(config-if-te)# ip firewall disable
esr(config-if-te)# exit
```

For Static NAT configuration, create 'LOCAL_NET' LAN address profile, that includes local subnet, and 'PUBLIC_POOL' public network address profile.

```
esr(config)# object-group network LOCAL_NET
esr(config-object-group-network)# ip prefix 21.12.2.0/24
esr(config-object-group-network)# exit
```

```
esr(config)# object-group network PUBLIC_POOL
esr(config-object-group-network)# ip prefix 200.10.0.0/24
esr(config-object-group-network)# exit
```

The range of public network addresses for Static NAT use is specified in 'PROXY' profile:

```
esr(config)# object-group network PROXY
esr(config-object-group-network)# ip address-range 200.10.0.100-200.10.0.150
esr(config-object-group-network)# exit
```

Configure Static NAT service in SNAT configuration mode. In the set attributes, specify that the rules are applying only to packets transferred to public network through te1/0/1 port. The rules include data source address test for belonging to 'LOCAL_NET' pool and destination addresses test for belonging to 'PUBLIC_POOL' pool.

```
esr(config)# nat source
esr(config-snat)# ruleset SNAT
esr(config-snat-ruleset)# to interface te1/0/1
esr(config-snat-ruleset)# rule 1
esr(config-snat-rule)# match source-address LOCAL_NET
esr(config-snat-rule)# action source-nat netmap 200.10.0.0/24 static
esr(config-snat-rule)# enable
esr(config-snat-rule)# exit
esr(config-snat-ruleset)# exit
```

In order the router could response to the ARP requests for addresses from the 'PROXY' translation pool, launch ARP Proxy service. ARP Proxy service is configured on the interface that IP address from 'PROXY' address profile subnet belongs to:

```
esr(config)# interface tengigabitethernet 1/0/1
esr(config-if-te)# ip nat proxy-arp PROXY
```

To enable 200.10.0.0/24 network access for LAN devices, they should be configured for routing – 21.12.2.1 should be defined as a gateway address.

The configuration changes come into effect after applying the following commands:

```
esr# commit
Configuration has been successfully committed
esr# confirm
Configuration has been successfully confirmed
```

Active translations can be displayed by using the following command:

```
esr# show ip nat translations
```

16.5 HTTP/HTTPS traffic proxying

16.5.1 Configuration algorithm

Step	Description	Command	Keys
1	Create an object with a URL.	**esr(config)# object-group url <NAME>**	
2	Specify the set.	**esr(config-object-group-url)# url <URL>**	<URL> – web page, site address.
3	Create proxy profile.	**esr(config)# ip http profile <NAME>**	<NAME> – profile name.
4	Choose default action.	**esr(config-profile)# default action {deny|permit|redirect} [redirect-url <URL>]**	<URL> – address of the host to which requests will be sent.
5	Specify description (optional).	**esr(config-profile)# description <description>**	<description> – up to 255 characters.
6	Specify a remote or local URL list and type of operation (block/traffic pass/redirect) (optional).	**esr(config-profile)# urls {local|remote} <URL_OBJ_GROUP_NAME> action {deny|permit|redirect} [redirect-url <URL>]**	<URL_OBJ_GROUP_NAME> – specify the name of the object containing the URL set.

Step	Description	Command	Keys
7	Specify the remote server where the necessary URL lists are (optional).	**esr(config)# ip http proxy server-url <URL>**	<URL> – server address where remote url lists will be taken from.
8	Specify a listening port for proxying (optional).	**esr(config)# ip http proxy listen-ports <OBJ_GROUP_NAME>**	<OBJ_GROUP_NAME> – port profile name, set by string of up to 31 characters.
9	Specify a listening port for proxying (optional).	**esr(config)# ip https proxy listen-ports <OBJ_GROUP_NAME>**	<OBJ_GROUP_NAME> – port profile name, set by string of up to 31 characters.
10	Specify a base port for proxying (optional).	**esr(config)# ip https proxy redirect-port <PORT>**	<PORT> – port number, set in the range of [1..65535]. Default value: 3128.
11	Enable proxying on the interface based on the selected HTTP profile.	**esr(config-if)# ip http proxy <PROFILE_NAME>**	<PROFILE_NAME> – profile name.
12	Enable proxying on the interface based on the selected HTTPS profile.	**esr(config-if)# ip https proxy <PROFILE_NAME>**	<PROFILE_NAME> – profile name.
13	Create services lists which will be used during filtration.	**esr(config)# object-group service <obj-group-name>**	<obj-group-name> – service profile name, set by the string of up to 31 characters.
14	Specify services list description (optional).	**esr(config-object-group-service)# description <description>**	<description> – profile description, set by the string of up to 255 characters.
15	Add necessary services (TCP/UDP ports) to the list.	**esr(config-object-group-service)# port-range 3128-3135**	ESR proxy server uses for its operation the ports starting from the base port defined in step 10. The http proxy uses ports from base port to base port + the number of cpu of this ESR model - 1. For https proxy, the ports used are from base port + number of cpu of the given ESR model to base port + number of cpu of the given ESR model * 2 - 1.

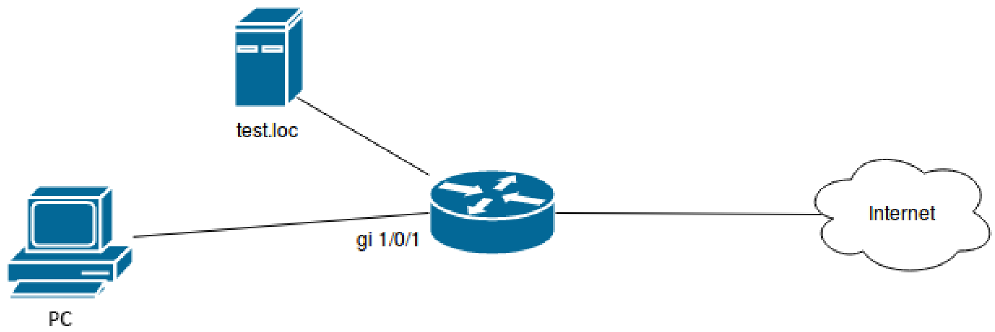
Step	Description	Command	Keys
16	Create an interzone interaction rule set.	**esr(config)# security zone-pair <src-zone-name1> self**	<src-zone-name> – security zone in which the interfaces with the ip http proxy or ip https proxy function are located. self – a predefined security zone for traffic entering the ESR itself.
17	Create an interzone interaction rule set.	**esr(config-zone-pair)# rule <rule-number>**	<rule-number> – 1..10000.
18	Specify rule description (optional).	**esr(config-zone-rule)# description <description>**	<description> – up to 255 characters..
19	Specify the given rule force.	**esr(config-zone-rule)# action <action> [log]**	<action> – permit. log – activation key for logging of sessions established according to this rule.
20	Set name of IP protocol for which the rule should work.	**esr(config-zone-rule)# match protocol <protocol-type>**	<protocol-type> – TCP. ESR proxy server uses ESR protocol.
21	Set the destination TCP/UDP ports profile for which the rule should work (if the protocol is specified).	**esr(config-zone-rule)# match [not] destination-port <obj-group-name>**	<obj-group-name> – name of the service profile created in step 12.
22	Create an interzone interaction rule.	**esr(config-zone-rule)# enable**	

If the Firewall function on the ESR is not forcibly disabled, create an allow rule for the Self zone.

16.5.2 HTTP proxy configuration example

Objective:

Organize URL filtering for a number of addresses using a proxy.



Solution:

Create a set of URLs to filter by. Configure a proxy filter and specify the actions for the created set of URLs:

```
esr# configure
esr(config)# object-group url test1
esr(config-object-group-url)# url http://speedtest.net/
esr(config-object-group-url)# url http://www.speedtest.net/
esr(config-object-group-url)# url https://speedtest.net/
esr(config-object-group-url)# url https://www.speedtest.net/
esr(config-object-group-url)# exit
```

Create a profile:

```
esr(config)# ip http profile list1
esr(config-profile)# default action permit
esr(config-profile)# urls local test1 action redirect redirect-url http://test.loc
esr(config-profile)# exit
```

Enable proxying on the interface by profile 'list1':

```
esr(config)# interface gi 1/0/1
esr(config-if)# ip http proxy list1
esr(config-if)# ip https proxy list1
```

If using Firewall, create permissive rules for it:

For example, ESR-20 with 4 CPUs is used.

For the http proxy, open ports 3128 to 3131.

For the https proxy, open ports 3132 to 3135.

Create a proxy server profile:

```
esr(config)# object-group service proxy
esr(config-object-group-service)# port-range 3128-3135
esr(config-object-group-service)# exit
```

Create a permissive interzonal interaction rule:

```
esr(config)# security zone-pair LAN self
esr(config-zone-pair)# rule 50
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol tcp
esr(config-zone-pair-rule)# match destination-port proxy
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
```

16.6 NTP configuration

NTP (*Network Time Protocol*) — network protocol for synchronizing the internal clock of equipment using IP networks, uses the UDP protocol for its operation, takes into account transmission times and uses algorithms to achieve high precision time synchronization.

16.6.1 Configuration algorithm

Step	Description	Command	Keys
1	Enable NTP.	**esr(config)# ntp enable**	
2	Set the IP address of the NTP server or NTP synchronization participant.	**esr(config)# ntp { server | peer } { <IP> }**	<IP> – destination IP address (gateway), defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
3	Set authentication key (optional).	**esr(config-ntp)# key <ID>**	<ID> – key identifier, set in the range of [1..255].
4	Set the maximum time interval between sending messages to the NTP server (optional).	**esr(config-ntp)# maxpoll <INTERVAL>**	<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).
5	Set the minimum time interval between sending messages to the NTP server (optional).	**esr(config-ntp)# minpoll <INTERVAL>**	<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).

Step	Description	Command	Keys
6	Mark this NTP server as preferred (optional).	**esr(config-ntp)# prefer**	
7	Define a list of trusted IP addresses with which ntp packets can be exchanged (optional).	**esr(config)# ntp access-addresses <NAME>**	<NAME> – IP addresses profile name, set by the string of up to 31 characters.
8	Specify the key ID from the key binding profile (optional).	**esr(config)# ntp authentication trusted-key <ID>**	<ID> – key ID from the key binding profile.
9	Specify the key binding profile name (optional).	**esr(config)# ntp authentication key-chain <WORD>**	<WORD> – key binding profile name.
10	Activate key-based authentication for NTP (optional).	**esr(config)# ntp authentication enable**	
11	Enable the mode of receiving broadcast messages from NTP servers for the global configuration and all existing VRFs (optional).	**esr(config)# ntp broadcast-client enable**	
12	Set the DSCP code value for the use in IP headers of NTP server egress packets (optional).	**esr(config)# ntp dscp <DSCP>**	<DSCP> – DSCP code value, takes values in the range of [0..63] Default value: 46.
13	Enable query-only mode that limits interaction via NTP for a certain profile of IP addresses (optional).	**esr(config)# ntp object-group query-only <NAME>**	<NAME> – IP addresses profile name, set by the string of up to 31 characters.
14	Enable serve-only mode that limits interaction via NTP for a certain profile of IP addresses (optional).	**esr(config)# ntp object-group serve-only <NAME>**	<NAME> – IP addresses profile name, set by the string of up to 31 characters.
15	Specify source-IP addresses for NTP packets for all peers (optional).	**esr(config)# ntp source address <ADDR>**	<ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Step	Description	Command	Keys
16	Set the current time and date manually (optional).	**esr# set date <TIME> [<DAY> <MONTH> [<YEAR>]]**	<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].

16.6.2 Configuration example

Objective:

Set the time synchronization from the NTP server.

ESR router IP address – 192.168.52.8,

NTP server IP address – 192.168.52.41.

Solution:

> **First, do the following:**
> - **specify security zone for gi1/0/1 interface;**
> - **configure the IP address for the gi1/0/1 interface to provide IP connectivity to the NTP server.**
>
> **Example:**

```
security zone untrust
exit
object-group service NTP
  port-range 123
exit
interface gigabitethernet 1/0/1
  security-zone untrust
  ip address 192.168.52.8/24
exit
security zone-pair untrust self
  rule 10
    action permit
    match protocol udp
    match destination-port NTP
    enable
  exit
exit
```

Main configuration step:

Enable synchronization of the system clock with remote servers:

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

NTP server configuration:

```
esr-(config)# ntp server 192.168.52.41
```

Specify the preference for this NTP server (optional):

```
esr-1000(config-ntp)# prefer
```

Specify the time interval between sending messages to the NTP server:

```
esr(config-ntp)# minpoll 4
esr(config-ntp)# end
esr# commit
esr# confirm
```

Command to view the current configuration of the NTP protocol:

```
esr# show ntp configuration
```

Command to view the current state of NTP servers (peers):

```
esr# show ntp peers
```

17 Monitoring

17.1 Netflow configuration

Netflow is a network protocol designed for traffic accounting and analysis. Netflow allows transmitting traffic information (source and destination address, port, quantity of information) from the network equipment (sensor) to the collector. Common server may serve as a collector.

> **In the current implementation, traffic dropped by the router for any reason will not be included in the statistics.**

17.1.1 Configuration algorithm

Step	Description	Command	Keys
1	Specify Netflow protocol version.	**esr(config)# netflow version <VERSION>**	<VERSION> – Netflow protocol version: 5, 9 and 10.
2	Set the maximum amount of observed sessions.	**esr(config)# netflow max-flows <COUNT>**	<COUNT> – amount of observed sessions, takes values of [10000..2000000]. Default value: 512000.

Step	Description	Command	Keys
3	Set the interval after which the information on active sessions is exported to the collector.	**esr(config)# netflow active-timeout <TIMEOUT>**	<TIMEOUT> – delay before sending active sessions information, set in seconds, takes the value of [5..36000]. Default value: 1800 seconds.
4	Set the interval after which the information on outdated sessions is exported to the collector.	**esr(config)# netflow inactive-timeout <TIMEOUT>**	<TIMEOUT> – delay before sending outdated sessions information, set in seconds, takes the value of [0..240]. Default value: 15 seconds.
5	Set the rate of the statistics sending to a Netflow collector.	**esr(config)# netflow refresh-rate <RATE>**	<RATE> – rate of the statistics sending, set in packets/flow, takes the value of [1..10000]. Default value: 10.
6	Enable Netflow on the router.	**esr(config)# netflow enable**	
7	Create the Netflow collector and switch to its configuration mode.	**esr(config)# netflow collector <ADDR>**	<ADDR> – collector IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
8	Set the Netflow service port on the statistics collection server.	**esr(config-netflow-host)# port <PORT>**	<PORT> – UDP port number in the range of [1..65535]. Default value: 2055.
9	Enable statistics sending to the Netflow server in the interface/tunnel/network bridge configuration mode.	**esr(config-if-gi)# ip netflow export**	

17.1.2 Configuration example

Objective:

Establish accounting for traffic from gi1/0/1 interface to be sent to the server via gi1/0/8 interface for processing purposes.

Solution:

First, configure addressing on interfaces.

Main configuration step:

Specify collector IP address:

```
esr(config)# netflow collector 10.10.0.2
```

Enable netflow statistics export collection for gi1/0/1 network interface:

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# ip netflow export
```

Enable netflow on the router:

```
esr(config)# netflow enable
```

To view the Netflow statistics, use the following command:

```
esr# show netflow statistics
```

Netflow configuration for traffic accounting between zones is performed by analogy to sFlow configuration; for description, see Section sFlow configuration.

17.2 sFlow configuration

Sflow is a computer network, wireless network and network device monitoring standard designed for traffic accounting and analysis.

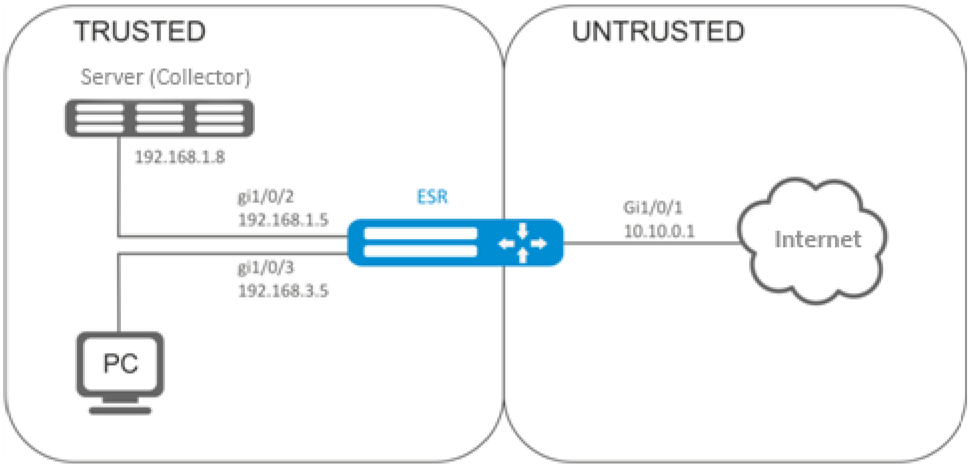
17.2.1 Configuration algorithm

Step	Description	Command	Keys
1	Set the rate of sending the unchanged user traffic packets to sFlow collector.	**esr(config)# sflow sampling-rate <RATE>**	<RATE> – rate of sending the user traffic packets to the collector, takes the value of [1..10000000]. If the rate value is 10, one of ten packets will be sent to the collector. Default value: 1000.
2	Set the interval after which the information on the network interface counters is obtained.	**esr(config)# sflow poll-interval <TIMEOUT>**	<TIMEOUT> – interval after which the information on the network interface counters is obtained, takes values of [1..10000]. Default value: 10 seconds.
3	Enable sFlow on the router.	**esr(config)# sflow enable**	
4	Create the sFlow collector and switch to its configuration mode.	**esr(config)# sflow collector <ADDR>**	<ADDR> – collector IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
5	Enable statistics sending to the sFlow server in the interface/ tunnel/network bridge configuration mode.	**esr(config-if-gi)# ip sflow export**	

17.2.2 Configuration example

Objective:

Establish accounting for traffic between 'trusted' and 'untrusted' zones.



Solution:

Create two security zones for ESR networks:

```
esr# configure
esr(config)# security zone TRUSTED
esr(config-zone)# exit
esr(config)# security zone UNTRUSTED
esr(config-zone)# exit
```

Configure network interfaces and identify their inherence to security zones:

```
esr(config)# interface gi1/0/1
esr(config-if-gi)# security-zone UNTRUSTED
esr(config-if-gi)# ip address 10.10.0.1/24
esr(config-if-gi)# exit
esr(config)# interface gi1/0/2-3
esr(config-if-gi)# security-zone TRUSTED
esr(config-if-gi)# exit
esr(config)# interface gi1/0/2
esr(config-if-gi)# ip address 192.168.1.5/24
esr(config-if-gi)# exit
esr(config)# interface gi1/0/3
esr(config-if-gi)# ip address 192.168.3.5/24
esr(config-if-gi)# exit
```

Specify collector IP address:

```
esr(config)# sflow collector 192.168.1.8
```

Enable sFlow protocol statistics export for all traffic within 'rule1' for TRUSTED-UNTRUSTED direction:

```
esr(config)# security zone-pair TRUSTED UNTRUSTED
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# action sflow-sample
esr(config-zone-pair-rule)# match protocol any
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# enable
```

Enable sFlow on the router:

```
esr(config)# sflow enable
```

SFlow configuration for traffic accounting from the interface is performed similar to Netflow configuration.

17.3 SNMP configuration

SNMP (*Simple Network Management Protocol*) is a protocol designed for device management in IP networks featuring TCP/UDP architecture. SNMP provides management data as variables that describe the configuration of a system being managed.

17.3.1 Configuration algorithm

Step	Description	Command	Keys
1	Enable SNMP server.	**esr(config)# snmp-server**	

Step	Description	Command	Keys
2	Specify community for the access via SNMPv2c.	**esr(config)# snmp-server community <COMMUNITY> [<TYPE>] [{ <IP-ADDR> | <IPV6-ADDR> }] [client-list <OBJ-GROUP-NETWORK-NAME>] [<VERSION>] [view <VIEW-NAME>] [vrf <VRF>]**	<COMMUNITY> – community for the access via SNMP; <TYPE> – access level: • **ro** – read-only access; • **rw** – read and write access. <IP-ADDR> – IP address of the client provided with the 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> – SNMP view profile name, set by the string of up to 31 characters; <VRF> – VRF instance name, set by the string of up to 31 characters, for which access will be granted.
3	Set the value of SNMP variable that contains contact information.	**esr(config)# snmp-server contact <CONTACT>**	<CONTACT> – contact information, sets by string with 255 characters length.
4	Set the DSCP code value for the use in IP headers of SNMP server egress packets (optional).	**esr(config)# snmp-server dscp <DSCP>**	<DSCP> – DSCP code value, takes values in the range of [0..63]. Default value: 63.
5	Enable router reboot by using snmp messages (optional).	**esr(config)# snmp-server system-shutdown**	
6	Create SNMPv3 user.	**esr(config)# snmp-server user <NAME>**	<NAME> – user name, set by the string of up to 31 characters.

Step	Description	Command	Keys
7	Set the value of SNMP value that contains the information on the device location.	**esr(config)# snmp-server location <LOCATION>**	<LOCATION> – information about equipment location, set by the string up to 255 characters.
8	Specify user access level via SNMPv3.	**esr(config-snmp-user)# access <TYPE>**	<TYPE> – access level: • **ro** – read-only access; • **rw** – read and write access.
9	Specify user security mode via SNMPv3.	**esr(config-snmp-user)# authentication access <TYPE>**	<TYPE> – security mode: • **auth** – used only for authentication; • **priv** – both authentication and data encryption are used.
10	Specify SNMPv3 queries authentication algorithm.	**esr(config-snmp-user)# authentication algorithm <ALGORITHM>**	<ALGORITHM> – encryption algorithm: • **md5** – password is hashed by md5 algorithm; • **sha1** – password is encrypted by sha1 algorithm.
11	Set the password for SNMPv3 queries authentication.	**esr(config-snmp-user)# authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }**	<CLEAR-TEXT> – password, set by the string of 8 to 16 characters; • **encrypted** – when specifying a command, an encrypted password is set: <ENCRYPTED-TEXT> – encrypted password of 8 to 16 bytes (from 16 to 32 characters) in hexadecimal format (0xYYYY ...) or (YYYY ...).
12	Enable filtration and set the profile of IP addresses from which SNMPv3 packets with the given SNMPv3 user name can be received.	**esr(config-snmp-user)# client-list <NAME>**	<NAME> – name of the previously conscious object-group, specified in a string of up to 31 characters.
13	Specify vrf for SNMPv3 user (optional).	**esr-21(config-snmp-user)# ip vrf forwarding <VRF>**	<VRF> – VRF instance name, set by the string of up to 31 characters, which contains SNMP notification collector.
14	Enable filtration and set IPv4/IPv6 address which is provided with the access to the router as the given SNMPv3 user.	**esr(config-snmp-user)# ip address <ADDR>**	<ADDR> – IP address of the client provided with the access, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Step	Description	Command	Keys
		esr(config-snmp-user)# ipv6 address <ADDR>	<IPV6-ADDR> – client IPv6 address, defined as X:X:X:X::X where each part takes values in hexadecimal format [0..FFFF].
15	Enable SNMPv3 user.	**esr(config-snmp-user)# enable**	Default value: process is disabled.
16	Specify the transmitted data encryption algorithm.	**esr(config-snmp-user)# privacy algorithm <ALGORITHM>**	<ALGORITHM> – encryption algorithm: • **aes128** – use AES-128 encryption algorithm; • **des** – use DES encryption algorithm.
17	Set password for the transmitted data encryption.	**esr(config-snmp-user)# privacy key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }**	<CLEAR-TEXT> – password, set by the string of 8 to 16 characters; <ENCRYPTED-TEXT> – encrypted password of 8 to 16 bytes (from 16 to 32 characters) in hexadecimal format (0xYYYY ...) or (YYYY ...).
18	Set the snmp view profile permitting or denying the access to one or another OID for user.	**esr(config-snmp-user)# view <VIEW-NAME>**	<VIEW-NAME> – name of SNMP view profile, on which based access to OID, set by the string up to 31 characters.
19	Enable SNMP notifications transmission to the specified IP address and switch to SNMP notifications configuration mode.	**esr(config)# snmp-server host { <IP-ADDR> | <IPV6-ADDR> } [vrf <VRF>]**	<IP-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.
20	Define the port of SNMP notifications collector on the remote server (optional).	**esr(config-snmp-host)# port <PORT>**	<PORT> – UDP port number in the range of [1..65535]. Default value: 162.

Step	Description	Command	Keys
21	Allow different types of SNMP notifications to be sent.	**esr(config)# snmp-server enable traps <TYPE>**	<TYPE> – type of filtered messages. May take the following values: config, entry, entry-sensor, environment, envmon, files-operations, flash, flash-operations, interfaces, links, ports, screens, snmp, syslog. Additional parameters depend on the filter type. See ESR-Series CLI command reference guide.
22	Create the snmp view profile permitting or denying the access to one or another OID for community (SNMPv2) and user (SNMPv3).	**esr(config)# snmp-server enable traps <TYPE>**	<VIEW-NAME> – SNMP view profile name, set by the string of up to 31 characters.

17.3.2 Configuration example

Objective:

Configure SNMPv3 server with authentication and data encryption for 'admin' user. ESR router IP address: 192.168.52.8, Syslog server IP address: 192.168.52.41.

Solution:

First, do the following:

- Specify zone for gi1/0/1 interface;
- Configure IP address for gi1/0/1 interface.

Main configuration step:

Enable SNMP server:

```
esr(config)# snmp-server
```

Create SNMPv3 user:

```
esr(config)# snmp-server user admin
```

Specify security mode:

```
esr(snmp-user)# authentication access priv
```

Specify authentication algorithm for SNMPv3 requests:

```
esr(snmp-user)# authentication algorithm md5
```

Set the password for SNMPv3 request authentication:

```
esr(snmp-user)# authentication key ascii-text 123456789
```

Specify the transmitted data encryption algorithm:

```
esr(snmp-user)# privacy algorithm aes128
```

Set password for the transmitted data encryption:

```
esr(snmp-user)# privacy key ascii-text 123456789
```

Enable SNMPv3 user:

```
esr(snmp-user)# enable
```

Define receiver-server of Trap-PDU messages:

```
esr(config)# snmp-server host 192.168.52.41
```

17.4 Zabbix-agent/proxy configuration

Zabbix-agent – agent designed to monitor the device, as well as execute remote commands from the Zabbix server. The agent can operate in two modes: passive and active. To operate in passive mode, by default, allow rule in the firewall – TCP protocol, port 10050. For active mode – TCP protocol, port 10051.

A Zabbix proxy is a process capable of collecting monitoring data from one or more monitored devices and sending this information to a Zabbix server.

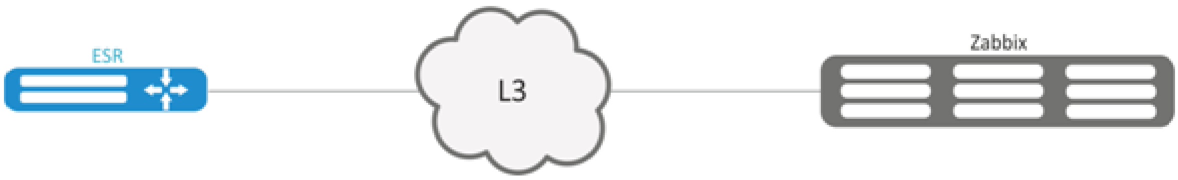
17.4.1 Configuration algorithm

Step	Description	Command	Keys
1	Switch to the agent/proxy configuration context.	**esr(config)# zabbix-agent** **esr(config)# zabbix-proxy**	

Step	Description	Command	Keys
2	Specify the host name (optional). For active mode, the name must match the host name on the Zabbix server.	**esr(config-zabbix)# hostname <WORD>** **esr(config-zabbix-proxy)# hostname <WORD>**	<WORD> – host name, set by the string of up to 255 characters.
3	Specify the address of the Zabbix server.	**esr(config-zabbix)# server <ADDR>** **esr(config-zabbix-proxy)# server <ADDR>**	<ADDR> – server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
4	Specify the server address for active checks (when using active mode).	**esr(config-zabbix)# active-server <ADDR> <PORT>** **esr(config-zabbix-proxy)# active-server <ADDR> <PORT>**	<ADDR> – server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. <PORT> – server port, set in the range of [1..65535]. Default value: 10051.
5	Specify the port that will be listened by the agent/proxy (optional).	**esr(config-zabbix)# port <PORT>** **esr(config-zabbix-proxy)# port <PORT>**	<PORT> – port that will be listened by zabbix agent/proxy, may take values in the range of [1..65535]. Default value: 10050.
6	Allow remote commands execution by Zabbix agent/proxy (when using active mode).	**esr(config-zabbix)# remote-commands** **esr(config-zabbix-proxy)# remote-commands**	
7	Specify the address from which the server will interact (optional).	**esr(config-zabbix)# source-address <ADDR>** **esr(config-zabbix-proxy)# source-address <ADDR>**	<ADDR> – server IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. Default value: nearest routing address.

Step	Description	Command	Keys
8	Specify the processing time for remote commands (optional).	**esr(config-zabbix)# timeout <TIME>** **esr(config-zabbix-proxy)# timeout <TIME>**	<TIME> – timeout, takes value in seconds [1..30]. Default value: 3. It is recommended to set the maximum value since some commands may take longer than the default. If the command is not completed within the specified time, processing of the command will be terminated.
9	Enable agent/proxy functionality.	**esr(config-zabbix)# enable** **esr(config-zabbix-proxy)# enable**	
10	Allow access to the router (to the self zone) on TCP ports 10050, 10051 from the appropriate firewall security zone. See Firewall configuration		

17.4.2 Zabbix-agent configuration example



Objective:

Configure the interaction between the agent and the server to execute remote commands from the server.

Solution:

In the context of the agent settings, specify the address of the Zabbix server, and the address from which the server will interact:

```
esr(config-zabbix)# server 192.168.32.101
esr(config-zabbix)# source-address 192.168.39.170
```

To activate the active mode, specify hostname, active-server, and also enable the execution of remote commands:

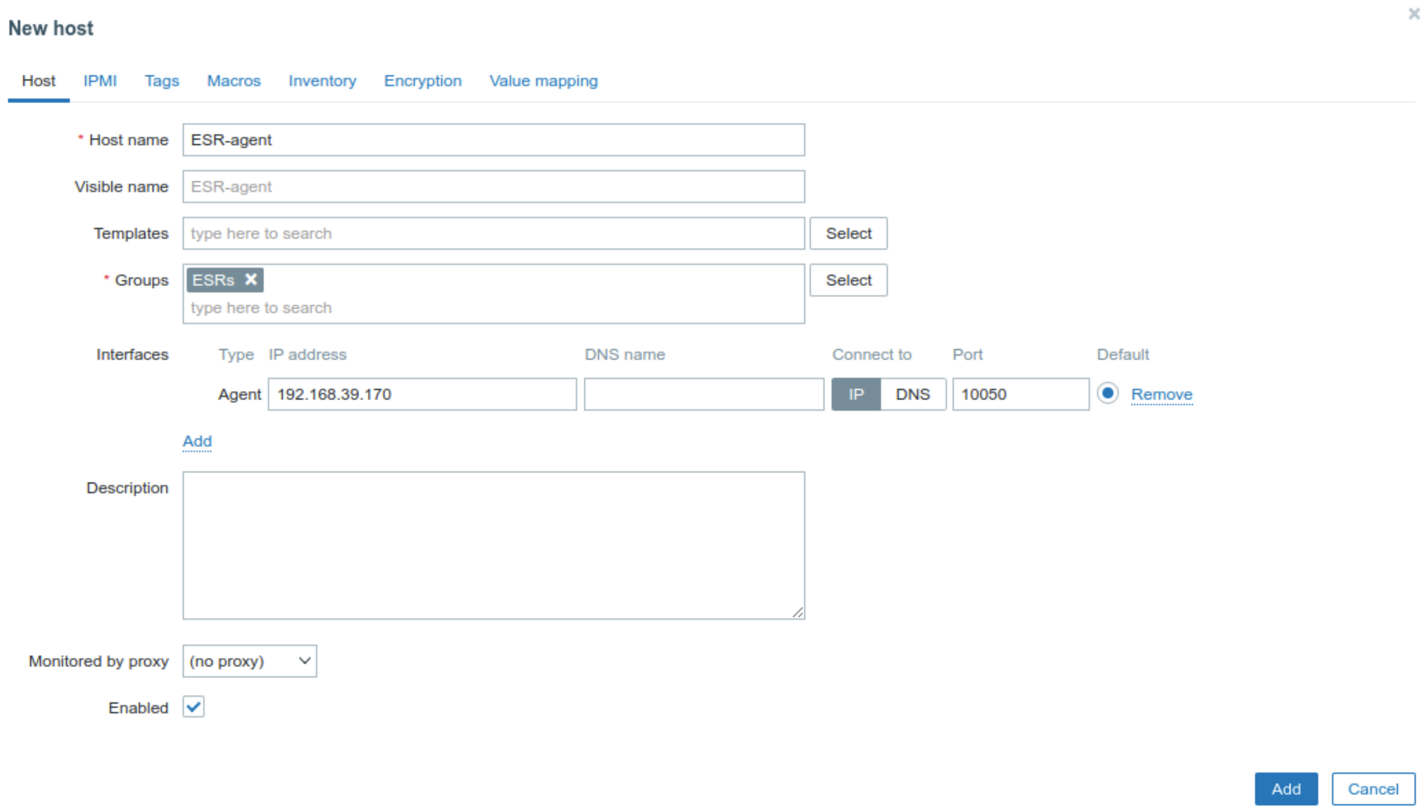
```
esr(config-zabbix)# hostname ESR-agent
esr(config-zabbix)# active-server 192.168.32.101
esr(config-zabbix)# remote-commands
```

Set the execution time of the remote commands, and activate the agent's functionality:

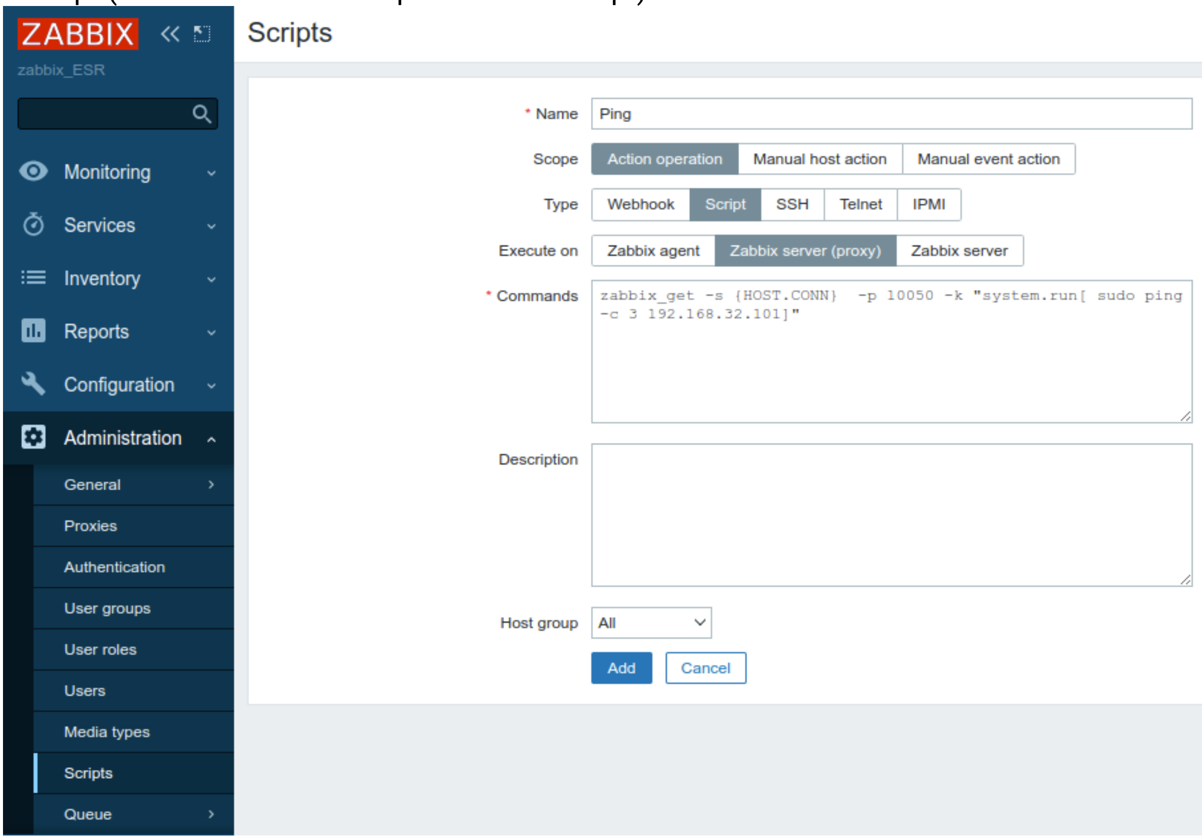
```
esr(config-zabbix)# timeout 30
esr(config-zabbix)# enable
```

17.4.3 Zabbix-server configuration example

Create the host:



Create the script (Administration -> Scripts-> Create Script):



ESR routers support execution of the following privilleged commands:

- **Ping**

```
zabbix_get -s {HOST.CONN}  -p 10050 -k "system.run[ sudo ping -c 3 192.168.32.101]"
```

The client (ESR) that received this command from the server will execute ping command to the specified host (in our example, up to 192.168.32.101) and return the result to the server.

> Using the '-c' key with the number of packets in the test is mandatory. Without this key, the ping command will not stop on its own and the test will not be considered complete.

- **Ping in VRF**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[sudo  netns -exec -n backup sudo ping 192.168.32.101 -c 5 -W 2 ]"
```

The command above will be executed in the specified VRF with backup name.

- **Fping**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[ sudo fping 192.168.32.101]"
```

The client (ESR) that received this command from the server will execute fping command to the specified host (in our example, up to 192.168.32.101) and return the result to the server.

- **Fping in VRF**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[sudo  netns-exec -n backup sudo fping 192.168.32.101 ]"
```

- **Traceroute**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[ sudo traceroute 192.168.32.101]
```

The client (ESR) that received this command from the server will execute traceroute command to the specified host (in our example, up to 192.168.32.101) and return the result to the server.

- **Traceroute in VRF**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[ sudo  netns-exec -n backup sudo
traceroute 192.168.32.179]"
```

- **Iperf**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[ sudo iperf -c 192.168.32.101 -u -b 100K
-i 1 -t 600]"
```

The client (ESR) that received this command from the server will execute iperf command to the specified server (in our example, up to 192.168.32.101) and return the result to the server.

- **Iperf in VRF**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[ sudo  netns-exec -n backup  sudo iperf
-c 192.168.32.101 -u -b 100K -i 1 -t 600]"
```

- **Nslookup**

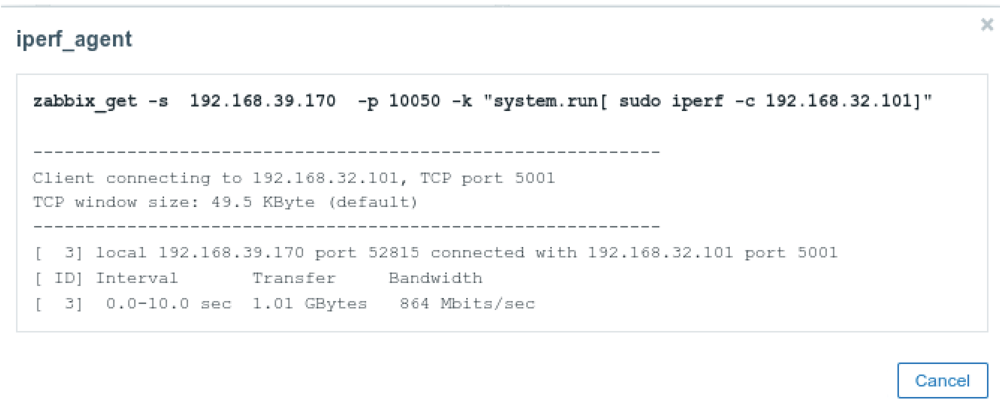
```
zabbix_get -s  {HOST.CONN} -p 10050 -k "system.run[sudo  nslookup ya.ru ]"
```

The client (ESR) that received this command from the server will execute nslookup command and return the result to the server.

- **Nslookup in VRF**

```
zabbix_get -s  {HOST.CONN} -p 10050 -k "system.run[sudo  netns-exec  sudo   nslookup
ya.ru ]"
```

Iperf command execution example:



It is also possible to execute commands that do not require privileges, such as: snmpget, cat, pwd, wget and others.

Example of the snmpget command execution:

17.5 Syslog configuration

Syslog (*System Log*) – standard for sending and registering messages about events occurring in the system is used in networks operating over IP.

17.5.1 Configuration algorithm

Step	Description	Command	Keys
1	Enable sending syslog messages as a snmp-trap to snmp server.	**esr(config)# syslog snmp**	
2	Enable or disable sending events of individual router processes operation to the snmp server (optional).	**esr(config-syslog-snmp)# match [not] process-name <PROCESS-NAME>**	<PROCESS-NAME> – see 'CLI command reference guide'. If allowing criteria are specified (match process-name), only messages of the specified processes are logged. If prohibiting criteria are specified (match not process-name), messages of all not prohibited processes are logged. By default, logging of messages from all processes is allowed.

Step	Description	Command	Keys
3	Set the severity for messages that will be sent to the SNMP server.	**esr(config)# syslog snmp <SEVERITY>**	<SEVERITY> – message importance level, takes the following 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.
4	Set display of syslog messages during remote connections (Telnet, SSH) (optional).	**esr(config)# syslog monitor**	
5	Enable or disable display of the events of individual router processes operation during remote connections (optional).	**esr(config-syslog-monitor)# match [not] process-name <PROCESS-NAME>**	<PROCESS-NAME> – described in point 2.
6	Set the severity for messages that will be displayed during remote connections.	**esr(config-syslog-monitor)# severity <SEVERITY>**	<SEVERITY> – described in point 3.
7	Enable display of syslog messages on console connection (optional).	**esr(config)# syslog console**	
8	Enable or disable display of the events of individual router processes operation during console connections (optional).	**esr(config-syslog-console)# match [not] process-name <PROCESS-NAME>**	<PROCESS-NAME> – described in point 2.
9	Set the severity for messages that will be displayed during console connections.	**esr(config-syslog-console)# severity <SEVERITY>**	<SEVERITY> – described in point 3.
10	Enable saving of syslog messages of a specified level of importance to the specified log file (when it is necessary to use local syslog file).	**esr(config)# syslog file <NAME>**	<NAME> – name of the file to which messages of a given level will be recorded, specified by string up to 31 characters.
11	Enable or disable saving of syslog messages of events of the operation of individual router processes (optional).	**esr(config-syslog-file)# match [not] process-name <PROCESS-NAME>**	<PROCESS-NAME> – described in point 2.

Step	Description	Command	Keys
12	Set the severity for messages that will be saved to the local syslog file (optional).	**esr(config-syslog-file)# severity <SEVERITY>**	<SEVERITY> – described in point 3.
13	Set maximum size of the log file (optional).	**esr(config)# syslog file-size <SIZE>**	<SIZE> – file size, takes value of [10..10000000] KB.
14	Set maximum number of files saved during rotation (optional).	**esr(config)# syslog max-files <NUM>**	<NUM> – maximal number of files, takes values of [1 .. 1000].
15	Enable sending of syslog messages to a remote syslog server (when it is necessary to send messages to a remote syslog server).	**esr(config)#syslog host <HOSTNAME> <ADDR><TRANSPORT>**	<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 command to delete all syslog servers; <ADDR> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]; <TRANSPORT> – data transfer protocol, optional parameter, takes values: • **TCP** – data transmission is carried out by TCP; • **UDP** – data transmission is carried out by UDP.
16	Specify IPv4/IPv6 address of the remote syslog server.	**esr(config-syslog-host)# remote-address { <ADDR> | <IPV6-ADDR> }**	<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].
17	Specify IPv4/IPv6 address of the router from which packets will be sent to the remote syslog server (optional).	**esr(config-syslog-host)# source-address { <ADDR> | <IPV6-ADDR> }**	<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 interface address from which packets will be sent to the remote syslog server.

Step	Description	Command	Keys
18	Specify transport protocol for packet transmission to the remote syslog server (optional).	**esr(config-syslog-host)# transport { tcp | udp }**	<VRF> – VRF instance name, set by the string of up to 31 characters, for which access will be granted. Default value: none (global routing table).
19	Specify name of the VRF instance within which packets will be sent to the remote syslog server (optional).	**esr(config-syslog-host)# vrf <VRF>**	
20	Specify number of the TCP/UDP port to which packets with syslog messages will be sent (optional).	**esr(config-syslog-host)# port <PORT>**	<PORT> – TCP/UDP port number to which packets with syslog messages will be sent. Default value: 514.
21	Enable or disable sending of the events of individual router processes operation to the remote server (optional).	**esr(config-syslog-host)# match [not] process-name <PROCESS-NAME>**	<PROCESS-NAME> – described in point 2.
22	Set the severity for messages that will be saved to local syslog file.	**esr(config-syslog-host)# severity <SEVERITY>**	<SEVERITY> – described in point 3.
23	Enable display of debugging messages during device boot (optional).	**esr(config)#syslog reload debugging**	
24	Enable logging of the entered user commands to the local syslog server (optional).	**esr(config)# syslog cli-commands**	
25	Enable message enumeration (optional).	**esr(config)#syslog sequence-numbers**	
26	Enable message date accuracy up to milliseconds (optional).	**esr(config)#syslog timestamp msec**	
27	Enable logging of failed authentications (optional).	**esr(config)#logging login on-failure**	
28	Enable logging of changes to the audit system settings (optional).	**esr(config)#logging syslog configuration**	
29	Enable logging of changes to the user settings (optional).	**esr(config)#logging userinfo**	

17.5.2 Configuration example

Objective:

Configure message sending for the following system events:

- failed user authentication;
- changes to the configuration of logging system events;
- start/stop of the system process;
- changes are made to the user profile.

ESR router IP address: 192.168.52.8, Syslog server IP address: 192.168.52.41. Use default settings for sending messages – UDP protocol, port 514.

Solution:

First, do the following:

- Specify zone for gi1/0/1 interface;
- Configure IP address for gi1/0/1 interface.

Main configuration step:

Create a file on the router for syslog, the level of messages for logging – info:

```
esr(config)# syslog file ESR info
```

Specify the IP address and parameters of the remote syslog server:

```
esr(config)# syslog host SERVER 192.168.17.30 info udp 514
```

Set the logging of failed authentication attempts:

```
esr(config)# logging login on-failure
```

Set the logging of syslog configuration changes:

```
esr(config)# logging syslog configuration
```

Set the logging of start/stop of the system process:

```
esr(config)# logging service start-stop
```

Set the logging of changes to the user profile:

```
esr(config)# logging userinfo
```

The configuration changes come into effect after applying the following commands:

```
esr# commit
Configuration has been successfully committed
esr# confirm
Configuration has been successfully confirmed
```

View the current syslog configuration:

```
esr# show syslog configuration
```

View the syslog entries:

```
esr# show syslog ESR
```

17.6 Integrity check

Integrity check involves checking the integrity of stored executable files.

17.6.1 Configuration process

Step	Description	Command	Keys
1	Launch system integrity check.	**esr# verify filesystem <detailed>**	**detailed** – detailed information output to the console.

17.6.2 Configuration example

Objective:

Check file system integrity.

Solution:

Launch integrity check:

```
esr# verify filesystem
Filesystem Successfully Verified
```

17.7 Router configuration file archiving

ESR routers have the option of local and/or remote configuration file copying by timer or when applying the configuration.

17.7.1 Configuration process

Step	Description	Command	Keys
1	Switch to the configuration file backup mode.	**esr(config)# archive**	
2	Set router configuration backup type (optional).	**esr(config-ahchive)# type <TYPE>**	<TYPE> – type of the router configuration backup. Takes the following values: • **local;** • **remote;** • **both.** Default value: remote.
3	Enable timer configuration backup mode (optional).	**esr(config-ahchive)# auto**	
4	Enable configuration backup after each successful configuration application mode (optional).	**esr(config-ahchive)# by-commit**	
5	Specify a path for remote copying of the router configuration (required for remote and both types).	**esr(config-ahchive)# path <PATH>**	<PATH> – defines the protocol, server address, location and prefix of the file name on the server
6	Set a period of time for automatic configuration backup (optional, relevant only for auto mode).	**esr(config-ahchive)# time-period <TIME>**	<TIME> – periodicity of automatic redundancy of the configuration, takes the value in minutes [1..35791394]. Default value: 720 minutes.
7	Set the maximum number of locally saved configuration backups (optional, relevant for local and both types).	**esr(config-ahchive)# count-backup <NUM>**	<NUM> – set the maximum number of locally saved configuration backups. Takes values in the range of [1..100]. Default value: 1.

17.7.2 Configuration example

Objective:

Configure local and remote backup of the router configuration once a day and upon successful configuration change. Remote copies should be sent to the TFTP server 172.16.252.77 in the esr-example subfolder. The maximum number of local copies is 30.

Solution:

For successful operation of remote configuration archiving, IP connectivity should be established between the router and the server, permissions for the passage of TFTP traffic over the network and saving files on the server should be configured.

Main configuration step:

Switch to the configuration backup mode:

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

Set local and remote configuration backup mode:

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

Configure the path for remote configuration backups and the maximum number of local backups:

```
esr(config-archive)# path tftp://172.16.252.77:/esr-example/esr-example.cfg
esr(config-archive)# count-backup 30
```

Set the interval for the configuration backup if there are no changes:

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

Enable archiving of router configuration by timer and upon successful configuration change:

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

After applying this configuration once a day and with each successful change of the router configuration, a configuration file with the 'esr-exampleYYYYMMDD_HHMMSS.cfg' name will be sent to the TFTP server. Also, on the router itself, in the flash:backup/ section, a file with the 'config_YYYYMMDD_HHMMSS' name will be created. When 30 files are accumulated in the flash:backup/ section, the oldest one will be deleted when creating a new one.

18 BRAS (Broadband Remote Access Server) management

18.1 Configuration algorithm

Step	Description	Command	Keys
1	Add RADIUS server to the list of used servers and switch to its configuration mode.	**esr(config)# radius-server host { <IP-ADDR> | <IPV6-ADDR> } [vrf <VRF>]** **esr(config-radius-server)#**	<IP-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]; <VRF> – VRF instance name, set by the string of up to 31 characters.
2	Set the password for authentication on remote RADIUS server.	**esr(config-radius-server)# key ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> }**	<TEXT> – string of [8..16] ASCII characters; <ENCRYPTED-TEXT> – encrypted password, [8..16] bytes size, set by the string of [16..32] characters.
3	Create AAA profile.	**esr(config)# aaa radius-profile <NAME>**	<NAME> – server profile name, set by the string of up to 31 characters.
4	Specify RADIUS server in AAA profile.	**esr(config-aaa-radius-profile)# radius-server host { <IP-ADDR> | <IPV6-ADDR> }**	<IP-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].
5	Create DAS server.	**esr(config)# das-server <NAME>**	<NAME> – DAS server name, set by the string of up to 31 characters.
6	Set the password for authentication on remote DAS server.	**esr(config-das-server)# key ascii-text {<TEXT>|encrypted <ENCRYPTED-TEXT> }**	<TEXT> – string of [8..16] ASCII characters; <ENCRYPTED-TEXT> – encrypted password, [8..16] bytes size, set by the string of [16..32] characters.

Step	Description	Command	Keys
7	Create AAA DAS profile.	**esr(config)# aaa das-profile <NAME>**	<NAME> – DAS profile name, set by the string of up to 31 characters.
8	Specify DAS server in DAs profile.	**esr(config-aaa-das-profile)# das-server <NAME>**	<NAME> – DAS server name, set by the string of up to 31 characters.
9	Configure BRAS.	**esr(config)# subscriber-control [vrf <VRF>]**	<VRF> – VRF instance name, set by the string of up to 31 characters, within which the user control will operate.
10	Select the profile of dynamic authorization servers to which CoS queries from PCRF will be sent.	**esr(config-subscriber-control)# aaa das-profile <NAME>**	<NAME> – DAS profile name, set by the string of up to 31 characters.
11	Select RADIUS server profile to obtain the user service parameters.	**esr(config-subscriber-control)# aaa services-radius-profile <NAME>**	<NAME> – RADIUS server profile name, set by the string of up to 31 characters.
12	Select RADIUS server profile to obtain the user session parameters.	**esr(config-subscriber-control)# aaa sessions-radius-profile <NAME>**	<NAME> – RADIUS server profile name, set by the string of up to 31 characters.
13	Set router IP address that will be used as source IP address in transmitted RADIUS packets.	**esr(config-subscriber-control)# nas-ip-address <ADDR>**	<ADDR> – source IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
14	Enable session authentication by MAC address (optional).	**esr(config-subscriber-control)# session mac-authentication**	
15	Organize transparent filter-based transmission of administrative traffic (DHCP, DNS and etc.).	**esr(config-subscriber-control)# bypass-traffic-a c l <NAME>**	<NAME> – name of the ACL being bound, set by the string of up to 31 characters.
16	Switch to the default service configuration mode.	**esr(config-subscriber-control)# default-service**	
17	Bind the specified QoS class to the default service.	**esr(config-subscriber-default-service)# class-map <NAME>**	<NAME> – name of the class being bound, set by the string of up to 31 characters.

Step	Description	Command	Keys
18	Specify a name of the URL list that will be used to filtrate HTTP/HTTPS traffic of non-authenticated users.	**esr(config-subscriber-default-service)# filter-name { local<LOCAL-NAME> | remote<REMOTE-NAME> }**	<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.
19	Specify the actions that should be applied for HTTP/HTTPS packets, whose URL is included in the list of URL assigned by the 'filter-name' command.	**esr(config-subscriber-default-service)# filter-action<ACT>**	<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.
20	Specify the actions that should be applied for HTTP/HTTPS packets, whose URL is not included in the list of URL assigned by the 'filter-name' command.	**esr(config-subscriber-default-service)# default -action<ACT>**	<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.
21	Enable user control profile.	**esr(config-subscriber-control)# enable**	
22	Change the identifier of a network interface (physical, sub interface or network bridge) (optional).	**esr(config-if)# location <ID>**	<ID> – network interface identifier, set by the string of up to 220 characters.
23	Enable user control on the interface.	**esr(config-if-gi)# service-subscriber-control {any| object-group <NAME>}**	<NAME> – IP addresses profile name, set by the string of up to 31 characters.
24	Enable iterative query of quota value when it expires for user services with a configured restriction on the amount of traffic or time (optional).	**esr(config-subscriber-control)# quota-expired-reauth**	
25	Enable session authentication by IP address (optional).	**esr(config-subscriber-control)# session ip-authentication**	

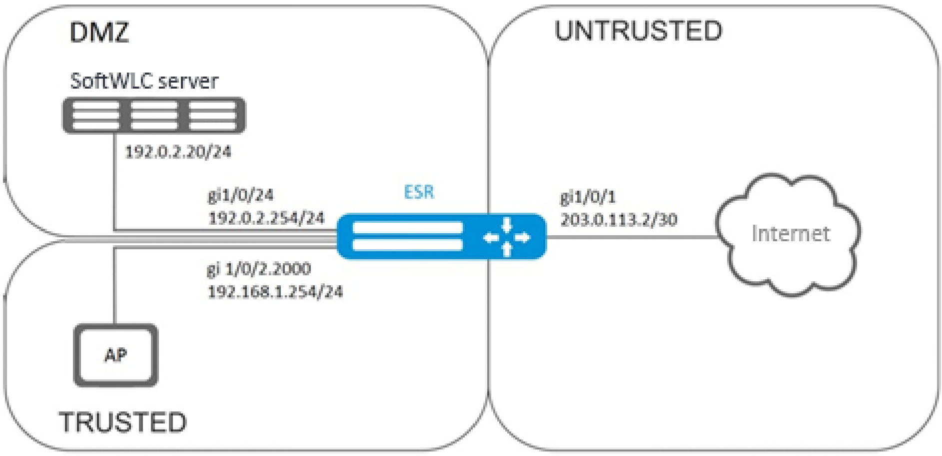
Step	Description	Command	Keys
26	Enable transparent transmission of backup traffic for BRAS (optional).	**esr(config-subscriber-control)# backup traffic-processing transparent**	
27	Specify the interval after which currently unused URL lists will be removed (optional).	**esr(config)# subscriber-control unused-filters-remove-delay <DELAY>**	<DELAY> – time interval in seconds, takes values of [10800..86400].
28	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 (optional).	**esr(config-subscriber-default-service)# session-timeout <SEC>**	<SEC> – time interval in seconds, takes values of [120..3600].
29	Specify the VRRP group on the basis of which user control service status is determined (primary/redundant) (optional).	**esr(config-subscriber-control)# vrrp-group <GRID>**	<GRID> – VRRP router group identifier, takes values in the range of [1..32].
30	Define destination TCP ports from which the traffic will be redirected to the router HTTP Proxy server (optional).	**esr(config-subscriber-control)# ip proxy http listen-ports <NAME>**	<NAME> – TCP/UDP ports profile name, set by the string of up to 31 characters.
31	Define HTTP Proxy server port on the router (optional).	**esr(config-subscriber-control)# ip proxy http redirect-port <PORT>**	<PORT> – port number, set in the range of [1..65535].
32	Define destination TCP ports from which the traffic will be redirected to the router HTTPS Proxy server (optional).	**esr(config-subscriber-control)# ip proxy https listen-ports <NAME>**	<NAME> – TCP/UDP ports profile name, set by the string of up to 31 characters.
33	Define HTTPS Proxy server port on the router (optional).	**esr(config-subscriber-control)# ip proxy https redirect-port <PORT>**	<PORT> – port number, set in the range of [1..65535].
34	Set router IP address that will be used as source IP address in HTTP/HTTPS packets transmitted by Proxy server (optional).	**esr(config-subscriber-control)# ip proxy source-address <ADDR>**	<ADDR> – source IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].
35	Specify URL address of the server providing lists of traffic filtration applications (optional).	**esr(config)# subscriber-control apps-server-url <URL>**	<URL> – reference address, set by the string from 8 to 255 characters.

Step	Description	Command	Keys
36	Enable the application control on the interface (optional).	**esr(config-if-gi)# subscriber-control application-filter <NAME>**	<NAME> – application profile name, set by the string of up to 31 characters.
37	Set/clear the upper bound of BRAS sessions amount (optional).	**esr(config-subscriber-control)# thresholds sessions-number high <Threshold>**	<Threshold> – number of BRAS sessions: • [0-50000] – for ESR-1700; • [0-10000] – for ESR-1200/1000/1500/1511/3100/3200; • [0-1000] – for ESR-100/200.
38	Set/clear the lower bound of BRAS sessions amount (optional).	**esr(config-subscriber-control)# thresholds sessions-number low <Threshold>**	<Threshold> – number of BRAS sessions: • [0-50000] – for ESR-1700; • [0-10000] – for ESR-1200/1000/1500/1511/3100/3200; • [0-1000] – for ESR-100/200.

18.2 Example of configuration with SoftWLC

Objective:

Provide access to the Internet only to authorized users.



Solution:

SoftWLC server keeps accounts data and tariff plan parameters. For more detailed information on installation and configuring SoftWLC server, use the following links:

SoftWLC – general SoftWLC article;

SoftWLC installation and update – installation of SoftWLC from repositories.

The BRAS license is obligatory for router, after its activation you can start configuring the device.

Create 3 security zones, according to the network structure:

```
esr# configure
esr(config)# security zone trusted
esr(config-zone)# exit
esr(config)# security zone untrusted
esr(config-zone)# exit
esr(config)# security zone dmz
esr(config-zone)# exit
```

Configure public port parameters and assign its default gateway:

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# security-zone untrusted
esr(config-if-gi)# ip address 203.0.113.2/30
esr(config-if-gi)# service-policy dynamic upstream
esr(config-if-gi)# exit
esr(config)# ip route 0.0.0.0/0 203.0.113.1
```

Configure port in direction to the SoftWLC server:

```
esr (config)# interface gigabitethernet 1/0/24
esr (config-if-gi)# security-zone dmz
esr (config-if-gi)# ip address 192.0.2.1/24
esr (config-if-gi)# exit
```

Configure port for Wi-Fi access point connection:

```
esr(config)# bridge 2
esr(config-bridge)# security-zone trusted
esr(config-bridge)# ip address 192.168.0.254/24
esr(config-bridge)# ip helper-address 192.0.2.20
esr(config-bridge)# service-subscriber-control object-group users
esr(config-bridge)# location ssid1
esr(config-bridge)# enable
esr(config-bridge)# exit
esr(config)# interface gigabitethernet 1/0/2.2000
esr(config-subif)# bridge-group 1
esr(config-subif)# exit
esr(config)# interface gigabitethernet 1/0/2
esr(config-if-gi)# service-policy dynamic downstream
esr (config-if-gi)# exit
```

Customer connection must be implemented through sub-interfaces to bridges. Selection of tariff plan depends on Location parameter (see bridge 2 configuration).

The module which is responsible for AAA operations is based on eltex-radius and available by SoftWLC IP address. Numbers of ports for authentication and accounting in the example below are the default values for SoftWLC.

Define parameters for interaction with the module:

```
esr(config)# radius-server host 192.0.2.20
esr(config-radius-server)#   key ascii-text password
esr(config-radius-server)#   auth-port 31812
esr (config-radius-server)#  acct-port 31813
esr (config-radius-server)# exit
```

Create AAA profile:

```
esr(config)# aaa radius-profile RADIUS
esr(config-aaa-radius-profile)# radius-server host 192.0.2.20
esr(config-aaa-radius-profile)# exit
```

Specify access parameters to the DAS (Direct-attached storage) server:

```
esr(config)# object-group network server
esr(config-object-group-network)#   ip address-range 192.0.2.20
esr(config-object-group-network)# exit
esr(config)# das-server CoA
esr(config-das-server)#   key ascii-text password
esr(config-das-server)#   port 3799
esr(config-das-server)#   clients object-group server
esr(config-das-server)# exit
esr(config)# aaa das-profile CoA
esr(config-aaa-das-profile)#   das-server CoA
esr(config-aaa-das-profile)# exit
```

The traffic from trusted zone is blocked before authentication as well as DHCP and DNS requests. Configure allowing rules in order to pass DHCP and DNS requests:

```
esr(config)# ip access-list extended DHCP
esr(config-acl)#  rule 10
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol udp
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# match source-port 68
esr(config-acl-rule)# match destination-port 67
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# rule 11
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol udp
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# match source-port any
esr(config-acl-rule)# match destination-port 53
esr(config-acl-rule)# enable
esr(config-acl-rule)#exit
esr(config-acl)# exit
```

Then, create rules for redirecting to portal and passing traffic to the Internet:

```
esr(config)# ip access-list extended WELCOME
esr(config-acl)# rule 10
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol any
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# exit
esr (config)# ip access-list extended INTERNET
esr(config-acl)#  rule 10
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol any
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# exit
```

Specify web resources which are available without authorization:

```
esr(config)# object-group url defaultservice
esr(config-object-group-url)# url http://eltex.nsk.ru
esr(config-object-group-url)# exit
```

The URL filtering lists are kept on SoftWLC server (change only IP address of SoftWLC server, if addressing is different from the example. Leave the rest of URL without changes):

```
esr(config)# subscriber-control filters-server-url http://192.0.2.20:7070/Filters/file/
```

Configure and enable BRAS, define NAS IP as address of the interface interacting with SoftWLC (gigabitethernet 1/0/24 in the example):

```
esr(config)# subscriber-control
esr(config-subscriber-control)#  aaa das-profile CoA
esr(config-subscriber-control)#  aaa sessions-radius-profile RADIUS
esr(config-subscriber-control)#  nas-ip-address 192.0.2.1
esr(config-subscriber-control)#  session mac-authentication
esr(config-subscriber-control)#  bypass-traffic-acl DHCP
esr(config-subscriber-control)#  default-service
esr(config-subscriber-default-service)#  class-map INTERNET
esr(config-subscriber-default-service)#  filter-name local defaultservice
esr(config-subscriber-default-service)#  filter-action permit
esr(config-subscriber-default-service)#  default-action redirect http://192.0.2.20:8080/
eltex_portal/
esr(config-subscriber-default-service)#  session-timeout 3600
esr(config-subscriber-default-service)# exit
esr(config-subscriber-control)# enable
esr(config-subscriber-control)# exit
```

Configure rules for transition between security zones:

```
esr(config)# object-group service telnet
esr(config-object-group-service)#   port-range 23
esr(config-object-group-service)# exit
esr(config)# object-group service ssh
esr(config-object-group-service)#   port-range 22
esr(config-object-group-service)# exit
esr(config)# object-group service dhcp_server
esr(config-object-group-service)#   port-range 67
esr(config-object-group-service)# exit
esr(config)# object-group service dhcp_client
esr(config-object-group-service)#   port-range 68
esr(config-object-group-service)# exit
esr(config)# object-group service ntp
esr(config-object-group-service)#   port-range 123
esr(config-object-group-service)# exit
```

Enable access to the Internet from trusted and dmz zones:

```
esr(config)# security zone-pair trusted untrusted
esr(config-zone-pair)# rule 10
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol any
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
esr(config)# security zone-pair dmz untrusted
esr(config-zone-pair)# rule 10
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol any
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
esr(config)# security zone-pair dmz trusted
esr(config-zone-pair)# rule 10
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol any
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
```

Enable DHCP transmitting from trusted to dmz:

```
esr (config)# security zone-pair trusted dmz
esr (config-zone-pair)# rule 10
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol udp
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# match source-port dhcp_client
esr(config-zone-pair-rule)# match destination-port dhcp_server
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
```

Enable ICMP transmission to the device. For BRAS operation, open ports for web proxying – TCP 3129/3128 (NetPortDiscovery Port/Active API Server port:

```
esr(config)# object-group service bras
esr(config-object-group-service)#   port-range 3129
esr(config-object-group-service)#   port-range 3128
esr(config-object-group-service)# exit
esr(config)#  security zone-pair trusted self
esr(config-zone-pair)#  rule 10
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol tcp
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# match source-port any
esr(config-zone-pair-rule)# match destination-port bras
esr(config-zone-pair-rule)# enable
esr (config-zone-pair-rule)# exit
esr(config-zone-pair)# rule 20
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol icmp
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair-rule)# exit
esr(config)# security zone-pair dmz self
esr(config-zone-pair)# rule 20
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol icmp
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair-rule)# exit
esr(config)# security zone-pair untrusted self
esr(config-zone-pair)# rule 20
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol icmp
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair-rule)# exit
```

Activate DHCP-Relay:

```
esr(config)# ip dhcp-relay
```

Configure SNAT for gigabitethernet 1/0/1 port:

```
esr(config)# nat source
esr(config-snat)# ruleset inet
esr(config-snat-ruleset)# to interface gigabitethernet 1/0/1
esr(config-snat-ruleset)# rule 10
esr(config-snat-rule)# match source-address any
esr(config-snat-rule)# action source-nat interface
esr(config-snat-rule)# enable
esr(config-snat-rule)# end
```

18.3 Example of configuration without SoftWLC

Objective:

Configure BRAS without SoftWLC support.

Given:

Subnet with clients 10.10.0.0/16, subnet for working with FreeRADIUS server 192.168.1.1/24

Solution:

Step 1:

RADIUS server configuration.

For FreeRADIUS server, specify the subnet that can send the queries and add a user list. To do this, add the following to the users file in the directory with FreeRADIUS server configuration files:

User profile:

```
<MACADDR> Cleartext-Password := <MACADDR>
```

User name:

```
 User-Name = <USER_NAME>,
```

Maximum session lifetime:

```
 Session-Timeout = <SECONDS>,
```

Maximum session lifetime when the system is idle:

```
 Idle-Timeout = <SECONDS>,
```

Session statistics update time:

```
 Acct-Interim-Interval = <SECONDS>,
```

Service name for a session (A – the service is enabled, N – the service is disabled):

```
Cisco-Account-Info = "{A|N}<SERVICE_NAME>"
```

Service profile:

```
<SERVICE_NAME> Cleartext-Password := <MACADDR>
```

Matches class-map name in ESR settings:

```
Cisco-AVPair = "subscriber:traffic-class=<CLASS_MAP>",
```

Action that is applied to the traffic by ESR (permit, deny, redirect):

```
Cisco-AVPair = "subscriber:filter-default-action=<ACTION>",
```

The ability of IP flows passing (enabled-uplink, enabled-downlink, enabled, disabled):

```
Cisco-AVPair = "subscriber:flow-status=<STATUS>"
```

Add a subnet, in which ESR is located, to the clients.conf file:

```
client ESR {
ipaddr = <SUBNET>
secret = <RADIUS_KEY>
}
```

In this case the RADIUS server configuration will be as follows:

Add the following strings to the 'clients.conf' file:

```
client BRAS {
ipaddr = 192.168.1.1
secret = password
}
```

Add the following strings to the 'users' file (specify a client MAC address instead of <MAC>):

```
"54-E1-AD-8F-37-35" Cleartext-Password := "54-E1-AD-8F-37-35"
User-Name = "Bras_user",
Session-Timeout = 259200,
Idle-Timeout = 259200,
Cisco-AVPair += "subscriber:policer-rate-in=1000",
Cisco-AVPair += "subscriber:policer-rate-out=1000",
Cisco-AVPair += "subscriber:policer-burst-in=188",
Cisco-AVPair += "subscriber:policer-burst-out=188",
Cisco-Account-Info = "AINTERNET"
INTERNET Cleartext-Password := "INTERNET"
User-Name = "INTERNET",
Cisco-AVPair = "subscriber:traffic-class=INTERNET",
Cisco-AVPair += "subscriber:filter-default-action=permit"
```

Step 2:

ESR configuration.

BRAS functional configuration requires the BRAS licence:

```
esr(config)# do sh licence
Licence information
-------------------
Name:     Eltex
Version: 1.0
Type:     ESR-X
S/N:      NP00000000
MAC:      XX:XX:XX:XX:XX:XX
Features:
  BRAS - Broadband Remote Access Server
```

Configuration of parameters for the interaction with RADIUS server:

```
esr(config)# radius-server host 192.168.1.2
esr(config-radius-server)# key ascii-text encrypted 8CB5107EA7005AFF
esr(config-radius-server)# source-address 192.168.1.1
esr(config-radius-server)# exit
```

Create AAA profile:

```
esr(config)#  aaa radius-profile bras_radius
esr(config-aaa-radius-profile)# radius-server host 192.168.1.2
esr(config-aaa-radius-profile)#  exit
esr(config)#  aaa radius-profile bras_radius_servers
esr(config-aaa-radius-profile)#  radius-server host 192.168.1.2
esr(config-aaa-radius-profile)#  exit
```

Specify parameters for the DAS server:

```
esr(config)# das-server das
esr(config-das-server)# key ascii-text encrypted 8CB5107EA7005AFF
esr(config-das-server)#  exit
esr(config)#  aaa das-profile bras_das
esr(config-aaa-das-profile)#  das-server das
esr(config-aaa-das-profile)# exit
esr(config)#  vlan 10
esr(config-vlan)#  exit
```

Then, create rules for redirecting to portal and passing traffic to the Internet:

```
esr(config)#  ip access-list extended BYPASS
esr(config-acl)# rule 1
esr(config-acl-rule)#  action permit
esr(config-acl-rule)# match protocol udp
esr(config-acl-rule)#  match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)#  match source-port 68
esr(config-acl-rule)#  match destination-port 67
esr(config-acl-rule)# enable
esr(config-acl-rule)#  exit
esr(config-acl)#  rule 2
esr(config-acl-rule)#  action permit
esr(config-acl-rule)#  match protocol udp
esr(config-acl-rule)#  match source-address any
esr(config-acl-rule)#  match destination-address any
esr(config-acl-rule)#  match source-port any
esr(config-acl-rule)#  match destination-port 53
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config)# ip access-list extended INTERNET
esr(config-acl)# rule 1
esr(config-acl-rule)#  action permit
esr(config-acl-rule)# match protocol any
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# enable
esr(config-acl-rule)#  exit
esr(config)# ip access-list extended WELCOME
esr(config-acl)# rule 10
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol tcp
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# match source-port any
esr(config-acl-rule)# match destination-port 443
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# rule 20
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol tcp
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# match source-port any
esr(config-acl-rule)# match destination-port 8443
```

```
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# rule 30
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol tcp
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# match source-port any
esr(config-acl-rule)# match destination-port 80
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# rule 40
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol tcp
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# match source-port any
esr(config-acl-rule)# match destination-port 8080
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
```

Configuration of filtering by URL is obligatory. It is necessary to configure http-proxy filtering on BRAS for non-authorised users:

```
esr(config)# object-group url defaultserv
esr(config-object-group-url)# url http://eltex.nsk.ru
esr(config-object-group-url)# url http://ya.ru
esr(config-object-group-url)# url https://ya.ru
esr(config-object-group-url)# exit
```

Configure and enable BRAS, define NAS IP as address of the interface interacting with RADIUS server (gigabitethernet 1/0/2 in the example):

```
esr(config)#  subscriber-control
esr(config-subscriber-control)# aaa das-profile bras_das
esr(config-subscriber-control)# aaa sessions-radius-profile bras_radius
esr(config-subscriber-control)# aaa services-radius-profile bras_radius_servers
esr(config-subscriber-control)# nas-ip-address 192.168.1.1
esr(config-subscriber-control)# session mac-authentication
esr(config-subscriber-control)# bypass-traffic-acl BYPASS
esr(config-subscriber-control)# default-service
esr(config-subscriber-default-service)# class-map BYPASS
esr(config-subscriber-default-service)# filter-name local defaultserv
esr(config-subscriber-default-service)# filter-action permit
esr(config-subscriber-default-service)# default-action redirect http://192.
168.1.2:8080/eltex_portal
esr(config-subscriber-default-service)# session-timeout 121
esr(config-subscriber-default-service)# exit
esr(config-subscriber-control)# enable
esr(config-subscriber-control)# exit
```

Perform the following settings on the interfaces that require BRAS operation (minimum one interface is required for the successful start):

```
esr(config)#  bridge 10
esr(config-bridge)# vlan 10
esr(config-bridge)# ip firewall disable
esr(config-bridge)#  ip address 10.10.0.1/16
esr(config-bridge)#  ip helper-address 192.168.1.2
esr(config-bridge)# service-subscriber-control any
esr(config-bridge)# location USER
esr(config-bridge)#  protected-ports
esr(config-bridge)#  protected-ports exclude vlan
esr(config-bridge)# enable
esr(config-bridge)#  exit
```

Configure port towards the RADIUS server:

```
esr(config)# interface gigabitethernet 1/0/2
esr(config-if-gi)# ip firewall disable
esr(config-if-gi)# ip address 192.168.1.1/24
esr(config-if-gi)#  exit
```

Port towards the Client:

```
esr(config)#  interface gigabitethernet 1/0/3.10
esr(config-subif)#  bridge-group 10
esr(config-subif)# ip firewall disable
esr(config-subif)#  exit
```

Configure SNAT for gigabitethernet 1/0/2 port:

```
esr(config)# nat source
esr(config-snat)# ruleset factory
esr(config-snat-ruleset)# to interface gigabitethernet 1/0/2
esr(config-snat-ruleset)# rule 10
esr(config-snat-rule)# description "replace 'source ip' by outgoing interface ip address"
esr(config-snat-rule)# match protocol any
esr(config-snat-rule)# match source-address any
esr(config-snat-rule)# match destination-address any
esr(config-snat-rule)# action source-nat interface
esr(config-snat-rule)# enable
esr(config-snat-rule)# exit
esr(config-snat-ruleset)# exit
esr(config-snat)# exit
esr(config)# ip route 0.0.0.0/0 192.168.1.2
```

The configuration changes come into effect after applying the following commands:

```
esr(config) # do commit
esr(config) # do confirm
```

To view the information and statistics on the user control sessions, use the following command:

```
esr # sh subscriber-control sessions status

Session id        User name     IP address       MAC address        Interface          Domain
--------------------   ---------------   ---------------   -----------------
1729382256910270473    Bras_user   10.10.0.3  54:e1:ad:8f:37:35     gi1/0/3.10     --
```

19 VoIP management

VoIP (*Voice over IP*) – set of protocols that allow transmission of voice information over IP networks. Within the given device, VoIP is used to connect analogue telephones to an IP network with the possibility to make phone calls.

19.1 SIP profile configuration algorithm

Step	Description	Command	Keys
1	Configure SIP profile.	**esr(config)# sip profile <NUM>**	<NUM> – SIP profile number, set in the form of a digit from 1 to 5.
2	Configure primary SIP proxy server and registration server.	**esr(config-sip-profile)# proxy primary**	
3	Configure SIP proxy server.	**esr(config-voip-sip-proxy)# ip address proxy-server <IP>**	<IP> – proxy server IP address
4	Configure a SIP proxy server port.	**esr(config-voip-sip-proxy)# ip port proxy-server <PORT>**	<PORT> – number of proxy server UDP port, takes values of [1..65535]. If standard 5060 port is used, there is no need to specify it.
5	Configure a registration server address.	**esr(config-voip-sip-proxy)# ip address registration-server <IP>**	<IP> – registration server IP address.
6	Configure a registration server port.	**esr(config-voip-sip-proxy)# ip portregistration-server <PORT>**	<PORT> – number of registration server UDP port, takes values of [1..65535]. If standard 5060 port is used, there is no need to specify it.
7	Enable registration.	**esr(config-voip-sip-proxy)# registration**	
8	Enable proxy server and registration server.	**esr(config-voip-sip-proxy)# enable**	

Step	Description	Command	Keys
9	Configure a registration server address.	**esr(config-voip-sip-proxy)# ip address registration-server <IP>**	<IP> – registration server IP address.
10	Configure a registration server port.	**esr(config-voip-sip-proxy)# ip portregistration-server <PORT>**	<PORT> – number of registration server UDP port, takes values of [1..65535]. If standard 5060 port is used, there is no need to specify it.
11	Specify SIP domain in which the device is located.	**esr(config-sip-profile)# sip-domain address <ADDRESS>**	<ADDRESS> – SIP domain in which the device is located, set by ipv4 address or domain name.
12	Enable the use of SIP domain when registering.	**esr(config-sip-profile)# sip-domain registration enable**	
13	Configure a SIP profile.	**esr(config)# sip profile <NUM>**	<NUM> – SIP profile number, set in the form of a digit from 1 to 5.
14	Assign a dial plan to the current SIP profile.	**esr(config-sip-profile)# dialplan pattern <DNAME>**	<DNAME> – name of the dial plan, set by the string of up to 31 characters.
15	Enable SIP profile.	**esr(config-sip-profile)# enable**	

19.2 FXS/FXO ports configuration algorithm

Step	Description	Command	Keys
1	Switch to the FXO/FXS ports configuration mode.	**esr(config)# interface voice-port <NUM>**	<NUM> – port number, takes values of [1..4].
2	Assign a subscriber number reserved for a telephone port.	**esr(config-voice-port-fxs)# sip user phone <PHONE>**	<PHONE> – subscriber number reserved for a telephone port, set by the string of up to 50 characters.
3	Assign the user name matched with the port.	**esr-12v(config-voice-port-fxs)# sip user display-name <LOGIN>**	<LOGIN> – user name displayed in the Display-Name field, set by the string of up to 31 characters.
4	Select SIP profile for a certain port.	**esr(config-voice-port-fxs)# profile sip <PROFILE>**	<PROFILE> – SIP profile number, set in the form of a digit from 1 to 5.

Step	Description	Command	Keys
5	Configure a login for authentication.	**esr(config-voice-port-fxs)# authentication name <LOGIN>**	<LOGIN> – login for authentication, set by the string of up to 31 characters
6	Configure a password for authentication.	**esr(config-voice-port-fxs)# authentication password <PASS>**	<PASS> – authentication password, set by the string of up to 16 characters.
7	Enable FXO port.	**esr(config)# interface voice-port <NUM>**	<NUM> – FXO port number, takes values of [1..4].
8	Assign a subscriber number reserved for a telephone port.	**esr(config-voice-port-fxo)# sip user phone <PHONE>**	<PHONE> – subscriber number reserved for a telephone port.
9	Specify UDP port from which and to which the FXO set will send and receive SIP messages.	**esr(config-voice-port-fxo)# sip port <PORT>**	<PORT> – UDP port number.
10	Assign the user name matched with the port.	**esr(config-voice-port-fxo)# sip user display-name <LOGIN>**	<LOGIN> – user name displayed in the Display-Name field, set by the string of up to 31 characters.
11	Configure a login for authentication.	**esr(config-voice-port-fxo)# authentication name <LOGIN>**	<LOGIN> – login for authentication, set by the string of up to 31 characters.
12	Configure a password for authentication.	**esr(config-voice-port-fxo)# authentication password <PASS>**	<PASS> – authentication password, set by the string of up to 16 characters.
13	Enable the number transmission to PSTN.	**esr(config-voice-port-fxo)# pstn transmit-number**	
14	Disable prefix transmission.	**esr(config-voice-port-fxo)# no pstn transmit-prefix**	
15	Enable the 'Hostline PSTN to IP' service.	**esr(config-voice-port-fxo)# hotline ipt**	
16	Number of the subscriber that will receive calls from PSTN.	**esr(config-voice-port-fxo)# hotline number ipt <PHONE>**	<PHONE> – phone number that calls are made to when using the service, takes the value from 1 to 50. 'Hot/Warm line' in the direction from analogue telephone line to VoIP.

19.3 Dial plan configuration algorithm

Step	Description	Command	Keys
1	Create a dial plan.	**esr(config)# dialplan pattern <DNAME>**	<DNAME> – name of the dial plan, set by the string of up to 31 characters.
2	Add dial rules.	**esr(config-dial-ruleset)# pattern <REGEXP>**	<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.
3	Enable the dial plan.	**esr(config-dial-ruleset)# enable**	

19.4 PBX server configuration algorithm

Step	Description	Command	Keys
1	PBX server configuration.	**esr(config)# pbx**	
2	Enable PBX server.	**esr(config-pbx)# enable**	
3	Create a routing plan.	**esr(config-pbx)# ruleset <rule_name>**	<rule_name> – name of the routing plan, set by the string of up to 31 characters.
4	Create a routing rule.	**esr(config-pbx-ruleset)# rule <rule_index>**	<rule_index> – number of the rule in the routing plan, takes values from 1 to 1000.
5	Create a pattern in a routing rule.	**esr(config-pbx-rule)# pattern <REGEXP>**	<REGEXP> – regular expression specifying the routing rule. Set by the string of up to 256 characters. The rules for creating regular expressions are described in section Dial plan configuration example.
6	Apply a routing rule.	**esr(config-pbx-rule)# enable**	
7	Create a SIP profile on a PBX Server.	**esr(config-pbx)# profile <PROFILE>**	<PROFILE> – name of the SIP profile, that used by PBX server, set by the string of 31 character.

Step	Description	Command	Keys
8	Select a codec supported by a SIP profile.	**esr(config-pbx-profile)# codec allow { G711A(alaw) | G711U(ulaw) | G722 | G726 }**	
9	Select SIP profile type.	**esr(config-pbx-profile)# client { peer | user | friend }**	• **peer** – incoming and outgoing calls are allowed without authorization. • **user** – only incoming calls are allowed. • **friend** – combines peer and user profile types.
10	Choose a NAT interaction policy (optional).	**esr(config-pbx-profile)# nat { 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.
11	Select a SIP profile routing plan.	**esr(config-pbx-profile)# ruleset <NAME>**	<NAME> – name of the routing plan, set by the string of up to 31 characters.
12	Create a subscriber.	**esr(config-pbx)# user <user>**	<user> – phone number or username, set by the string of up to 31 characters.
13	Create a password for the subscriber (optional).	**esr(config-pbx-user)# password <password>**	<password> – password that will be used by the user for authentication, set by the string of up to 16 characters.
14	Use of SIP profile for the subscriber.	**esr(config-pbx-user)# profile <SIPPROFILE>**	<SIPPROFILE> – SIP profile used for this subscriber, set by the string of up to 31 characters.

19.5 Registration trunk creation algorithm

Step	Description	Command	Keys
1	Configure PBX server.	**esr(config)# pbx**	
2	Create trunk.	**esr(config-pbx)# register-server <name>**	<name> – trunk name, set by the string of up to 31 characters.
3	Configure registration server address.	**esr(config-pbx-reg-server)# ip address <IP>**	<IP> – address of the server on which registration proceeds, takes values of an IP address or can be specified by the string of up to 31 characters.

Step	Description	Command	Keys
4	Configure registration server port.	**esr(config-pbx-reg-server)# ip port <PORT>**	<PORT> – number of registration server UDP port, takes values of [1..65535]. If standard 5060 port is used, you do not need to specify it.
5	Specify the authentication name.	**esr(config-pbx-reg-server)# username <user>**	<user> – username for this trunk on the upstream domain, set by the string of up to 31 characters.
6	Specify the authentication password.	**esr(config-pbx-reg-server)# authentication password <password>**	<user> – password for this trunk on the upstream domain, set by the string of up to 16 characters.
7	Use of SIP profile for the trunk.	**esr(config-pbx-reg-server)# profile <PROFILE>**	<PROFILE> – name of the SIP profile, that used for this trunk, set by the string of 31 character.
8	Select the transport protocol (optional).	**esr(config-pbx-reg-server)# protocol {tcp | udp }**	The default is udp.
9	Enable trunk.	**esr(config-pbx-reg-server)# enable**	

19.6 VoIP configuration example

Objective:

Connect analogue telephones and fax modems to the IP network via ESR router. SIP server, located on the ESR, functions as proxy server and registration server.

Solution:

Configure a SIP profile:

```
esr(config)# sip profile 1
```

Configure a primary SIP proxy server and registration server:

```
esr(config-sip-profile)# proxy primary
```

Configure SIP proxy server address (use an embedded SIP server as SIP proxy server):

```
esr(config-voip-sip-proxy)# ip address proxy-server 192.0.2.5
```

Configure a SIP proxy server port:

```
esr(config-voip-sip-proxy)# ip port proxy-server 5080
```

If standard 5060 port is used, there is no need to specify it.

If it is necessary to use the registration, perform the following steps:

Configure registration server address (use an embedded SIP server as registration server):

```
esr(config-voip-sip-proxy)# ip address registration-server 192.0.2.5
```

Configure a registration server port:

```
esr(config-voip-sip-proxy)# ip port registration-server 5080
```

If standard 5060 port is used, you do not need to specify it.

Enable registration:

```
esr(config-voip-sip-proxy)# registration
```

Enable proxy server and registration server:

```
esr(config-voip-sip-proxy)# enable
```

This completes the configuration of SIP proxy server and registration server:

```
esr(config-voip-sip-proxy)# exit
```

The next step is to continue SIP profile configuration.

Configure a SIP domain:

```
esr(config-sip-profile)# sip-domain address sipdomain.com
```

If it is necessary to use SIP Domain for the registration, use the following command:

```
esr(config-sip-profile)# sip-domain registration enable
```

In this configuration all calls will be directed to SIP proxy server. If it is necessary to specify another direction for outgoing calls, you should perform the following:
Create a numbering plan, see section Dial plan configuration example.

Next, assign the created dial plan to the SIP profile:

```
esr(config)# sip profile 1
esr(config-sip-profile)# dialplan pattern firstDialplan
```

This completes the configuration of a dial plan for SIP profile.

Enable SIP profile:

```
esr-12v(config-sip-profile)# enable
```

This completes the baseline configuration of SIP profile:

```
esr(config-sip-profile)# exit
```

The next step is to configure subscriber ports:

```
esr(config)# interface voice-port 1
```

Specify a subscriber number:

```
esr(config-voice-port-fxs)# sip user phone 4101
```

Specify a displayed name:

```
esr(config-voice-port-fxs)# sip user display-name user-one
```

Used SIP profile:

```
esr(config-voice-port-fxs)# profile sip 1
```

Configure login and password for authentication

```
esr(config-voice-port-fxs)# authentication name login-4101
esr(config-voice-port-fxs)# authentication password superpassword
```

This completes the baseline configuration of a subscriber port:

```
esr(config-voice-port-fxs)# exit
```

19.7 Dial plan configuration example

Objective:

Configure a dial plan in such a manner that calls to local numbers (connected to the given ESR-12V) are switched locally and calls to all other directions – through SIP proxy.

Solution:

Create a dial plan:

```
esr(config)# dialplan pattern firstDialplan
```

Dial plan is specified by regular expressions:

```
esr(config-dial-ruleset)# pattern "<regular expressions>"
```

For the objective mentioned above, the '<*regular* expressions>' is given by:

'S5, L5 (410[1-3]@{local} | [xABCD*#].S)'

where:

- **410[1-3]@{local}** – calls to 4101, 4102, 4103 numbers will be switched locally;
- **[xABCD*#]. S** – calls to all other numbers will be directed to SIP proxy.

Enable the dial plan:

```
esr(config-dial-ruleset)# enable
```

Dial plan configuration is finished.

```
esr(config-dial-ruleset)# exit
```

Regular expression structure:

Sxx, Lxx ( ),

where:

- **xx** – random values of S and L timers;
- **()** – dialplan limits.

The basis is designators for dialled digits sequence to be written. Sequence of digits is written by several designators: digits dialled from a phone keyboard: 0, 1, 2, 3, ..., 9, # and *.

The use of # character in dial plan can block the completion of dialing with this key.

Bracketed sequence of digits corresponds to any bracketed character.

- Example: ([1239]) - corresponds to any of this digits: 1, 2, 3 and 9.
  You may specify the hyphenated range of characters. Usually it is used inside the square brackets.
- Example 1: (1-5) - any digit from 1 to 5.
- Example 2: ([1-39]) – example from previous paragraph with other record format.
  ‘X’ character corresponds to any digit from 0 to 9.
- Example: (1XX) - any three-digit number, starting at 1.
  '.' - Previous symbol repeating from 0 to infinity.
  '+' – repeating the previous character from 1 to infinity number of times.

{a,b} – repeating the previous character from a to b times;
{a,} – repeating the previous character equal to or more than a times;
{,b} – repeating the previous character equal to or less than b times.

- Example: (810X.) - international number with any digits amount.
  Settings influencing on the dial plan processing:
- Interdigit Long Timer (letter 'L' in dial plan entry) – timeout to enter the next digit if there are no templates matching the dialed combination;
- Interdigit Short Timer (letter 'S' in dial plan entry) – timeout to enter the next digit if at least one pattern completely matches the dialed combination and there is at least one more pattern before matching with that it is necessary to perform the extension dialing.

Additional features:

1. Replacement of a dialed sequence
   Syntax: *<arg1:arg2>*
   This feature allows to replace a dialed sequence to any sequence of dialed characters. In this case, the second argument must be specified with a certain value, both arguments may be empty.
   - Example: (<83812:> XXXXXX) — this record will comply to dialed digits 83812, but this sequence will omitted and will not be transmitted to SIP server.
2. Insert a tone in the set
   For long-distance access (for city access in case of office PBX), it is common to hear a ringback, that may be implemented by inserting comma in a sequence of digits.
   - Example: (8, 770) - after digit 8 a continuous tone will output when dialing number 8770.
3. Number dialing restriction
   If at the end of pattern add symbol '!' the dialing of numbers corresponding to the template will be blocked.
   - Example: (8 10X xxxxxxx ! | 8 xxx xxxxxxx) - expression allows dialing only intercity numbers and exclude international calls.
4. Replacement of number dialing timers values
   Timers values can be assigned both to a whole dial plan and to a certain template. 'S' is responsible for the '*Interdigit Short Timer*' setup and 'L' – for the '*Interdigit Long Timer*' Timers values can be specified for all templates in a dial plan if the values are listed before the opening parenthesis.
   - Example: S4 (8XXX.) or S4,L8 (XXX)
     If these values are listed in one sequence only, they are effective only for this sequence. Also, in this case it is not necessary to put a colon between the key and the timeout value, the value can be located anywhere in the template.
   - Example: (S4 8XXX. | XXX) или ([1-5] XX S0) – entry will call instant call transmission when three-digit number starting at 1, 2, ..., 5 is dialed.
5. Dialing via direct address (IP Dialing)
   '@' character put after the number means that the address of the server, to which the dialed number call will be sent, will be specified. We recommend to use *'IP Dialing'* and receive and transmission of call without registration *('Call Without Reg', 'Answer Without Reg'*). This can help in case of server failure.
   In addition, the format of address with IP Dialing can be used in numbers intended to forward calls.
   - Example 1: (8 xxx xxxxxxx) – 11-digit number, starting with 8.
   - Example 2: (8 xxx xxxxxxx | <:8495> xxxxxxx) – 11-digit number, starting with 8; if 7-digit number was entered, add 8495 to the number being transmitted.
   - Example 3: (0[123] | 8 [2-9]xx [2-9]xxxxxx) – emergency service numbers dialing as well as unusual dialing of long-distance call numbers.
   - Example 4: (S0 <:82125551234>) – specified number speed dial, «Hotline» mode analogue on another gateways.
   - Example 5: (S5 <:1000> | xxxx) – the given dial plan allows to dial any number consisting of digits; if nothing is entered during 5 seconds, call number 1000 (let it be a secretary).
   - Example 6: (8, 10x.|1xx@10.110.60.51:5060) – the given dial plan allows to dial numbers starting with 810 and containing at least one digit after '810'. After entering 8, the 'station response' signal

will be returned. Also a set of three-digit numbers starting with '1', the Invite of which will be sent to 10.110.60.51 IP address and 5060 port, will be returned.

- Example 7: (S3 *xx#|#xx#|#xx#|*xx*x+#) – management and the use of VAS.
  Local calls inside the device may be required in some cases. If the device's IP address is not known or is periodically changed, it is convenient to use the reserved word {local} as the server address, which means sending the corresponding sequence of digits to the device's own address.
- Example: (123@{local}) – call on number 123 will be locally processed within the device.

19.8 FXO port configuration

Objective:

Add the ability to make a call to PSTN subscriber through the ESR-12V FXO port.

Solution:

Enable FXO port:

```
esr(config)# interface voice-port 4
```

Specify FXO port number same as PSTN access prefix:

```
esr(config-voice-port-fxo)# sip user phone 9
```

Specify UDP port from which and to which the FXO set will send and receive SIP messages:

```
esr(config-voice-port-fxo)# sip port 5064
```

Specify a displayed name:

```
esr(config-voice-port-fxo)# sip user display-name user-one
```

Configure login and password for authentication:

```
esr(config-voice-port-fxo)# authentication name login-9
esr(config-voice-port-fxo)# authentication password superpassword
```

Assign SIP profile to FXO port:

```
esr(config-voice-port-fxo)# profile sip 1
```

Enable the number transmission to PSTN:

```
esr(config-voice-port-fxo)# pstn transmit-number
```

Disable prefix transmission:

```
esr(config-voice-port-fxo)# no pstn transmit-prefix
```

For outgoing calls to work, specify the following rule in the dial plan settings, which means that outgoing calls to numbers with prefix 9 are routed locally to the FXO set:

9x.@{local}:5064

This completes the baseline configuration of outgoing calls to PSTN. To make a call to PSTN, dial the callee number with the specified prefix (FXO set phone number).

To receive calls from PSTN, you should select the subscriber that will receive all calls from PSTN, let it be a subscriber with number 305.

Enable the 'Hostline PSTN to IP' service:

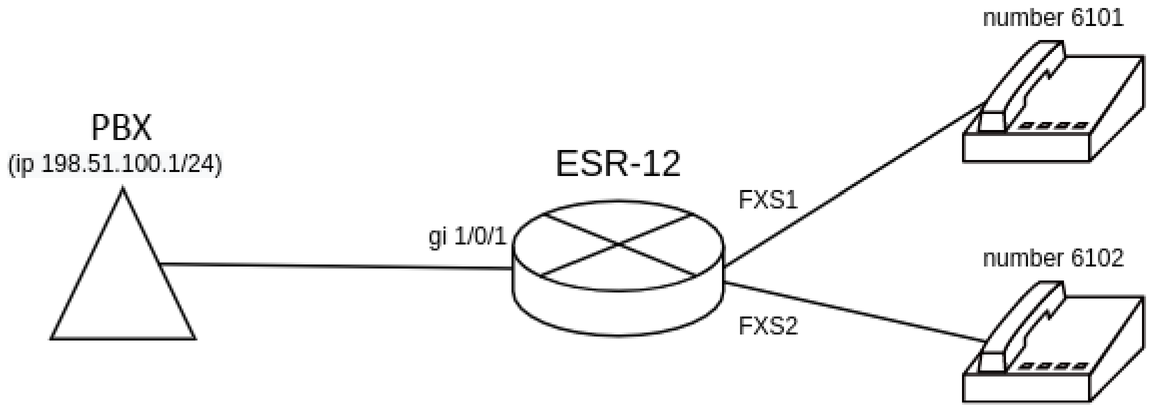
```
esr(config-voice-port-fxo)# hotline ipt
```

Number of the subscriber that will receive calls from PSTN:

```
esr(config-voice-port-fxo)# hotline number ipt 305
```

19.9 Example of VoIP configuration for FXS ports registration on external SIP server

Diagram:



Objective:

Configure VoIP for FXS ports registration on external SIP server.

Solution:

Configure SIP profile. It is necessary to configure proxy-server for registration of the phones connected to FXS ports:

```
esr(config)# sip profile 1
esr(config-sip-profile)# enable
esr(config-sip-profile)# sip-domain address 198.51.100.1
esr(config-sip-profile)# proxy primary
esr(config-voip-sip-proxy)# enable
esr(config-voip-sip-proxy)# ip address proxy-server 198.51.100.1
esr(config-voip-sip-proxy)# registration
esr(config-voip-sip-proxy)# ip address registration-server 198.51.100.1
esr(config-voip-sip-proxy)# exit
esr(config-sip-profile)# exit
esr(config)#
```

Configure FXS ports. Specify number, parameters for authentication on an external server and sip profile:

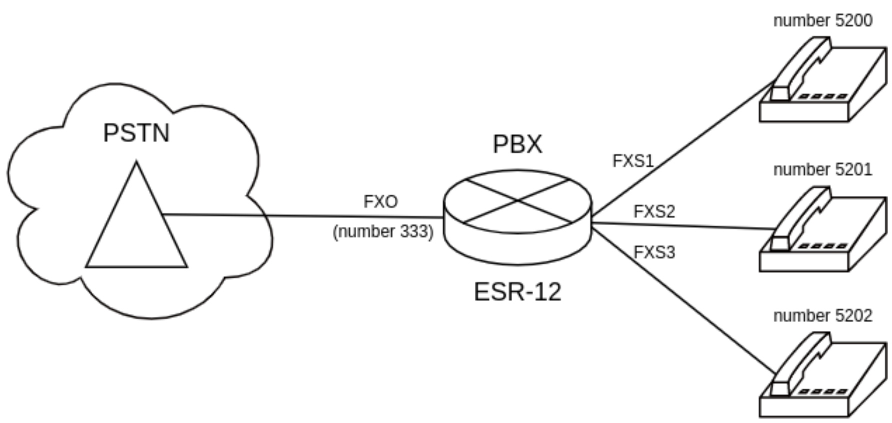
```
esr(config)# interface voice-port 1
esr(config-voice-port-fxs)# sip user phone 6101
esr(config-voice-port-fxs)# authentication name as-phone
esr(config-voice-port-fxs)# authentication password password
esr(config-voice-port-fxs)# profile sip 1
esr(config-voice-port-fxs)# exit
esr(config)# interface voice-port 2
esr(config-voice-port-fxs)# sip user phone 6102
esr(config-voice-port-fxs)# authentication name as-phone
esr(config-voice-port-fxs)# authentication password password
esr(config-voice-port-fxs)# profile sip 1
esr(config-voice-port-fxs)# exit
esr(config)#
```

To register and pass VoIP traffic over NAT, enable tracking and NAT for SIP on the router, and enable service-voip routing on the interface through which the SIP server is available:

```
esr(config)# ip firewall sessions tracking sip
esr(config)# nat alg sip
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# service-voip routing
esr(config-if-gi)# exit
esr(config)#
```

19.10 Example of VoIP configuration on internal PBX server

Diagram:



Objective:

Configure registration of phones connected to the FXS port, and configure number registration of the phone line connected to the FXO port. It is also necessary to configure call transfer from FXO port to phone with number 5200.

Solution:

Configure PBX.

Create PBX server:

```
esr(config)# pbx
esr(config-pbx)#
```

The structure of the 'pattern' regular expression is described in the section Dial plan configuration example.

Configure routing context for FXO port (example of dial plan for call transfer to 5200-5202):

```
esr(config-pbx)# ruleset FXO
esr(config-pbx-ruleset)# rule 1
esr(config-pbx-rule)# pattern '_X.,1,Dial(SIP/5200&SIP/5201&SIP/5202)'
esr(config-pbx-rule)# enable
esr(config-pbx-rule)# exit
esr(config-pbx-ruleset)# exit
esr(config-pbx)#
```

Configure routing context for FXS ports (example of dial plan for call transfer to 5200-5202):

```
esr(config-pbx)# ruleset main_rule
esr(config-pbx-ruleset)# rule 1
esr(config-pbx-rule)# pattern '_520[0-3],1,Dial(SIP/${EXTEN},,t)'
esr(config-pbx-rule)# enable
esr(config-pbx-rule)# exit
esr(config-pbx-ruleset)# exit
esr(config-pbx)#
```

Configure SIP profile for FXS ports. Specify client type, enable necessary codecs and routing context:

```
esr(config-pbx)# profile fxs_ports
esr(config-pbx-profile)# client friend
esr(config-pbx-profile)# codec allow g711a
esr(config-pbx-profile)# codec allow g711u
esr(config-pbx-profile)# codec allow g729
esr(config-pbx-profile)# ruleset main_rule
esr(config-pbx-profile)# exit
esr(config-pbx)#
```

Configure SIP profile for FXO ports. Specify client type, enable necessary codecs, configure invite-port, enable routing context:

```
esr(config-pbx)# profile fxo_ports
esr(config-pbx-profile)# client friend
esr(config-pbx-profile)# codec allow g711a
esr(config-pbx-profile)# codec allow g711u
esr(config-pbx-profile)# security level invite-port
esr(config-pbx-profile)# ruleset FXO
esr(config-pbx-profile)# exit
esr(config-pbx)#
```

Configure subscribers on PBX server:

```
esr(config-pbx)# user 5200
esr(config-pbx-user)# profile fxs_ports
esr(config-pbx-user)# exit
esr(config-pbx)# user 5201
esr(config-pbx-user)# profile fxs_ports
esr(config-pbx-user)# exit
esr(config-pbx)# user 5202
esr(config-pbx-user)# profile fxs_ports
esr(config-pbx-user)# exit
esr(config-pbx)# user 333
esr(config-pbx-user)# profile fxo_ports
esr(config-pbx-user)# exit
esr(config-pbx)#
```

Enable PBX server:

```
esr(config-pbx)# enable
esr(config-pbx)# exit
esr(config)#
```

Configure FXS ports. Specify number and SIP profile:

```
esr(config)# interface voice-port 1
esr(config-voice-port-fxs)# sip user phone 5200
esr(config-voice-port-fxs)# profile pbx fxs_ports
esr(config-voice-port-fxs)# exit
esr(config)# interface voice-port 2
esr(config-voice-port-fxs)# sip user phone 5201
esr(config-voice-port-fxs)# profile pbx fxs_ports
esr(config-voice-port-fxs)# exit
esr(config)# interface voice-port 3
esr(config-voice-port-fxs)# sip user phone 5202
esr(config-voice-port-fxs)# profile pbx fxs_ports
esr(config-voice-port-fxs)# exit
esr(config)#
```

Configure FXO port. Specify number and SIP profile, disable caller-id, enable 'Hostline PSTN to IP' service. Then specify number of subscriber that will receive calls from PSTN:

```
esr(config)# interface voice-port 4
esr(config-voice-port-fxo)# sip user phone 333
esr(config-voice-port-fxo)# profile pbx fxo_ports
esr(config-voice-port-fxo)# caller-id mode off
esr(config-voice-port-fxo)# hotline ipt
esr(config-voice-port-fxo)# hotline number ipt 5200
esr(config-voice-port-fxo)# exit
esr(config)#
```

20 Frequently asked questions

Receiving of routes, which are configured in VRF via BGP or/and OSPF, failed. The neighbor adjacency is successfully established, but record of routes in RIB is denied

%ROUTING-W-KERNEL: Can not install route. Reached the maximum number of BGP routes in the RIB

Allocate RIB resource for VRF (0 by default). Do it in VRF configuration mode:

```
esr(config)# ip vrf <NAME>
esr(config-vrf)# ip protocols ospf max-routes 12000
esr(config-vrf)# ip protocols bgp max-routes 1200000
esr(config-vrf)# end
```

SSH/Telnet sessions, which go through ESR router, are closing.

Configure transmission of keepalive packets in order to keep session active. Keepalive transmission option is configured on SSH client, for instance, section 'Connection' for PuTTY client.

It is possible to set timeout before closing inactive TCP sessions (1 hour in example):

```
esr(config)# ip firewall sessions tcp-estabilished-timeout 3600
```

Firewall was disabled on interface (ip firewall disable). However access for active sessions from the port was not closed, according to security zone-pair rules, after including this interface to security zone, removing from 'ip firewall disable' configuration and applying changes.

Changes in Firewall configuration will be active only for new sessions. The reset of Firewall active sessions does not occur. You can clear active sessions in firewall, using following command:

```
esr# clear ip firewall session
```

LACP does not launch on XG ports of ESR-1000/1200/1500/1700

Port-channel has speed 1000M mode by default. Enable speed 10G mode:

```
esr(config)# interface port-channel 1
esr(config-port-channel)# speed 10G
```

How to clear ESR configuration completely and reset it to factory default?

Copy blank configuration in candidate-config and apply it in running-config.

```
esr# copy system:default-config system:candidate-config
```

Reset to factory default is similar.

```
esr# copy system:factory-config system:candidate-config
```

How to attach sub-interface to created VLAN?
While sub-interface creation, VLAN is created and attached automatically (direct correlation index sub – VID).

```
esr(config)# interface gigabitethernet 1/0/1.100
```

Information messages are shown after applying:

```
2016-07-14T012:46:24+00:00 %VLAN: creating VLAN 100
```

Do the ESR series routers have features for traffic analysis?

Opportunity of analyzing traffic through CLI interfaces is realized on ESR-series routers. A packet sniffer is launched by monitor command.

```
esr# monitor gigabitethernet 1/0/1
```

How to configure ip-prefix-list 0.0.0.0./0?

Example of prefix-list configuration is shown below. The configuration allows route reception by default.

```
esr(config)# ip prefix-list eltex
esr(config-pl)# permit default-route
```

Problem of asynchronous traffic transmission is occurred

In case of asynchronous routing, Firewall will forbid 'incorrect' ingress traffic (which does not open new connection and does not belong any established connection) for security reasons.

Allowing rule in Firewall does not solve the problem.

Firewall should be disabled on the ingress interface.

```
esr(config-if-gi)# ip firewall disable
```

How to save the local copy of the router configuration?

If you need to copy the current running or candidate configuration on the router itself, you can use the copy command specifying 'system:running-config' or 'system:candidate-config' as the copy source, and the file in the 'flash:data/' section as the copy destination.

```
esr# copy system:candidate-config flash:data/temp.txt
```

Also, it is possible to copy previously saved configuration files (automatically from the flash:backup/ section or manually from the flash:data/ section) to the candidate configuration:

```
esr# copy flash:data/temp.txt system:candidate-config
esr# copy flash:backup/config_20190918_164455 system:candidate-config
```

TECHNICAL SUPPORT

For technical assistance in issues related to handling Eltex Ltd. equipment, please, address to Service Center of the company:

http://www.eltex-co.com/support

You are welcome to visit Eltex official website to get the relevant technical documentation and software, to use our knowledge base or consult a Service Center Specialist in our technical forum.

http://www.eltex-co.com/

http://www.eltex-co.com/support/downloads/