

Маршрутизатор серии ESR

ESR-10

Справочник команд CLI

Версия ПО 1.0.8

Версия документа	Дата выпуска	Содержание изменений
Версия 1.0	02.12.2016	Первая публикация
Версия программного обеспечения	1.0.8	

СОДЕРЖАНИЕ

1	ВВЕДЕНИЕ.....	19
1.1	Аннотация	19
1.2	Целевая аудитория	19
1.3	Условные обозначения.....	19
1.4	Используемые сокращения.....	20
2	ПРАВИЛА ПОЛЬЗОВАНИЯ КОМАНДНОЙ СТРОКОЙ.....	21
3	СТРУКТУРА СИСТЕМЫ КОМАНД	24
3.1	Глобальный режим	24
3.2	Конфигурирование маршрутизатора	25
3.3	Типы и порядок именования интерфейсов маршрутизатора	30
3.4	Типы и порядок именования туннелей маршрутизатора	32
4	КОМАНДЫ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА	33
4.1	exit	33
4.2	end	33
4.3	help	33
4.4	show history.....	34
4.5	do	34
4.6	logout.....	35
4.7	reload system.....	35
4.8	terminal datadump	35
4.9	uptime.....	36
4.10	ping	36
4.11	traceroute	38
4.12	ssh	39
4.13	telnet.....	40
4.14	monitor	41
4.15	clear ssh host	42
5	УПРАВЛЕНИЕ ПРОГРАММНЫМ ОБЕСПЕЧЕНИЕМ И КОНФИГУРАЦИЕЙ	44
5.1	commit	44
5.2	commit update.....	44
5.3	confirm.....	45
5.4	restore.....	45
5.5	rollback.....	46
5.6	save	46
5.7	boot system	46
5.8	delete	47
5.9	show candidate-config.....	48
5.10	show running-config	50
5.11	show version	51
5.12	show bootvar	51
5.13	show licence.....	52
5.14	show crypto certificates.....	52
5.15	copy.....	53
5.16	archive.....	56
5.17	path.....	57
5.18	by-commit	57
5.19	auto	58
5.20	time-period	58
6	НАСТРОЙКА ОБЩЕСИСТЕМНЫХ ПАРАМЕТРОВ	60
6.1	hostname	60
6.2	system fan-speed	60

6.3	show system.....	60
6.4	show system id.....	61
6.5	show cpu network-load.....	62
6.6	show cpu utilization	62
7	УПРАВЛЕНИЕ СИСТЕМНЫМИ ЧАСАМИ	64
7.1	set date.....	64
7.2	clock timezone	64
7.3	show date.....	65
7.4	ntp enable	65
7.5	ntp broadcast-client enable	66
7.6	ntp dscp.....	66
7.7	ntp peer.....	67
7.8	ntp server	67
7.9	maxpoll.....	68
7.10	minpoll	69
7.11	prefer.....	69
7.12	version.....	70
7.13	show ntp configuration	70
7.14	show ntp peers.....	71
8	НАСТРОЙКА AAA	73
8.1	username	73
8.2	password.....	73
8.3	privilege.....	74
8.4	enable.....	75
8.5	disable	75
8.6	enable password	76
8.7	aaa authentication login	76
8.8	aaa authentication enable	77
8.9	aaa accounting commands	78
8.10	aaa accounting login	78
8.11	line.....	79
8.12	login authentication.....	79
8.13	enable authentication	80
8.14	tacacs-server timeout	81
8.15	tacacs-server dscp.....	81
8.16	tacacs-server host	82
8.17	radius-server timeout	82
8.18	radius-server retransmit	83
8.19	radius-server dscp	83
8.20	radius-server host	84
8.21	ldap-server base-dn	84
8.22	ldap-server bind timeout	85
8.23	ldap-server bind authenticate root-dn	85
8.24	ldap-server bind authenticate root-password	86
8.25	ldap-server search filter user-object-class	86
8.26	ldap-server search scope	87
8.27	ldap-server search timeout	87
8.28	ldap-server naming-attribute.....	88
8.29	ldap-server privilege-level-attribute	88
8.30	ldap-server dscp.....	89
8.31	ldap-server host	89
8.32	usage	90
8.33	key	90
8.34	port.....	91

8.35	priority	92
8.36	timeout	92
8.37	source-address.....	93
8.38	privilege.....	93
8.39	tech-support login enable privilege	94
8.40	show users accounts	94
8.41	show users	95
8.42	show aaa authentication	95
8.43	show aaa accounting	96
8.44	show aaa ldap-servers	96
8.45	show aaa radius-servers	97
8.46	show aaa tacacs-servers	97
9	НАСТРОЙКА И МОНИТОРИНГ ИНТЕРФЕЙСОВ.....	99
9.1	Ethernet-интерфейсы	99
9.1.1	interface.....	99
9.1.2	description	100
9.1.3	load-average	100
9.1.4	speed.....	101
9.1.5	system jumbo-frames	102
9.1.6	mtu.....	102
9.1.7	shutdown	103
9.1.8	switchport mode.....	104
9.1.9	show interfaces switch-port status.....	104
9.1.10	show interfaces utilization.....	105
9.1.11	show interfaces counters.....	106
9.1.12	show interfaces description.....	107
9.1.13	show interfaces status	107
9.1.14	show interfaces sfp	108
9.1.15	clear interfaces counters	109
9.2	TDM (E1)	109
9.2.1	ppp authentication chap.....	109
9.2.2	ppp chap refuse	110
9.2.3	ppp chap hostname	110
9.2.4	ppp chap password.....	111
9.2.5	ppp chap username	111
9.2.6	password.....	112
9.2.7	ppp ipcp accept-address.....	112
9.2.8	ppp ipcp remote-address.....	113
9.2.9	ppp max-configure.....	113
9.2.10	ppp max-failure.....	114
9.2.11	ppp max-terminate	114
9.2.12	ppp mru	115
9.2.13	ppp timeout keepalive	115
9.2.14	ppp timeout retry	116
9.2.15	switchport e1 slot	116
10	УПРАВЛЕНИЕ ГРУППАМИ АГРЕГАЦИИ КАНАЛОВ	118
10.1	Link Agregation Group (LAG)	118
10.1.1	channel-group.....	118
10.1.2	lACP system-priority	118
10.1.3	port-channel load-balance.....	119
10.1.4	lACP timeout.....	119
10.1.5	lACP port-priority.....	120
10.1.6	show lACP statistics	120
10.1.7	show lACP parameters.....	121

10.1.8	show lacp	121
10.1.9	show interfaces port-channel	122
10.1.10	show lacp port-channel	123
10.2	MLPPP – Multilink PPP	123
10.2.1	ppp multilink	123
10.2.2	ppp multilink-group	124
11	УПРАВЛЕНИЕ ТУННЕЛЯМИ.....	125
11.1	tunnel	125
11.2	enable.....	126
11.3	description	126
11.4	local address.....	127
11.5	remote address	128
11.6	mtu	128
11.7	ttl	129
11.8	dscp	129
11.9	key	130
11.10	local checksum	130
11.11	remote checksum.....	131
11.12	protocol.....	131
11.13	local cookie.....	132
11.14	local port	132
11.15	local session-id	133
11.16	local tunnel-id	133
11.17	remote cookie	133
11.18	remote port.....	134
11.19	remote session-id.....	134
11.20	remote tunnel-id	135
11.21	mode	135
11.22	default-profile	136
11.23	show tunnels configuration	136
11.24	show tunnels status	137
11.25	show tunnels counters.....	137
11.26	show tunnels utilization	138
11.27	clear tunnels counters.....	139
12	УПРАВЛЕНИЕ L2 ФУНКЦИЯМИ.....	140
12.1	Управление L2 маршрутизацией	140
12.1.1	bridge	140
12.1.2	enable.....	140
12.1.3	description	141
12.1.4	vlan.....	141
12.1.5	bridge-group	142
12.1.6	mac-address	142
12.1.7	show interfaces bridge.....	143
12.2	Настройка и мониторинг VLAN.....	143
12.2.1	vlan.....	143
12.2.2	name	144
12.2.3	switchport default-vlan tagged.....	144
12.2.4	switchport forbidden default-vlan	145
12.2.5	switchport access vlan	145
12.2.6	switchport trunk allowed vlan	145
12.2.7	switchport trunk native vlan	146
12.2.8	show vlans.....	146
13	РАБОТА С АДРЕСНЫМИ ТАБЛИЦАМИ.....	148
13.1	ip arp reachable-time.....	148
13.2	ipv6 nd reachable-time	148

13.3	mac address-table aging time	149
13.4	mac address-table save-secure-freq	150
13.5	show mac address-table	150
13.6	clear mac address-table	151
13.7	show arp	151
13.8	clear arp-cache	152
13.9	show ipv6 neighbors	152
13.10	clear ipv6 neighbors	153
13.11	show arp configuration	154
13.12	show ipv6 neighbors configuration	154
14	НАСТРОЙКА VRF	155
14.1	ip vrf	155
14.2	description	155
14.3	ip vrf forwarding	155
14.4	ip source-vrf	156
14.5	show ip vrf	157
15	НАСТРОЙКА IP АДРЕСАЦИИ	158
15.1	ip address	158
15.2	show ip interfaces	159
16	НАСТРОЙКА IPV6 АДРЕСАЦИИ	161
16.1	ipv6 address	161
16.2	show ipv6 interfaces	162
17	УПРАВЛЕНИЕ ПРОФИЛЯМИ IP-АДРЕСОВ И ПОРТОВ	163
17.1	object-group network	163
17.2	object-group service	163
17.3	description	164
17.4	ip prefix	164
17.5	ip address-range	165
17.6	ipv6 prefix	165
17.7	ipv6 address-range	165
17.8	port-range	166
17.9	show object-group	166
18	УПРАВЛЕНИЕ СПИСКАМИ КОНТРОЛЯ ДОСТУПА (ACL)	168
18.1	ip access-list extended	168
18.2	description	168
18.3	service-acl input	169
18.4	rule	169
18.5	enable	169
18.6	match source-address	170
18.7	match source-mac	170
18.8	match source-port	171
18.9	match destination-address	171
18.10	match destination-mac	172
18.11	match destination-port	172
18.12	match protocol	173
18.13	match cos	173
18.14	match dscp	174
18.15	match ip-precedence	174
18.16	match vlan	175
18.17	action	175
18.18	show ip access-list	176
19	УПРАВЛЕНИЕ FIREWALL	178
19.1	security zone	178
19.2	description	178
19.3	security-zone	179

19.4	ip firewall disable	179
19.5	security zone-pair.....	180
19.6	rearrange.....	180
19.7	renumber	181
19.8	rule	181
19.9	description	182
19.10	enable.....	182
19.11	match source-address	183
19.12	match source-mac.....	183
19.13	match source-port.....	184
19.14	match destination-address	184
19.15	match destination-mac	185
19.16	match destination-port	185
19.17	match protocol.....	186
19.18	match icmp.....	186
19.19	match destination-nat.....	187
19.20	action.....	187
19.21	ip firewall sessions counters	188
19.22	ip firewall sessions max-expect.....	188
19.23	ip firewall sessions max-tracking	189
19.24	ip firewall sessions icmp-timeout	189
19.25	ip firewall sessions icmpv6-timeout.....	190
19.26	ip firewall sessions tcp-connect-timeout	190
19.27	ip firewall sessions tcp-disconnect-timeout	191
19.28	ip firewall sessions tcp-established-timeout.....	191
19.29	ip firewall sessions tcp-latecome-timeout.....	192
19.30	ip firewall sessions udp-assured-timeout	192
19.31	ip firewall sessions udp-wait-timeout.....	193
19.32	ip firewall sessions generic-timeout	194
19.33	show security zone.....	194
19.34	show security zone-pair	195
19.35	show security zone-pair configuration.....	195
19.36	show ip firewall counters	196
19.37	show ip firewall sessions	196
19.38	show ipv6 firewall counters	197
19.39	show ipv6 firewall sessions	197
19.40	clear ip firewall counters.....	198
19.41	clear ip firewall sessions.....	199
19.42	clear ipv6 firewall counters.....	199
19.43	clear ipv6 firewall sessions.....	200
20	УПРАВЛЕНИЕ NAT	201
20.1	nat source.....	201
20.2	nat destination	201
20.3	ruleset	202
20.4	pool	202
20.5	ip nat proxy-arp.....	203
20.6	from.....	203
20.7	to	204
20.8	description	204
20.9	rule	205
20.10	enable.....	205
20.11	match source-address	206
20.12	match source-port.....	206
20.13	match destination-address	207

20.14 match destination-port.....	208
20.15 match protocol.....	208
20.16 match icmp	209
20.17 action source-nat	209
20.18 action destination-nat.....	210
20.19 ip address	211
20.20 ip port.....	211
20.21 ip address-range	212
20.22 ip port-range	212
20.23 persistent	213
20.24 show ip nat ruleset	213
20.25 show ip nat pool.....	214
20.26 show nat proxy-arp	215
20.27 show ip nat translations.....	215
21 НАСТРОЙКИ IPSEC VPN	217
21.1 Управление VPN. Настройки IKE	217
21.1.1security ike proposal.....	217
21.1.2security ike policy.....	217
21.1.3security ike gateway	218
21.1.4description	218
21.1.5authentication method.....	219
21.1.6encryption algorithm	219
21.1.7dh-group	220
21.1.8authentication algorithm.....	220
21.1.9lifetime.....	221
21.1.10 mode.....	221
21.1.11 pre-shared-key	222
21.1.12 proposal.....	223
21.1.13 version	223
21.1.14 ike-policy.....	224
21.1.15 dead-peer-detection action.....	224
21.1.16 dead-peer-detection interval	225
21.1.17 dead-peer-detection timeout.....	225
21.1.18 mode.....	226
21.1.19 bind-interface vti	226
21.1.20 show security ike	227
21.2 Управление VPN. Настройки IPsec	228
21.2.1security ipsec proposal	228
21.2.2security ipsec policy	228
21.2.3security ipsec vpn.....	229
21.2.4description	229
21.2.5ike ipsec-policy.....	230
21.2.6ike gateway	230
21.2.7ike establish-tunnel.....	231
21.2.8ike dscp	231
21.2.9mode.....	232
21.2.10 manual mode.....	232
21.2.11 enable	233
21.2.12 manual bind-interface vti	233
21.2.13 manual protocol	234
21.2.14 manual spi	235
21.2.15 manual authentication algorithm.....	235
21.2.16 manual authentication key.....	236
21.2.17 manual encryption algorithm.....	236

21.2.18	manual encryption key	237
21.2.19	protocol	237
21.2.20	authentication algorithm	238
21.2.21	encryption algorithm	239
21.2.22	lifetime	239
21.2.23	proposal	240
21.2.24	show security ipsec	240
21.2.25	show security ipsec vpn status	241
22	УПРАВЛЕНИЕ VPN. НАСТРОЙКИ УДАЛЕННОГО ДОСТУПА	242
22.1	Общие команды настройки удаленного доступа	242
22.1.1	description	242
22.1.1	security-zone	242
22.1.2	show remote-access configuration	243
22.1.3	show remote-access counters	243
22.1.4	show remote-access status	244
22.1.5	clear remote-access session	245
22.1.6	clear remote-access counters	246
22.2	Настройка L2TP over IPsec/PPTP	246
22.2.1	remote-access pptp	246
22.2.2	remote-access l2tp	247
22.2.3	authentication mode	247
22.2.4	dns-servers	248
22.2.5	dscp	248
22.2.6	enable	249
22.2.7	ipsec authentication method	249
22.2.8	ipsec authentication pre-shared-key	250
22.2.9	local-address	250
22.2.10	mtu	251
22.2.11	outside-address	251
22.2.12	remote-address	252
22.2.13	username	253
22.2.14	password	253
22.2.1	wins-servers	254
22.2.1	dns-servers	254
22.3	Настройка OpenVPN	255
22.3.1	remote-access openvpn	255
22.3.1	enable	255
22.3.1	address-range	256
22.3.1	encryption algorithm	256
22.3.2	network	257
22.3.3	route	257
22.3.4	port	258
22.3.1	compression	258
22.3.2	client-isolation	259
22.3.3	username	259
22.3.4	client-max	260
22.3.1	certificate	260
22.3.2	protocol	261
22.3.3	tunnel	261
22.3.1	timers keepalive	262
22.3.1	timers holdtime	262
22.3.2	redirect-gateway	263
22.3.3	bridge-group	263
22.3.1	wins-servers	264

22.3.2	dns-servers.....	264
23	МАРШРУТИЗАЦИЯ	266
23.1	Общие настройки маршрутизации	266
23.1.1	ip protocols preference	266
23.1.1	ip protocols max-routes.....	266
23.1.2	ipv6 protocols max-routes.....	267
23.1.3	router log-adjacency-changes	268
23.1.4	ipv6 router log-adjacency-changes.....	268
23.1.5	ip path-mtu-discovery.....	269
23.1.6	ip tcp adjust-mss	269
23.1.7	ipv6 tcp adjust-mss	270
23.1.8	show ip route	270
23.1.9	show ipv6 route	271
23.2	Общие команды анонсирования и приема маршрутов	272
23.2.1	ip prefix-list	272
23.2.2	ipv6 prefix-list	273
23.2.3	permit/deny.....	273
23.2.4	redistribute static.....	274
23.2.5	redistribute connected	274
23.2.6	redistribute rip.....	275
23.2.7	redistribute ospf	275
23.2.8	redistribute bgp	276
23.2.9	network.....	277
23.2.10	prefix-list.....	277
23.3	Маршрутизация на основе политик (PBR).....	278
23.3.1	route-map	278
23.3.2	rule	278
23.3.3	description	279
23.3.4	action	279
23.3.5	match as-path	280
23.3.6	match community	280
23.3.7	match extcommunity.....	281
23.3.8	match ip access-group	281
23.3.9	match ip address.....	282
23.3.10	match ip next-hop	282
23.3.11	match ip route-source	283
23.3.12	match ipv6 address.....	283
23.3.13	match ipv6 next-hop.....	284
23.3.14	match ipv6 route-source	284
23.3.15	match metric bgp	285
23.3.16	match metric ospf.....	285
23.3.17	match metric rip	285
23.3.18	match tag ospf.....	286
23.3.19	match tag rip	286
23.3.20	action set as-path prepend.....	287
23.3.21	action set community	287
23.3.22	action set extcommunity	288
23.3.23	action set ip bgp-next-hop	288
23.3.24	action set ip next-hop.....	289
23.3.25	action set ipv6 bgp-next-hop.....	290
23.3.26	action set ipv6 next-hop.....	290
23.3.27	action set ip next-hop verify-availability	291
23.3.28	action set local-preference.....	291
23.3.29	action set origin	292

23.3.30	action set metric bgp	292
23.3.31	action set metric ospf	292
23.3.32	action set metric rip.....	293
23.3.33	action set tag ospf.....	293
23.3.34	action set tag rip	294
23.3.35	route-map	294
23.3.36	ip policy route-map.....	295
23.3.37	show ip route-map	295
23.4	Настройка связей ключей	297
23.4.1	key-chain	297
23.4.2	key	297
23.4.3	key-string.....	297
23.4.4	send-lifetime	298
23.4.5	accept-time	299
23.5	Настройка статических маршрутов IPv4/IPv6.....	300
23.5.1	ip route.....	300
23.5.2	ipv6 route	301
23.6	Настройка протокола BGP.....	303
23.6.1	router bgp	303
23.6.1	router bgp maximum-paths	303
23.6.2	address-family	304
23.6.3	router-id	304
23.6.4	timers keepalive	305
23.6.5	timers holdtime.....	305
23.6.6	neighbor	306
23.6.7	remote-as	306
23.6.8	ebgp-multihop.....	307
23.6.9	next-hop-self	307
23.6.10	remove-private-as	307
23.6.11	allow-local-as	308
23.6.12	default-originate	308
23.6.13	cluster-id	309
23.6.14	route-reflector-client	309
23.6.15	preference	310
23.6.16	authentication algorithm	310
23.6.17	authentication key	311
23.6.18	enable	311
23.6.19	router bgp log-neighbor-changes	312
23.6.20	show ip bgp.....	312
23.6.21	show ipv6 bgp	313
23.6.22	show ip bgp neighbors.....	314
23.6.23	show ipv6 bgp neighbors	315
23.6.24	clear ip bgp	316
23.6.25	clear ipv6 bgp.....	317
23.7	Настройка протокола RIP	317
23.7.1	router rip.....	317
23.7.2	enable.....	318
23.7.3	preference.....	318
23.7.4	authentication algorithm	319
23.7.5	authentication key	319
23.7.6	authentication key-chain	320
23.7.7	passive-interface	320
23.7.8	timers update.....	320
23.7.9	timers invalid.....	321

23.7.10	timers flush	322
23.7.11	ip rip metric	322
23.7.12	ip rip mode	323
23.7.13	ip rip neighbor	324
23.7.14	ip rip summary-address	324
23.7.15	show ip rip	325
23.7.16	clear ip rip	325
23.8	Настройка протокола OSPF	326
23.8.1	router ospf	326
23.8.2	router-id	326
23.8.3	preference	327
23.8.4	compatible rfc1583	327
23.8.5	area	327
23.8.6	area-type	328
23.8.7	default-information-originate	328
23.8.8	summary-address	329
23.8.9	virtual-link	329
23.8.10	retransmit-interval	330
23.8.11	hello-interval	330
23.8.12	dead-interval	331
23.8.13	enable	332
23.8.14	ip ospf instance	332
23.8.15	ip ospf area	333
23.8.16	ip ospf	334
23.8.17	ip ospf mtu-ignore	334
23.8.18	ip ospf authentication algorithm	335
23.8.19	ip ospf authentication key	335
23.8.20	ip ospf authentication key-chain	336
23.8.21	ip ospf wait-interval	337
23.8.22	ip ospf retransmit-interval	337
23.8.23	ip ospf hello-interval	338
23.8.24	ip ospf dead-interval	339
23.8.25	ip ospf poll-interval	339
23.8.26	ip ospf neighbor	340
23.8.27	ip ospf network	341
23.8.28	ip ospf priority	342
23.8.29	ip ospf cost	342
23.8.30	router ospf log-neighbor-changes	343
23.8.31	clear ip ospf	343
23.8.32	show ip ospf	344
23.8.33	show ip ospf interface	344
23.8.34	show ip ospf database	345
23.8.35	show ip ospf neighbors	346
23.8.36	show ip ospf virtual-links	346
23.9	Настройка протокола OSPFv3	347
23.9.1	ipv6 router ospf	347
23.9.2	router-id	347
23.9.3	preference	348
23.9.4	compatible rfc1583	348
23.9.5	area	349
23.9.6	area-type	349
23.9.7	default-information-originate	350
23.9.8	summary-address	350
23.9.9	virtual-link	351

23.9.10	retransmit-interval.....	351
23.9.11	hello-interval.....	352
23.9.12	dead-interval.....	352
23.9.13	enable	353
23.9.14	ipv6 ospf instance	354
23.9.15	ipv6 ospf area	354
23.9.16	ipv6 ospf.....	355
23.9.17	ipv6 ospf mtu-ignore	355
23.9.18	ipv6 ospf wait-interval	356
23.9.19	ipv6 ospf retransmit-interval	356
23.9.20	ipv6 ospf hello-interval	357
23.9.21	ipv6 ospf dead-interval	358
23.9.22	ipv6 ospf poll-interval	358
23.9.23	ipv6 ospf neighbor	359
23.9.24	ipv6 ospf network	360
23.9.25	ipv6 ospf priority.....	360
23.9.26	ipv6 ospf cost.....	361
23.9.27	clear ipv6 ospf.....	362
23.9.28	show ipv6 ospf.....	362
23.9.29	show ipv6 ospf interface.....	362
23.9.30	show ipv6 ospf database	363
23.9.31	show ipv6 ospf neighbors	364
23.9.32	show ipv6 ospf virtual-links	364
24	РЕЗЕРВИРОВАНИЕ.....	366
24.1	Управление VRRP.....	366
24.1.1	vrrp.....	366
24.1.2	vrrp ip	366
24.1.3	vrrp id	367
24.1.4	vrrp priority	368
24.1.5	vrrp group	368
24.1.6	vrrp source-ip	369
24.1.7	vrrp timers advertise.....	369
24.1.8	vrrp timers garp delay	370
24.1.9	vrrp timers garp repeat.....	371
24.1.10	vrrp timers garp refresh.....	371
24.1.11	vrrp timers garp refresh-repeat.....	372
24.1.12	vrrp preempt disable	373
24.1.13	vrrp preempt delay	373
24.1.14	vrrp authentication key	374
24.1.15	vrrp authentication algorithm	375
24.1.16	vrrp version.....	375
24.1.17	show vrrp	376
24.2	Управление IPv6 VRRP	377
24.2.1	ipv6 vrrp	377
24.2.2	ipv6 vrrp ip	377
24.2.3	ipv6 vrrp id	378
24.2.4	ipv6 vrrp priority	378
24.2.5	ipv6 vrrp group.....	379
24.2.6	ipv6 vrrp source-ip	379
24.2.7	ipv6 vrrp timers advertise	380
24.2.8	ipv6 vrrp preempt disable.....	381
24.2.9	ipv6 vrrp preempt delay.....	381
24.2.10	ipv6 vrrp authentication key.....	382
24.2.11	ipv6 vrrp authentication algorithm	382

24.2.12	ipv6 vrrp version	383
24.2.13	show ipv6 vrrp	384
24.3	Настройка резервирования	384
24.3.1	Настройка резервирования DHCP	384
24.3.2	Настройка резервирования Firewall	387
24.3.3	show high-availability state	391
24.4	Управление Dual-Homing	392
24.4.1	backup-interface preemption	392
24.4.2	backup-interface mac-duplicate	392
24.4.3	backup-interface mac-per-second	393
24.4.4	backup interface	393
24.4.5	show interfaces backup	394
24.5	Настройка MultiWAN	394
24.5.1	wan load-balance rule	394
24.5.2	ipv6 wan load-balance rule	395
24.5.3	wan load-balance target-list	395
24.5.4	ipv6 wan load-balance target-list	396
24.5.5	enable	396
24.5.6	outbound	397
24.5.7	description	398
24.5.8	failover	398
24.5.9	target	399
24.5.10	resp-time	399
24.5.11	ip address	400
24.5.12	ipv6 address	400
24.5.13	wan load-balance enable	401
24.5.14	ipv6 wan load-balance enable	401
24.5.15	wan load-balance failure-count	402
24.5.16	ipv6 wan load-balance failure-count	402
24.5.17	wan load-balance success-count	403
24.5.18	ipv6 wan load-balance success-count	404
24.5.19	wan load-balance nexthop	404
24.5.20	ipv6 wan load-balance nexthop	405
24.5.21	wan load-balance source-address	405
24.5.22	ipv6 wan load-balance source-address	406
24.5.23	wan load-balance target-list check-all	406
24.5.24	ipv6 wan load-balance target-list check-all	407
25	УПРАВЛЕНИЕ QOS	408
25.1	qos enable	408
25.2	qos trust	409
25.3	qos map dscp-queue	409
25.4	qos map cos-queue	410
25.5	qos map dscp-mutation	411
25.6	qos dscp-mutation	411
25.7	qos queue default	412
25.8	priority-queue out num-of-queues	412
25.9	qos wrr-queue	413
25.10	traffic-shape	413
25.11	rate-limit	414
25.12	service-policy	415
25.13	policy-map	416
25.14	ip firewall session classification enable	416
25.15	class-map	416
25.16	class	417

25.17	set dscp	417
25.18	set cos	418
25.19	set ip-precedence.....	418
25.20	set queue.....	419
25.21	match access-group	419
25.22	service-policy.....	420
25.23	shape auto-distribution.....	420
25.24	shape average	420
25.25	shape peak	421
25.26	mode	421
25.27	priority class	422
25.28	priority level	423
25.29	fair-queue.....	423
25.30	queue-limit.....	424
25.31	random-detect	424
25.32	random-detect precedence	425
25.33	show qos interface.....	425
25.34	show qos tunnel.....	426
25.35	show qos statistics	426
25.36	show qos policy binding.....	427
25.37	show qos policy configuration	427
25.38	show qos policy statistics.....	428
25.39	show qos map dscp-queue	429
25.40	show qos map cos-queue.....	429
25.41	show qos map dscp-mutation.....	430
26	УПРАВЛЕНИЕ NETFLOW	431
26.1	netflow collector	431
26.2	port.....	431
26.3	netflow version	432
26.4	netflow max-flows.....	432
26.5	netflow inactive-timeout	433
26.6	netflow refresh-rate.....	433
26.7	netflow enable	434
26.8	ip netflow export.....	434
26.9	show netflow configuration	435
26.10	show netflow statistics.....	435
26.11	show netflow statistics cpu.....	436
27	УПРАВЛЕНИЕ SFLOW.....	437
27.1	sflow collector	437
27.2	port.....	437
27.3	sflow poll-interval	438
27.4	sflow sampling-rate.....	438
27.5	sflow enable	439
27.6	ip sflow export.....	439
27.7	show sflow configuration.....	440
28	МОНИТОРИНГ И УПРАВЛЕНИЕ	441
28.1	Настройка SNMP	441
28.1.1	snmp-server	441
28.1.2	snmp-server host	441
28.1.3	port.....	442
28.1.4	snmp-server community	442
28.1.5	snmp-server filter.....	443
28.1.6	snmp-server dscp.....	444
28.1.7	snmp-server user	444
28.1.8	access	445

28.1.9	authentication access	445
28.1.10	authentication algorithm.....	446
28.1.11	authentication key.....	446
28.1.12	enable	447
28.1.13	privacy algorithm.....	447
28.1.14	privacy key.....	448
28.2	Управление SYSLOG.....	449
28.2.1	syslog console	449
28.2.2	syslog monitor	450
28.2.3	syslog cli-commands	450
28.2.4	syslog file.....	451
28.2.5	syslog file-size	451
28.2.6	syslog max-files	452
28.2.7	syslog host	452
28.2.8	clear log.....	453
28.2.9	show syslog.....	454
28.3	Настройка доступа SSH, Telnet	455
28.3.1	ip ssh server	455
28.3.2	ip ssh port	455
28.3.3	ip ssh dscp.....	456
28.3.4	ip ssh client username	456
28.3.5	ip ssh client password.....	457
28.3.6	crypto key generate	457
28.3.7	show crypto key mypubkey	458
28.3.8	ip telnet server.....	458
28.3.9	ip telnet port.....	459
28.3.10	ip telnet dscp	459
28.3.11	ip ftp client username.....	460
28.3.12	ip ftp client password	460
29	НАСТРОЙКА DHCP	462
29.1	Управление DHCP-клиентом	462
29.1.1	ip address dhcp	462
29.1.2	ip dhcp server address	462
29.1.3	ip dhcp client lease-time.....	463
29.1.4	ip dhcp client timeout.....	463
29.1.5	ip dhcp client retry.....	464
29.1.6	ip dhcp client reboot.....	465
29.1.7	ip dhcp client select-timeout	465
29.1.8	ip dhcp client vendor-class-id	466
29.1.9	ip dhcp client ignore	466
29.2	Управление IPv6 DHCP-клиентом	467
29.2.1	ipv6 address dhcp	467
29.2.2	ipv6 dhcp server address	468
29.2.3	ipv6 dhcp client lease-time	468
29.2.4	ipv6 dhcp client timeout.....	469
29.2.5	ipv6 dhcp client retry	470
29.2.6	ipv6 dhcp client reboot.....	470
29.2.7	ipv6 dhcp client select-timeout	471
29.2.8	ipv6 dhcp client vendor-class-id	472
29.2.9	ipv6 dhcp client ignore.....	472
29.3	Управление DHCP Relay агентом.....	473
29.3.1	ip dhcp-relay	473
29.3.2	ip helper-address	473
29.4	Управление IPv6 DHCP Relay агентом	474

29.4.1ipv6 dhcp-relay	474
29.4.2ipv6 dhcp-relay interface	474
29.5 Настройка и мониторинг DHCP-сервера.....	475
29.5.1ip dhcp-server	475
29.5.2ip dhcp-server dscp	476
29.5.3ip dhcp-server pool	476
29.5.4network	477
29.5.5address-range	477
29.5.6address	478
29.5.7default-router	478
29.5.8domain-name	479
29.5.9dns-server	479
29.5.10 max-lease-time	480
29.5.11 default-lease-time	480
29.5.12 ip dhcp-server vendor-class-id.....	481
29.5.13 vendor-specific-options	481
29.5.14 netbios-name-server	482
29.5.15 show ip dhcp binding.....	482
29.5.16 show ip dhcp server dscp	483
29.5.17 show ip dhcp server pool.....	483
29.5.18 show ip dhcp server vendor-specific	484
29.6 Настройка и мониторинг IPv6 DHCP-сервера.....	484
29.6.1ipv6 dhcp-server	484
29.6.2ipv6 dhcp-server pool	485
29.6.3network	485
29.6.4address-range	485
29.6.5address	486
29.6.6domain-name	487
29.6.7dns-server	487
29.6.8max-lease-time	488
29.6.9default-lease-time	488
29.6.10 ipv6 dhcp-server vendor-class-id.....	489
29.6.11 vendor-specific-options	489
29.6.12 show ipv6 dhcp binding	490
29.6.13 show ipv6 dhcp server pool	490
29.6.14 show ipv6 dhcp server vendor-specific	491

1 ВВЕДЕНИЕ

1.1 Аннотация

В настоящем руководстве приведено описание команд CLI для администратора маршрутизатора серии ESR (в дальнейшем именуемого устройством).

Интерфейс командной строки (Command Line Interface, CLI) – интерфейс, предназначенный для управления, просмотра состояния и мониторинга устройства. Для работы потребуется любая установленная на ПК программа, поддерживающая работу по протоколу Telnet, SSH или прямое подключение через консольный порт (например, HyperTerminal).

1.2 Целевая аудитория

Справочник команд CLI предназначен для технического персонала, выполняющего настройку и мониторинг маршрутизатора серии ESR посредством интерфейса командной строки (CLI). Квалификация технического персонала предполагает знание основ работы стека протоколов TCP/IP, принципов построения Ethernet-сетей.

1.3 Условные обозначения

Обозначения	Описание
Полужирный шрифт	Полужирным шрифтом выделены примечания и предупреждения, название глав, заголовков, заголовков таблиц.
<code>Courier New</code>	Шрифтом Courier New записаны примеры ввода команд, результат их выполнения, вывод программ.
[]	В квадратных скобках в командной строке указываются необязательные параметры, но их ввод предоставляет определенные дополнительные опции.
{ }	В фигурных скобках в командной строке указываются возможные обязательные параметры. Необходимо выбрать один из параметров.
« »	Данный знак в описании команды обозначает «или».

Примечания и предупреждения



Примечания содержат важную информацию, советы или рекомендации по использованию и настройке устройства.



Предупреждения информируют пользователя о ситуациях, которые могут нанести вред программно-аппаратному комплексу, привести к некорректной работе системы или потере данных.

1.4 Используемые сокращения

BGP – Border Gateway Protocol
DHCP – Dynamic Host Configuration Protocol
DNS – Domain Name System
GRE – Generic Routing Encapsulation
ICMP – Internet Control Message Protocol
IKE – Internet Key Exchange
IP4IP4 – IP in IP
IPsec – IP Security
LACP – Link Aggregation Control Protocol
LAG – Link Aggregation Group
L2TP – Layer 2 Tunneling Protocol
L2TPv3 – Layer 2 Tunneling Protocol version 3
MTU – Maximum Transmission Unit
NAT – Network Address Translation
NTP – Network Time Protocol
OSPF – Open Shortest Path First
PPP – Point-to-Point Protocol
PPTP – Point-to-Point Tunneling Protocol
QoS – Quality of service
RIP – Routing Informational Protocol
SNMP – Simple Network Management Protocol
SP – Strict Priority
VPN – Virtual Private Network
VRF - Virtual Routing/Forwarding
VRRP – Virtual Router Redundancy Protocol
WAN – Wide Area Network
WINS – Windows Internet Name Service
WRR – Weighted Round Robin

2 ПРАВИЛА ПОЛЬЗОВАНИЯ КОМАНДНОЙ СТРОКОЙ

Для упрощения использования командной строки интерфейс поддерживает функцию автоматического дополнения команд. Эта функция активируется при неполно набранной команде и вводе символа табуляции <Tab>.

Другая функция, помогающая пользоваться командной строкой – контекстная подсказка. На любом этапе ввода команды можно получить подсказку о следующих элементах команды путем ввода вопросительного знака <?>.

Для упрощения команд всей системе команд придана иерархическая структура. Для перехода между уровнями иерархии предназначены специальные команды перехода. Это позволяет использовать менее объемные команды на каждом из уровней. Для обозначения текущего уровня, на котором находится пользователь, динамически изменяется строка приглашения системы.

Пример:

<code>esr> enable</code>	включение 15 уровня привилегий
<code>esr# configure</code>	переход в режим конфигурирования устройства
<code>esr(config)#</code>	
<code>esr(config)# exit</code>	возврат на уровень выше
<code>esr#</code>	

Для удобства использования командной строки реализована поддержка горячих клавиш, перечисленных в таблице 2.1.

Таблица 2.1 – Описание горячих клавиш командной строки CLI

Сочетание клавиши	Описание
Ctrl+D	Во вложенном командном режиме – выход в предыдущий командный режим (команда exit), в корневом командном режиме – выход из CLI (команда logout)
Ctrl+Z	Выход в корневой командный режим (команда top)
Ctrl+A	Переход в начало строки
Ctrl+E	Переход в конец строки
Ctrl+U	Удаление символов слева от курсора
Ctrl+K	Удаление символов справа от курсора
Ctrl+C	Очистка строки, а также обрыв выполнения команды
Ctrl+W	Удаление слова слева от курсора
Ctrl+B	Переход курсора на одну позицию назад
Ctrl+F	Переход курсора на одну позицию вперед
Ctrl+L	Очистка экрана

Для удобства чтения добавлен постраничный вывод большой по объему информации.

Например:

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

interface gigabitethernet 1/0/1
 ip address 10.100.14.1/24
exit
interface gigabitethernet 1/0/1.101
exit
interface gigabitethernet 1/0/2
 ip address 192.168.1.1/24
```

```
ip address 10.100.100.2/24
exit
interface gigabitethernet 1/0/2.150
  ip address 10.150.150.2/24
exit
interface gigabitethernet 1/0/2.151
  ip address 10.151.151.15/24
exit
interface gigabitethernet 1/0/3
  ip address dhcp enable
exit
interface gigabitethernet 1/0/5.55
More? Enter - next line; Space - next page; Q - quit; R - show the rest.
```

Для отключения постраничного вывода в текущей сессии необходимо ввести команду:

```
esr# terminal datadump
```

Интерфейс командной строки обеспечивает авторизацию пользователей и ограничивает доступ к командам на основании уровня привилегий, заданного администратором.

В системе может быть создано необходимое количество пользователей. Необходимый уровень привилегий задаётся индивидуально для каждого из них.



В заводской конфигурации в системе создан один пользователь с именем admin и паролем password.

Для обеспечения безопасности командного интерфейса команды распределены между 1, 10 и 15 уровнем привилегий:

1 уровень – доступен только мониторинг устройства;

10 уровень – доступно конфигурирование устройства, кроме создания пользователей, перезагрузки устройства, загрузки ПО;

15 уровень – нет ограничений.

Получение 15 уровня привилегий:

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

Возвращение на первоначальный уровень привилегий:

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

Система позволяет нескольким пользователям одновременно подключаться к устройству.

Для уменьшения объема отображаемых данных в ответ на запросы пользователя и облегчения поиска необходимой информации можно воспользоваться фильтрацией. Для фильтрации информации требуется добавить в конец командной строки символ «|» и использовать одну из опций фильтрации:

- begin – выводить все после строки, содержащей заданный шаблон;
- include – выводить все строки, содержащие заданный шаблон;
- exclude – выводить все строки, не содержащие заданный шаблон.

Шаблон поиска может быть задан регулярным выражением и содержать:

- Перечень символов. Можно определить перечень, заключив символы в квадратные скобки «[]». Соответствие будет проверяться по символам, перечисленным в перечне. Если первый символ перечня «^», то соответствие будет проверяться по любому символу, не входящему в перечень. Примеры:
 [-az] - 'a', 'z' и '-';
 [a-z] - все латинские буквы от 'a' до 'z'.
- Специальные символы:
 ^ – начало строки;
 \$ – конец строки;
 . – любой символ в строке;
 * – ноль или более раз.

Вывод команды «show running-config syslog» без параметров:

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

Вывод команды «show running-config syslog» с параметром «begin»:

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

Вывод команды «show running-config syslog» с параметром «include»:

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

Вывод команды «show running-config syslog» с параметром «exclude»:

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

Примеры использования регулярных выражений:

```
esr# show interfaces status | include "^[^tgi -]"
bridge 1      Up      Up      1500      a8:f9:4b:aa:05:c0
bridge 2      Up      Up      1500      a8:f9:4b:aa:05:c0
```

3 СТРУКТУРА СИСТЕМЫ КОМАНД

Система команд интерфейса командной строки маршрутизатора серии ESR разделена на иерархические уровни (разделы).

3.1 Глобальный режим

Верхний уровень иерархии команд приведен в таблице 3.1.

Таблица 3.1 – Иерархия командных режимов (верхний уровень)

Уровень	Команда входа	Вид строки подсказки	Команда выхода
Корневой режим (ROOT)		esr> esr#	exit end
Режим конфигурирования (CONFIG)	configure	esr(config)#	
Режим отладки работы устройства (DEBUG)	debug	esr(debug)#	

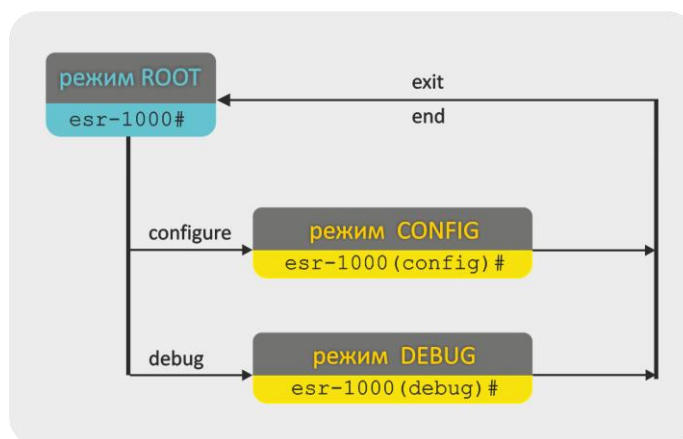


Рисунок 1 – Верхний уровень иерархии режимов команд

В корневом командном режиме (ROOT) осуществляется:

- работа с файлами конфигурации:
 - применение;
 - подтверждение;
 - сброс;
 - сохранение;
 - отмена не примененных изменений;
 - возврат к подтвержденной конфигурации;
- перезагрузка маршрутизатора;
- мониторинг работы и просмотр текущей конфигурации устройства.

Из корневого режима (ROOT) осуществляется переход к следующим разделам:

- режим конфигурирования устройства (CONFIG);
- режим отладки работы устройства (DEBUG).

3.2 Конфигурирование маршрутизатора

Конфигурирование маршрутизатора серии ESR выполняется в режиме **CONFIG**. Данный режим доступен из корневого режима (ROOT). Переход в режим конфигурирования осуществляется только в привилегированном режиме.

Для перехода из корневого режима (ROOT) необходимо выполнить следующие команды:

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

В режиме конфигурирования маршрутизатора серии ESR выполняется:

- управление системными часами;
- управление системным журналом;
- управление удаленным доступом;
- настройка QoS;
- настройка Spanning Tree;
- настройка VLAN;
- настройка статических маршрутов;
- настройка приоритетности протоколов маршрутизации;
- переход к режимам конфигурирования функций, описание режимов приведено в таблице 3.2.

Таблица 3.2 – Командные режимы для управления маршрутизатором

Уровень	Команда входа ¹	Вид строки подсказки	Предыдущий уровень
Настройка GigabitEthernet интерфейсов (CONFIG-GI)	interface gigabitethernet <PORT>	esr(config-if-gi)#	CONFIG
Настройка группы агрегации каналов (CONFIG-PORT-CHANNEL)	intrerface port-channel <CH>	esr(config-port-channel)#	CONFIG
Настройка субинтерфейсов (CONFIG-SUBIF)	interface gigabitethernet <PORT>.<VLAN> или interface port-channel <CH>.<VLAN>	esr(config-subif)#	CONFIG
Настройка виртуальных интерфейсов (CONFIG-LOOPBACK)	interface loopback <PORT>	esr(config-loopback)#	CONFIG
Настройка E1-интерфейсов (CONFIG-E1)	interface e1 <PORT>	esr(config-e1)#	CONFIG
Настройка группы агрегации E1-каналов (CONFIG-MULTILINK)	interface multilink <PORT>	esr(config-multilink)#	CONFIG
Настройка виртуальных туннельных интерфейсов VTI (CONFIG-VTI)	tunnel vti <VTI>	esr(config-vti)#	CONFIG
Настройка L2TPv3-туннелей (CONFIG-L2TPV3)	tunnel l2tpv3 <L2TPV3>	esr(config-l2tpv3)#	CONFIG
Настройка GRE-туннелей	tunnel gre <GRE>	esr(config-gre)#	CONFIG

¹ Подробное описание команд приведено ниже

(CONFIG-GRE)			
Настройка SoftGRE-туннелей (CONFIG-SOFTGRE)	tunnel softgre <SOFTGRE>	esr(config-softgre)#	CONFIG
Настройка сабинтерфейса на L2 туннеле (CONFIG-SUBTUNNEL)	tunnel softgre <SOFTGRE>.<VLAN>	esr(config-subtunnel)#	CONFIG
Настройка IPv4-over-IPv4 туннелей (CONFIG-IP4IP4)	tunnel ip4ip4 <IP4IP4>	esr(config-ip4ip4)#	CONFIG
Настройка сетевых мостов (CONFIG-BRIDGE)	bridge <BRIDGE>	esr(config-bridge)#	CONFIG
Настройка VLAN (CONFIG-VLAN)	vlan <VLAN>	esr(config-vlan)#	CONFIG
Настройка пула адресов DHCP-сервера (CONFIG-DHCP-SERVER)	ip dhcp-server pool <NAME>	esr(config-dhcp-server)#	CONFIG
Настройка DHCP опции 60 (CONFIG-DHCP-VENDOR-ID)	ip dhcp-server vendor-class-id <NAME>	esr(config-dhcp-vendor-id)#	CONFIG
Настройка пула адресов DHCP-сервера (CONFIG-IPV6-DHCP-SERVER)	ipv6 dhcp-server pool <NAME>	esr(config-ipv6-dhcp-server)#	CONFIG
Настройка DHCP опции 60 (CONFIG-IPV6-DHCP-VENDOR-ID)	ipv6 dhcp-server vendor-class-id <NAME>	esr(config-dhcp-vendor-id)#	CONFIG
Настройка профиля IP-адресов (CONFIG-OBJECT-GROUP-NETWORK)	object-group network <NAME>	esr(config-object-group-network)#	CONFIG
Настройка профиля TCP/UDP-портов (CONFIG-OBJECT-GROUP-SERVICE)	object-group service <NAME>	esr(config-object-group-service)#	CONFIG
Настройка шлюза протокола IKE (CONFIG-IKE-GATEWAY)	security ike gateway <NAME>	esr(config-ike-gw)#	CONFIG
Настройка политики протокола IKE (CONFIG-IKE-POLICY)	security ike policy <NAME>	esr(config-ike-policy)#	CONFIG
Настройка профиля протокола IKE (CONFIG-IKE-PROPOSAL)	security ike proposal <NAME>	esr(config-ike-proposal)#	CONFIG
Настройка политики набора протоколов IPsec (CONFIG-IPSEC-POLICY)	security ipsec policy <NAME>	esr(config-ipsec-policy)#	CONFIG
Настройка профиля набора протоколов IPsec (CONFIG-IPSEC-PROPOSAL)	security ipsec proposal <NAME>	esr(config-ipsec-proposal)#	CONFIG
Настройка VPN на основе набора протоколов IPsec (CONFIG-IPSEC-VPN)	security ipsec vpn <NAME>	esr(config-ipsec-vpn)#	CONFIG
Настройка списка контроля доступа (CONFIG-ACL)	ip access-list extended <NAME>	esr(config-acl)#	CONFIG
Настройка правила для списка контроля доступа (CONFIG-ACL-RULE)	rule <ORDER>	esr(config-acl-rule)#	CONFIG-ACL
Настройка зоны безопасности (CONFIG-ZONE)	security zone <NAME>	esr(config-zone)#	CONFIG
Настройка группы правил для пар зон безопасности (CONFIG-ZONE-PAIR)	security zone-pair <FROM> <TO>	esr(config-zone-pair)#	CONFIG
Настройка правила для пары зон безопасности (CONFIG-ZONE-PAIR-RULE)	rule <ORDER>	esr(config-zone-rule)#	CONFIG-ZONE-PAIR
Настройка сервиса трансляции адресов получателя (CONFIG-DNAT)	nat destination	esr(config-dnat)#	CONFIG

Настройка пула IP-адресов и TCP/UDP-портов для DNAT (CONFIG-DNAT-POOL)	pool <NAME>	esr(config-dnat-pool)#	CONFIG-DNAT
Настройка группы правил для DNAT (CONFIG-DNAT-RULESET)	ruleset <NAME>	esr(config-dnat-ruleset)#	CONFIG-DNAT
Настройка правила для DNAT (CONFIG-DNAT-RULE)	rule <ORDER>	esr(config-dnat-rule)#	CONFIG-DNAT-RULESET
Настройка сервиса трансляции адресов отправителя (CONFIG-SNAT)	nat source	esr(config-snat)#	CONFIG
Настройка пула IP-адресов и TCP/UDP-портов для SNAT (CONFIG-SNAT-POOL)	pool <NAME>	esr(config-snat-pool)#	CONFIG-SNAT
Настройка группы правил для SNAT (CONFIG-SNAT-RULESET)	ruleset <NAME>	esr(config-snat-ruleset)#	CONFIG-SNAT
Настройка правила для SNAT (CONFIG-SNAT-RULE)	rule <ORDER>	esr(config-snat-rule)#	CONFIG-SNAT-RULESET
Настройка пользователей системы (CONFIG-USER)	username <NAME>	esr(config-user)#	CONFIG
Настройка локальной консоли (CONFIG-LINE-CONSOLE)	line console	esr(config-line-console)#	CONFIG
Настройка удаленной консоли (CONFIG-LINE-TELNET)	line telnet	esr(config-line-telnet)#	CONFIG
Настройка защищенной удаленной консоли (CONFIG-LINE-SSH)	line ssh	esr(config-line-ssh)#	CONFIG
Настройка TACACS-сервера (CONFIG-TACACS-SERVER)	tacacs-server host <ADDR>	esr(config-tacacs-server)#	CONFIG
Настройка RADIUS-сервера (CONFIG-RADIUS-SERVER)	radius-server host <ADDR>	esr(config-radius-server)#	CONFIG
Настройка LDAP-сервера (CONFIG-LDAP-SERVER)	ldap-server host <ADDR>	esr(config-ldap-server)#	CONFIG
Настройка сервера для удаленного доступа по протоколу PPTP (CONFIG-PPTP)	remote-access pptp <NAME>	esr(config-pptp)#	CONFIG
Настройка пользователей для удаленного доступа по протоколу PPTP (CONFIG-PPTP-USER)	username <NAME>	esr(config-pptp-user)#	CONFIG-PPTP
Настройка сервера для удаленного доступа по протоколу L2TP (CONFIG-L2TP)	remote-access l2tp <NAME>	esr(config-l2tp)#	CONFIG
Настройка пользователей для удаленного доступа по протоколу L2TP (CONFIG-L2TP-USER)	username <NAME>	esr(config-l2tp-user)#	CONFIG-L2TP
Настройка SNMP-пользователя (CONFIG-SNMP-USER)	snmp-server <NAME>	esr(snmp-user)#	CONFIG
Настройка NTP-сервера или пир (CONFIG-NTP)	service ntp peer <ADDR> service ntp server <ADDR>	esr(config-ntp)#	CONFIG
Настройка BGP-процесса (CONFIG-BGP)	router bgp <AS>	esr(config-bgp)#	CONFIG
Настройка ipv4/ipv6 адресации BGP-процесса (CONFIG-BGP-FAMILY)	address-family { ipv4 ipv6 }	esr(config-bgp-af)# esr(config-ipv6-bgp-af)#	CONFIG-BGP
Настройка соседа BGP-процесса (CONFIG-BGP-NEIGHBOR)	neighbor <ADDR>	esr(config-bgp-neighbor)#	CONFIG-BGP-FAMILY
Настройка списка подсетей	ip prefix-list <NAME>	esr(config-pl)#	CONFIG

(CONFIG-PL)			
Настройка списка подсетей (CONFIG-IPV6-PL)	ipv6 prefix-list <NAME>	esr(config-ipv6-pl)#	CONFIG
Настройка маршрутной карты (CONFIG-ROUTE-MAP)	route-map <NAME>	esr(config-route-map)#	CONFIG
Настройка правила маршрутной карты (CONFIG-ROUTE-MAP-RULE)	rule <ORDER>	esr(config-route-map-rule)#	CONFIG-ROUTE-MAP
Настройка RIP-протокола (CONFIG-RIP)	router rip	esr(config-rip)#	CONFIG
Настройка OSPF-процесса (CONFIG-OSPF)	router ospf <ID>	esr(config-ospf)#	CONFIG
Настройка OSPF-области (CONFIG-OSPF-AREA)	area <ID>	esr(config-ospf-area)#	CONFIG-OSPF
Настройка виртуального соединения OSPF (CONFIG-OSPF-VLINK)	virtual-link <ID>	esr(config-ospf-vlink)#	CONFIG-OSPF-AREA
Настройка OSPFv3-процесса (CONFIG-OSPFV3)	ipv6 router ospf <ID>	esr(config-ipv6-ospf)#	CONFIG
Настройка OSPFv3-области (CONFIG-OSPFV3-AREA)	area <ID>	esr(config-ipv6-ospf-area)#	CONFIG-OSPFV3
Настройка виртуального соединения OSPFv3 (CONFIG-OSPFV3-VLINK)	virtual-link <ID>	esr(config-ipv6-ospf-vlink)#	CONFIG-OSPFV3-AREA
Настройка списка ключей (CONFIG-KEYCHAIN)	key-chain <KEYCHAIN>	esr(config-keychain)#	CONFIG
Настройка ключа (CONFIG-KEYCHAIN-KEY)	key <ID>	esr(config-keychain-key)#	CONFIG-KEYCHAIN
Настройка правил WAN (CONFIG-WAN-RULE)	wan load-balance rule <ID>	esr(config-wan-rule)#	CONFIG
Настройка правил WAN (ipv6) (CONFIG-IPV6-WAN-RULE)	Ipv6 wan load-balance rule <ID>	esr(config-ipv6-wan-rule)#	CONFIG
Настройка target-листов (CONFIG-TARGET-LIST)	wan load-balance target-list <NAME>	esr(config-target-list)#	CONFIG
Настройка target-листов (ipv6) (CONFIG-IPV6-TARGET-LIST)	Ipv6 wan load-balance target-list <NAME>	esr(config-ipv6-target-list)#	CONFIG
Настройка target (CONFIG-WAN-TARGET)	target <ID>	esr(config-wan-target)#	CONFIG-TARGET-LIST
Настройка target (ipv6) (CONFIG-IPV6-WAN-TARGET)	target <ID>	esr(config-ipv6-wan-target)#	CONFIG-IPV6-TARGET-LIST
Настройка VRF (CONFIG-VRF)	ip vrf <NAME>	esr(config-vrf)#	CONFIG
Настройка политики QoS (CONFIG-POLICY-MAP)	policy-map <NAME>	esr(config-policy-map)#	CONFIG
Настройка класса QoS (CONFIG-CLASS-MAP)	class-map <NAME>	esr(config-class-map)#	CONFIG
Настройка класса внутри политики QoS (CONFIG-POLICY-MAP-CLASS)	class <NAME>	esr(config-class-policy-map)#	CONFIG
Настройка PPP-пользователя для аутентификации удаленной стороны (CONFIG-PPP-USER)	ppp chap username <NAME>	esr(config-ppp-user)#	CONFIG-E1 CONFIG-MULTILINK
Настройка параметров резервирования конфигурации (CONFIG-ARCHIVE)	archive	esr(config-archive)#	CONFIG
Настройка сервера сбора статистики Netflow	netflow collector <ADDR>	esr(config-netflow-host)#	CONFIG

(CONFIG-NETFLOW-HOST)			
Настройка сервера сбора статистики sFlow (CONFIG-SFLOW-HOST)	sflow collector <ADDR>	esr(config-sflow-host)#	CONFIG
Настройка сервера получения уведомлений SNMP (CONFIG-SNMP-HOST)	snmp-server host <ADDR>	esr(config-snmp-host)#	CONFIG

3.3 Типы и порядок именования интерфейсов маршрутизатора

При работе маршрутизатора используются сетевые интерфейсы различного типа и назначения. Система именования позволяет однозначно адресовать интерфейсы по их функциональному назначению и местоположению в системе. Далее в таблице приведен перечень типов интерфейсов.

Таблица 3.3 – Типы и порядок именования интерфейсов маршрутизатора

Тип интерфейса	Обозначение
Физические интерфейсы	Обозначение физического интерфейса включает в себя его тип и идентификатор. Идентификатор физических интерфейсов имеет вид <UNIT>/<SLOT>/<PORT> , где - <UNIT> – номер устройства в группе устройств, - <SLOT> – номер модуля в составе устройства или 0 при отсутствии деления устройства на модули, - <PORT> – порядковый номер порта.
Порты 1Гбит/с	gigabitethernet <UNIT>/<SLOT>/<PORT> Пример обозначения: gigabitethernet 1/0/2 Примечание: Допускается использовать сокращенное наименование, например gi1/0/2.
Группы агрегации каналов	Обозначение группы агрегации каналов включает в себя его тип и порядковый номер интерфейса: port-channel <CHANNEL_ID> Пример обозначения: port-channel 6  Допускается использовать сокращенное наименование, например, po1.
Субинтерфейсы	Обозначение субинтерфейса образуется из обозначения базового интерфейса и идентификатора (VLAN) субинтерфейса, разделенных точкой. Примеры обозначений: gigabitethernet 1/0/12.100 port-channel 1.6 Примечание: Идентификатор субинтерфейса может принимать значения [1..4094].
E1-интерфейсы	Обозначение E1-интерфейса включает в себя его тип и идентификатор. Идентификатор E1-интерфейсов имеет вид <UNIT>/<SLOT>/<STREAM> , где - <UNIT> – номер устройства в группе устройств, - <SLOT> – номер E1-модуля в составе устройства, - <STREAM> – порядковый номер E1-потока. Пример обозначения: e1 1/0/1
Группы агрегации E1-каналов	Обозначение группы агрегации E1-каналов включает в себя его тип и порядковый номер интерфейса: multilink <CHANNEL_ID> Пример обозначения: multilink <CHANNEL_ID>
Логические интерфейсы	Обозначение логического интерфейса является порядковым

	номером интерфейса: Примеры обозначений: loopback 4 bridge 60



1. Количество интерфейсов каждого типа зависит от модели маршрутизатора.
2. Текущая версия ПО не поддерживает стекирование устройств. Номер устройства в группе устройств unit может принимать только значение 1.
3. Некоторые команды поддерживают одновременную работу с группой интерфейсов. Для указания группы интерфейсов может быть использовано перечисление через запятую или указание диапазона идентификаторов через дефис «-».

Примеры указания групп интерфейсов:

```
interface gigabitethernet 1/0/1, gigabitethernet 1/0/5
interface gi1/0/1-3,gi1/0/7
```

3.4 Типы и порядок именования туннелей маршрутизатора

При работе маршрутизатора используются сетевые туннели различного типа и назначения. Система именования позволяет однозначно адресовать туннели по их функциональному назначению. Далее в таблице приведен перечень типов туннелей.

Таблица 3.4 – Типы и порядок именования туннелей маршрутизатора

Тип туннеля	Обозначение
L2TPv3-туннель	Обозначение L2TPv3-туннеля состоит из обозначения типа и порядкового номера туннеля: l2tpv3 <L2TPV3_ID> Пример обозначения: l2tpv3 1
GRE-туннель	Обозначение GRE-туннеля состоит из обозначения типа и порядкового номера туннеля: gre <GRE_ID> Пример обозначения: gre 1
IPv4-over-IPv4-туннель	Обозначение IPv4-over-IPv4-туннеля состоит из обозначения типа и порядкового номера туннеля: ip4ip4 <IPIP_ID> Пример обозначения: ip4ip4 1
IPsec туннель	Обозначение виртуального IPsec туннеля состоит из обозначения типа и порядкового номера интерфейса: vti <VTI_ID> Пример обозначения: vti 1



Количество туннелей каждого типа зависит от модели и ПО маршрутизатора.

4 КОМАНДЫ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА

4.1 exit

Данная команда служит для возврата на уровень выше.

Синтаксис

exit

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

Все режимы, кроме корневого.

4.2 end

Команда служит для возврата в корневой командный режим (ROOT).

Синтаксис

end

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

Все режимы, кроме корневого.

4.3 help

Данной командой на дисплей выводится информация о работе с командной строкой.

Синтаксис

help

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

Все режимы.

4.4 show history

Данной командой на дисплей выводится информация о командах, которые использовались в текущей сессии.

Синтаксис

show history [limit]

Параметры

[limit] – число последних введенных команд для отображения.

Необходимый уровень привилегий

1

Командный режим

Все режимы.

Пример

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

4.5 do

Команда do позволяет выполнять команды корневого режима (ROOT) из любого другого режима командного интерфейса.

Синтаксис

do <command>

Параметры

<command> – команда корневого режима.

Необходимый уровень привилегий

1

Командный режим

Все режимы, кроме корневого.

Пример

```
esr(config)# do show version
Boot version:
 1.0.7.16 (date 18/11/2016 time 13:40:59)
SW version:
 1.0.8 build 17[d9bdbda] (date 21/11/2016 time 18:06:41)
HW version:
 1v1
```

4.6 logout

Данной командой завершается сеанс работы пользователя с интерфейсом командной строки CLI.

Синтаксис

logout

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# logout
```

4.7 reload system

Данной командой осуществляется перезагрузка устройства.

Синтаксис

reload system

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
esr# reload system
```

4.8 terminal datadump

Команда используется для выключения постраничного режима вывода трассировок для текущей сессии.

Использование отрицательной команды включает постраничный режим вывода трассировок.

Синтаксис

[no] terminal datadump

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# terminal datadump
```

4.9 uptime

Данной командой осуществляется просмотр продолжительности времени работы устройства.

Синтаксис

uptime

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

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

4.10 ping

Данная команда используется для проверки доступности указанного сетевого устройства.

Синтаксис

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

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа;

<ADDR> – IP-адрес устройства, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес устройства, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<HOSTNAME> – DNS-имя устройства, задаётся строкой до 255 символов;

<TTL> – время жизни IP-пакета, принимает значение [1..255], по умолчанию 64;

<COUNT> – количество передаваемых пакетов, по умолчанию не ограничено;

<SIZE> – размер пакета в байтах, принимает значение [1..65468], по умолчанию 56 байт, что соответствует 64 байтам после добавления заголовка ICMP;

<TIMEOUT> – время ожидания ответа, в секундах. Опция влияет на таймаут, если отсутствуют какие-либо ответы, в противном случае утилита ждёт два RTTs. Принимает значение [1..60], по умолчанию 1 секунда;

<SRC-ADDR> – IP-адрес отправителя, в качестве данного адреса может использоваться любой IP-адрес маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-IPV6-ADDR> – IPv6-адрес отправителя, в качестве данного адреса может использоваться любой IPv6-адрес маршрутизатора, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<IF> – имя интерфейса маршрутизатора, через который будут отправлены пакеты, задаётся в виде, описанном в разделе 3.3;

<TUN> – имя туннеля маршрутизатора, через который будут отправлены пакеты, задаётся в виде, описанном в разделе 3.4;

<HEX> – шаблон данных, которым будет заполняться пакет, задаётся числом в шестнадцатеричной системе до 16 байт;

<DSCP> – DSCP-приоритет в соответствии с RFC 2474, принимает значение [0..63], значение по умолчанию 0;

flood – при указании данной команды пакеты будут отправляться с максимальной скоростью, ответы от устройства не отображаются до окончания выполнения команды;

broadcast – при указании данной команды будет разрешено отправлять пакеты на широковещательный адрес;

<STRATEGY> – стратегия фрагментации пакетов, принимает одно из следующих значений:

- allow-fragmentation – разрешить фрагментацию, не устанавливать флаг DF (don't fragment);
- discovery-pmtu – выполнять изучение PMTU (Path MTU), фрагментировать локально, если размер пакета слишком большой;
- disallow-fragmentation – запретить фрагментацию, в том числе локальную.

<INFO> – только для IPv6. Отправка ICMPv6 Node Information Queries (RFC4620), вместо Echo Request принимает одно из следующих значений:

- name – запрос DNS имен(и) узла;
- ipv6 – запрос IPv6-адресов узла;
- ipv6-global – запрос глобальных IPv6-адресов узла;
- ipv6-sitelocal – запрос site-local IPv6-адресов узла;
- ipv6-linklocal – запрос link-local IPv6-адресов узла;
- ipv6-all – запрос unicast IPv6-адресов узла;
- ipv4 – запрос IP-адресов узла;
- ipv4-all – запрос IP-адресов со всех сетевых интерфейсов узла.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# ping 192.168.100.39 packets 5 size 1400
PING 192.168.100.39 (192.168.100.39) 1400(1428) bytes of data.
1408 bytes from 192.168.100.39: icmp_req=1 ttl=64 time=0.084 ms
1408 bytes from 192.168.100.39: icmp_req=2 ttl=64 time=0.053 ms
1408 bytes from 192.168.100.39: icmp_req=3 ttl=64 time=0.082 ms
1408 bytes from 192.168.100.39: icmp_req=4 ttl=64 time=0.051 ms
1408 bytes from 192.168.100.39: icmp_req=5 ttl=64 time=0.075 ms

--- 192.168.100.39 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3999ms
rtt min/avg/max/mdev = 0.051/0.069/0.084/0.014 ms

esr# ping ipv6 fc00::1
PING fc00::1(fc00::1) 56 data bytes
64 bytes from fc00::1: icmp_seq=1 ttl=64 time=0.379 ms
64 bytes from fc00::1: icmp_seq=2 ttl=64 time=0.161 ms

--- fc00::1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 0.161/0.270/0.379/0.109 ms
```

4.11 traceroute

Данная команда используется для трассировки маршрута до указанного сетевого устройства.

Синтаксис

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

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа;

<ADDR> – IP-адрес устройства, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес устройства, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<HOSTNAME> – DNS-имя устройства, задаётся строкой до 255 символов;

<FIRST-TTL> – время жизни IP-пакета, значение с которого начинается трассировка маршрута, принимает значение [1..255], по умолчанию 1;

<MAX-TTL> – время жизни IP-пакета, значение на котором заканчивается трассировка маршрута, принимает значение [1..255], по умолчанию 30;

<TIMEOUT> – время ожидания ответа, в секундах. Опция влияет на таймаут, если отсутствуют какие-либо ответы, в противном случае утилита ждёт два RTTs. Принимает значение [1..60], по умолчанию 5 секунд;

<SRC-ADDR> – IP-адрес отправителя, в качестве данного адреса может использоваться любой IP-адрес маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-IPV6-ADDR> – IPv6-адрес отправителя, в качестве данного адреса может использоваться любой IPv6-адрес маршрутизатора, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<IF> – имя интерфейса маршрутизатора, через который будут отправлены пакеты, задаётся в виде, описанном в разделе 3.3;

<TUN> – имя туннеля маршрутизатора, через который будут отправлены пакеты, задаётся в виде, описанном в разделе 3.4;

<DSCP> – DSCP-приоритет в соответствии с RFC 2474, принимает значение [0..63], значение по умолчанию 0;

<PORT> – номер TCP/UDP-порта, принимает значение [1..65535], значение по умолчанию 53 для UDP и 80 для TCP;

<GW-ADDR> – IP-адрес шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. При указании данного параметра в исходящий пакет добавляется IP source routing опция, которая сообщает маршрутизатору, через какой шлюз должен маршрутизироваться пакет в сети. На большинство маршрутизаторов отключена маршрутизация по данной опции из соображений безопасности;

<GWIPV6-ADDR> – IPv6-адрес шлюза, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. При указании данного параметра в исходящий пакет добавляется IP source routing опция, которая сообщает маршрутизатору, через какой шлюз должен маршрутизироваться пакет в сети. На большинство маршрутизаторов отключена маршрутизация по данной опции из соображений безопасности.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

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

4.12 ssh

Данная команда используется для подключения к удаленному узлу по протоколу SSH.

Синтаксис

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

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа;

<USERNAME> – имя пользователя, задаётся строкой до 31 символа;

<ADDR> – IP-адрес устройства, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес устройства, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<HOSTNAME> – DNS-имя устройства, задаётся строкой до 255 символов;

<PORT> – номер TCP-порта, прослушиваемого SSH-сервером, принимает значения [1..65535]. По умолчанию установлено 22;

<VERSION> – версия SSH-протокола, принимает значения [1..2];

<SRC-ADDR> – IP-адрес отправителя, в качестве данного адреса может использоваться любой IP-адрес маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-IPV6-ADDR> – IPv6-адрес отправителя, в качестве данного адреса может использоваться любой IPv6-адрес маршрутизатора, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<DSCP> – DSCP-приоритет в соответствии с RFC 2474, принимает значение [0..63], значение по умолчанию 0.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# ssh tester 10.100.100.1
The authenticity of host '10.100.100.1 (10.100.100.1)' can't be established.
ECDSA key fingerprint is db:e4:0a:93:59:87:7d:9f:90:5c:19:a3:e7:97:ec:d5.
Are you sure you want to continue connecting (yes/no)? yes
%AAA-I-SSH: Warning: Permanently added '10.100.100.1' (ECDSA) to the list of known
hosts.
tester@10.100.100.1's password:
Welcome to Ubuntu 14.04.2 LTS (GNU/Linux 3.13.0-51-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

   System information as of Mon May 25 09:25:10 NOVT 2015

Last login: Tue May 12 19:39:11 2015
(teste@kubuntu ~) $
```

4.13 telnet

Данная команда используется для подключения к удаленному узлу по протоколу Telnet.

Синтаксис

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

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа;

<ADDR> – IP-адрес устройства, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес устройства, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<HOSTNAME> – DNS-имя устройства, задаётся строкой до 255 символов;

<PORT> – номер TCP-порта, прослушиваемого SSH-сервером, принимает значения [1..65535], по умолчанию 23;

<SRC-ADDR> – IP-адрес отправителя, в качестве данного адреса может использоваться любой IP-адрес маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-IPV6-ADDR> – IPv6-адрес отправителя, в качестве данного адреса может использоваться любой IPv6-адрес маршрутизатора, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<DSCP> – DSCP-приоритет в соответствии с RFC 2474, принимает значение [0..63], значение по умолчанию 0.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# telnet 10.100.100.1

Entering character mode
Escape character is '^]'.

Ubuntu 14.04.2 LTS
kubuntu login: tester
Password:
Last login: Mon May 25 15:23:06 NOVT 2015 from sw31-1.eltex.loc on pts/16
Welcome to Ubuntu 14.04.2 LTS (GNU/Linux 3.13.0-51-generic x86_64)

* Documentation:  https://help.ubuntu.com/

System information as of Mon May 25 15:23:01 NOVT 2015

(tester@kubuntu ~)$
```

4.14 monitor

Данной командой включается мониторинг трафика на сетевом интерфейсе в режиме реального времени по пакетно.

Синтаксис

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

Параметры

<IF> – интерфейс или группа интерфейсов, задаётся в виде, описанном в разделе 3.3;

<TUN> – имя туннеля, задаётся в виде, описанном в разделе 3.4;

<SRC-ADDR> – IP-адрес отправителя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<DST-ADDR> – IP-адрес получателя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<DST-IPV6-ADDR> – IPv6-адрес получателя, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<SRC-IPV6-ADDR> – IPv6-адрес отправителя, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<SRC-PORT> – номер TCP/UDP-порта отправителя, принимает значения [1..65535];

<DST-PORT> – номер TCP/UDP-порта получателя, принимает значения [1..65535];

<VALUE> - количество пакетов, после получения которых анализ будет остановлен, указывается в диапазоне [1...4294967295];

detailed – информация выдается в детализированном формате.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

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

4.15 clear ssh host

Данной командой осуществляется сброс сохраненного SSH ключа удаленного хоста.

Синтаксис

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

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

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

5 УПРАВЛЕНИЕ ПРОГРАММНЫМ ОБЕСПЕЧЕНИЕМ И КОНФИГУРАЦИЕЙ

5.1 *commit*

Данная команда позволяет применить (сделать действующими) изменения конфигурации. RUNNING-конфигурация замещается конфигурацией CANDIDATE. Для того чтобы примененные изменения стали постоянно действующими, эту операцию необходимо подтвердить командой «confirm» в течение времени, не превышающего время действия таймера подтверждения (по умолчанию 600 секунд).

Синтаксис

commit

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
esr# commit
```

Изменения конфигурации применены.

5.2 *commit update*

Данная команда позволяет переприменить RUNNING-конфигурацию.

Синтаксис

commit update

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
esr# commit update
```

Изменения конфигурации применены.

5.3 confirm

Команда предназначена для подтверждения применения конфигурации. Если в течение заданного времени (по умолчанию 600 секунд) после применения конфигурации командой «commit» не было введено подтверждение, произойдет автоматический откат на действующую ранее конфигурацию. Автоматическая система откатов полностью предотвращает ситуации потери связи с устройством.

Синтаксис команды

confirm

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
esr# confirm
```

Подтверждение изменений в конфигурации.

5.4 restore

Данная команда позволяет отменить примененную, но неподтвержденную конфигурацию и вернуться к последней подтвержденной. Команда применяется ко всей конфигурации устройства. Отмена изменений может быть выполнена только до ввода команды «confirm». При выполнении команды «restore» происходит потеря неподтвержденной конфигурации.

Синтаксис

restore

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
esr# restore
```

Осуществлен возврат к последней подтвержденной конфигурации.

5.5 rollback

Данная команда позволяет отменить непримененные изменения конфигурации. В результате выполнения команды будет удалена CANDIDATE конфигурация. Команда может быть использована только до ввода команды «commit».

Команда применяется ко всей конфигурации устройства.

Синтаксис

rollback

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
esr# rollback
```

Произведена отмена всех непримененных изменений в конфигурации.

5.6 save

Команда служит для сохранения CANDIDATE конфигурации в постоянную память устройства.

Синтаксис

save

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
esr# save
```

Сохранение текущей конфигурации на Flash-память устройства.

5.7 boot system

Данная команда служит для выбора активного образа программного обеспечения, загруженного на устройство.

Синтаксис

boot system <IMAGE>

Параметры

<IMAGE> – название образа программного обеспечения, который будет загружаться на устройство:

image-1 – следующая загрузка устройства будет выполнена из первого образа ПО;

image-2 – следующая загрузка устройства будет выполнена из второго образа ПО.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

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

5.8 delete

Данная команда служит для удаления лицензий, сертификатов, ключей.

Синтаксис

delete <FILE>

Параметры

<FILE> - тип файла, может принимать следующие значения (при удалении из папки необходимо указать название файла):

fs://certificate/ca/ - папка для хранения сертификата удостоверяющего центра;

fs://certificate/dh/ - папка для хранения ключа Диффи-Хелмена;

fs://certificate/server_key/ - папка для хранения приватного ключа OpenVPN сервера;

fs://certificate/server_cert/ - папка для хранения публичного сертификата OpenVPN сервера;

fs://certificate/ta/ - папка для хранения HMAC ключа;

fs://certificate/crl/ - папка для хранения списка отозванных сертификатов.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

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

5.9 show candidate-config

Данной командой осуществляется просмотр конфигурации устройства, которая будет установлена после применения настроек (команда «commit»).

Синтаксис

show candidate-config [<SECTION>]

Параметры

<SECTION> – раздел конфигурации:

aaa – настройка параметров аутентификации, авторизации и учета;

access-list – конфигурация списков доступа;

bridges – конфигурация сетевых мостов;

channel-group – конфигурация группы агрегации каналов;

clock – конфигурация системных часов маршрутизатора и NTP-протокола;

dhcp – конфигурация DHCP-сервера, клиента и Relay-агента;

dual-homing – конфигурация сервиса Dual Homing;

hostname – сетевое имя маршрутизатора;

interfaces [<IF>] – конфигурация интерфейсов:

- <IF> – наименование интерфейса, задаётся в виде, описанном в разделе 3.3;

ip-address – конфигурация IP-интерфейсов;

ipv6 [<SUBSECTION>] – конфигурация IPv6:

- address – конфигурация IPv6-интерфейсов;
- dhcp [<SUBSECTION>] – конфигурация IPv6 DHCP-сервисов:
 - client – конфигурация IPv6 DHCP-сервера;
 - relay – конфигурация IPv6 DHCP Relay-агента;
 - server – конфигурация IPv6 DHCP-клиента;
- routing [<SUBSECTION>] – конфигурация IPv6-маршрутизации:
 - bgp – конфигурация протокола IPv6 BGP;
 - ospf – конфигурация протокола OSPFv3;
 - prefix-lists – конфигурация IPv6 префикс-листов;
 - static – конфигурация статических маршрутов;
- vrrp – конфигурация IPv6 VRRP-протокола

mac-address-table – конфигурация таблицы MAC-адресов;

mirroring – конфигурация зеркалирования;

multiwan – конфигурация сервиса резервирования и балансировки WAN-интерфейсов;

nat [<SUBSECTION>] – конфигурация сервиса NAT:

- source – конфигурация сервиса Source NAT;
- destination – конфигурация сервиса Destination NAT;

netflow – конфигурация Netflow протокола;

object-groups – конфигурация профилей IP-адресов и TCP/UDP-портов;

port-security – конфигурация Port Security;

qos – конфигурация QoS;

remote-access [<SUBSECTION>] – конфигурация профилей L2TP over IPsec и PPTP-серверов;

- l2tp – конфигурация профилей L2TP over IPsec-серверов;
- pptp – конфигурация профилей PPTP-серверов;
- openvpn – конфигурация профилей OpenVPN-серверов;

remote-client – конфигурация удаленного доступа (SSH, Telnet, etc.);

routing [<SUBSECTION>] – конфигурация маршрутизации:

- bgp – конфигурация протокола BGP;
- key-chains – конфигурация ключей аутентификации;
- ospf – конфигурация протокола OSPF;
- prefix-list – конфигурация префикс-листов;
- rip – конфигурация протокола RIP;
- route-maps – конфигурация маршрутных карт;
- static – конфигурация статических маршрутов;

security [<SUBSECTION>] – конфигурация сервисов IPsec VPN и Firewall;

- ike – конфигурация IKE;
- ipsec – конфигурация IPSec;
- zone – конфигурация зон Firewall;
- zone-pair – конфигурация переходов между зонами Firewall;

sflow – конфигурация sFlow протокола;

snmp – конфигурация SNMP-сервера;

spanning-tree – конфигурация протоколов семейства Spanning Tree;

syslog – конфигурация сервиса Syslog;

system – конфигурация общесистемных параметров;

tunnels [<SUBSECTION>] – конфигурация туннелей:

- gre – конфигурация GRE-туннелей;
- ip4ip4 – конфигурация IPv4 over IPv4-туннелей;
- l2tpv3 – конфигурация L2TPv3-туннелей;

- softgre – конфигурация SoftGRE-туннелей;
- vti – конфигурация VTI-туннелей;

vlan – конфигурация VLAN;

vrf – конфигурация VRF;

vrrp – конфигурация VRRP-протокола;

wisla – конфигурация системы мониторинга качества услуг wiSLA;

wireless-controller – конфигурация параметров Wi-Fi контроллера.

Командный режим

ROOT

Пример

```
esr# show candidate-config
service ntp enable
service ntp broadcast-client enable

syslog max-files 3
syslog file-size 512
syslog file default info
vlan 2
exit
security zone trusted
exit
security zone untrusted
exit

object-group service telnet
  port-range 23
exit
object-group service ssh
  port-range 22
exit
object-group service dhcp_server
  port-range 67
exit
More? Enter - next line; Space - next page; Q - quit; R - show the rest.
```

5.10 *show running-config*

Данная команда служит для просмотра текущей конфигурации устройства.

Синтаксис

show running-config [<SECTION>]

Параметры

<SECTION> – раздел конфигурации, описание приведено в разделе 5.9.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

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

5.11 show version

Данная команда служит для просмотра текущей версии программного обеспечения и аппаратной части устройства.

Синтаксис

show version

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

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

5.12 show bootvar

Данная команда служит для просмотра информации об образах программного обеспечения, загруженных на устройство.

Синтаксис

show bootvar

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show bootvar
```

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

5.13 show licence

Данная команда служит для просмотра информации об активной лицензии устройства.

Синтаксис

show licence

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

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

5.14 show crypto certificates

Данная команда выводит информацию о сертификатах.

Синтаксис

show crypto certificates [<CERTIFICATE-TYPE>]

Параметры

<CERTIFICATE-TYPE> - тип сертификата или ключа, может принимать следующие значения:

- ca – сертификат удостоверяющего сервера;
- crl – список отозванных сертификатов;
- dh – ключ Диффи-Хелмана;

- server-crt – публичный сертификат сервера;
- server-key – приватный ключ сервера;
- ta – HMAC ключ.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

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

5.15 copy

Данная команда служит для копирования файлов между различными источниками и получателями.

Синтаксис

copy <SOURCE> <DESTINATION>

Параметры

<SOURCE> – источник, задаётся в виде:

tftp://<ip>[<port>]:/<path> – адрес файла на TFTP-сервере, где:

- <ip> – IP-адрес TFTP-сервера;
- <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
- <path> – путь к файлу на TFTP-сервере.

tftp://<ipv6>[%<interface>][<port>]:/<path> – адрес файла на TFTP-сервере, где:

- <ipv6> – IPv6-адрес TFTP-сервера;
- <interface> – исходящий сетевой интерфейс для link-local адресов;
- <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
- <path> – путь к файлу на TFTP-сервере.

ftp://[<user>[:<password>]@]<ip>[<port>]:/<path>

- <ip> – IP-адрес FTP-сервера;
- <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой ip ftp client username, описанной в разделе 28.3.11);

- <password> – пароль (настроить пароль по умолчанию можно командой `ip ftp client password`, описанной в разделе 28.3.12);
- <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
- <path> – путь к файлу на FTP-сервере.

`ftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>`

- <ipv6> – IPv6-адрес FTP-сервера;
- <interface> – исходящий сетевой интерфейс для link-local адресов;
- <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой `ip ftp client username`, описанной в разделе 28.3.11);
- <password> – пароль (настроить пароль по умолчанию можно командой `ip ftp client password`, описанной в разделе 28.3.12);
- <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
- <path> – путь к файлу на FTP-сервере.

`scp://[<user>:<password>@]<ip>[<port>]:/<path>`

- <ip> – IP-адрес сервера;
- <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой `ip ssh client username`, описанной в разделе 28.3.4);
- <password> – пароль (настроить пароль по умолчанию можно командой `ip ssh client password`, описанной в разделе 28.3.5);
- <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
- <path> – путь к файлу на сервере.

`scp://[<user>:<password>@]<ipv6>[%<interface>][<port>]:/<path>`

- <ipv6> – IPv6-адрес сервера;
- <interface> – исходящий сетевой интерфейс для link-local адресов;
- <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой `ip ssh client username`, описанной в разделе 28.3.4);
- <password> – пароль (настроить пароль по умолчанию можно командой `ip ssh client password`, описанной в разделе 28.3.5);
- <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
- <path> – путь к файлу на сервере.

`fs://factory-config` – заводская конфигурация;

`fs://default-config` – конфигурация по умолчанию;

`fs://running-config` – текущая конфигурация;

`fs://candidate-config` – конфигурация, которая будет применена после выполнения команды «commit»;

`fs://firmware` – программное обеспечение устройства. Копирование производится с неактивного образа программного обеспечения устройства;

fs://boot – загрузчик устройства.

<DESTINATION> – назначение, задаётся в виде:

tftp://<ip>[<port>]:/<path> – адрес файла на TFTP-сервере, где:

- <ip> – IP-адрес TFTP-сервера;
- <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
- <path> – путь к файлу на TFTP-сервере.

tftp://<ipv6>[%<interface>][<port>]:/<path> – адрес файла на TFTP-сервере, где:

- <ipv6> – IPv6-адрес TFTP-сервера;
- <interface> – исходящий сетевой интерфейс для link-local адресов;
- <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
- <path> – путь к файлу на TFTP-сервере.

ftp://[<user>[:<password>]@]<ip>[<port>]:/<path>

- <ip> – IP-адрес FTP-сервера;
- <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой ip ftp client username, описанной в разделе 28.3.11);
- <password> – пароль (настроить пароль по умолчанию можно командой ip ftp client password, описанной в разделе 28.3.12);
- <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
- <path> – путь к файлу на FTP-сервере.

ftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>

- <ipv6> – IPv6-адрес FTP-сервера;
- <interface> – исходящий сетевой интерфейс для link-local адресов;
- <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой ip ftp client username, описанной в разделе 28.3.11);
- <password> – пароль (настроить пароль по умолчанию можно командой ip ftp client password, описанной в разделе 28.3.12);
- <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
- <path> – путь к файлу на FTP-сервере.

scp://[<user>:<password>@]<ip>[<port>]:/<path>

- <ip> – IP-адрес сервера;
- <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой ip ssh client username, описанной в разделе 28.3.4);
- <password> – пароль (настроить пароль по умолчанию можно командой ip ssh client password, описанной в разделе 28.3.5);
- <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
- <path> – путь к файлу на сервере.

scp://[<user>:<password>@]<ipv6>[%<interface>][<port>]:/<path>

- <ipv6> – IPv6-адрес сервера;
- <interface> – исходящий сетевой интерфейс для link-local адресов;
- <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой `ip ssh client username`, описанной в разделе 28.3.4);
- <password> – пароль (настроить пароль по умолчанию можно командой `ip ssh client password`, описанной в разделе 28.3.5);
- <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
- <path> – путь к файлу на сервере.

fs://candidate-config – конфигурация, которая будет применена после выполнения команды «commit»;

fs://licence – лицензия устройства;

fs://firmware – программное обеспечение устройства. Копирование всегда происходит в неактивный образ программного обеспечения устройства;

fs://boot – загрузчик устройства;

fs://certificate/ca/ - папка для хранения сертификата удостоверяющего центра;

fs://certificate/dh/ - папка для хранения ключа Диффи-Хелмена;

fs://certificate/server_key/ - папка для хранения приватного ключа OPENVPN сервера;

fs://certificate/server.crt/ - папка для хранения публичного сертификата OPENVPN сервера;

fs://certificate/ta/ - папка для хранения HMAC ключа;

fs://certificate/crl/ - папка для хранения списка отозванных сертификатов.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример 1

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

Пример 2

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

5.16 archive

Данной командой осуществляется переход в режим настройки параметров резервирования конфигурации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию для параметров резервирования конфигурации.

Синтаксис

[no] archive

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

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

5.17 path

Данной командой определяется протокол, адрес сервера, а также расположение и префикс имени файла на сервере. При выполнении резервирования к префиксу имени файла добавляется текущее время и дата в формате ГГГММДД_ЧЧММСС.

Использование отрицательной формы команды (no) удаляет установленное значение.

Синтаксис

path <PATH>

no path

Параметры

<PATH> – формат пути до удаленного сервера описан в разделе 5.14.

Необходимый уровень привилегий

15

Командный режим

CONFIG-ARCHIVE

Пример

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

5.18 by-commit

Данной командой включается режим отправки файла конфигурации на сервер резервирования после удачного применения конфигурации.

Использование отрицательной формы команды (no) выключает режим отправки после удачного применения конфигурации.

Синтаксис

[no] by-commit

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-ARCHIVE

Пример

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

5.19 auto

Данной командой включается режим отправки файла конфигурации на сервер резервирования через указанный промежуток времени (раздел 5.20).

Использование отрицательной формы команды (no) выключает режим отправки через указанный промежуток времени.

Синтаксис

[no] auto

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-ARCHIVE

Пример

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

5.20 time-period

Данной командой задаётся период времени, по истечении которого будет осуществляться автоматическое резервирование конфигурации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

time-period <TIME>

no time-period

Параметры

<TIME> – периодичность автоматического резервирования конфигурации, принимает значение в минутах [1..35791394].

Значение по умолчанию

720 минут

Необходимый уровень привилегий

15

Командный режим

CONFIG-ARCHIVE

Пример

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

6 НАСТРОЙКА ОБЩЕСИСТЕМНЫХ ПАРАМЕТРОВ

6.1 *hostname*

Команда позволяет назначить сетевое имя для маршрутизатора.

Синтаксис

hostname <NAME>

Параметры

<NAME> – сетевое имя маршрутизатора, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

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

6.2 *system fan-speed*

Команда определяет режим работы системы охлаждения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

[no] system fan-speed { auto | max }.

Параметры

- auto - режим автоматического регулирования (по умолчанию на ESR-100, ESR-200);
- max - режим максимального охлаждения (по умолчанию на ESR-1000).

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

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

6.3 *show system*

Данной командой осуществляется просмотр параметров окружения устройства.

Синтаксис

show system

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show system
System type:      Eltex ESR-10 Service Router
System name:      esr-1000
Software version:  1.0.8 build 18[53264b8] (date 10/11/2016 time 10:23:23)
Hardware version:  1v1
System uptime:    4 minutes and 5 seconds
System MAC address: A8:F9:4B:AA:03:A0
System serial number: NP01000050

Main power supply installed: Present
Main power supply status:      Ok
Reserve power supply installed: Absent

Fan Level:          46%

  Fan Table
  ~~~~~
      Fan 1   Fan 2   Fan 3   Fan 4
  -----
Status  Ok     Ok     Ok     Ok

  Temperature Table
  ~~~~~
              CPU       Sensor 1   Sensor 2   Sensor 3
  -----
Temperature, C  63       39         37         49

  Memory Table
  ~~~~~
      Total, MB       Used, MB       Free, MB
  -----
RAM      3798.25      1643.50 (44%)    2154.75 (56%)
FLASH   20.00         1.06 (6%)       18.94 (94%)
```

6.4 show system id

Данной командой осуществляется просмотр серийного номера устройства.

Синтаксис

show system id

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

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

6.5 show cpu network-load

Данной командой осуществляется просмотр нагрузки, производимой сетевым трафиком.

Синтаксис

show cpu network-load

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show cpu network-load
CPU ID      CPU load      Heaviest session      Session
-----      -
0           0             --                     0
1          49413      80.88.157.57 ->      9826
                        172.129.22.57
```

6.6 show cpu utilization

Данной командой осуществляется просмотр использования ресурсов CPU.

Синтаксис

show cpu utilization

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

Командный режим

ROOT

Пример

esr# show cpu utilization			
CPU	Last	Last	Last
	5 sec	1 min	5 min
---	-----	-----	-----
0	1.98%	6.75%	20.02%
1	67.50%	15.62%	6.88%

7 УПРАВЛЕНИЕ СИСТЕМНЫМИ ЧАСАМИ

7.1 *set date*

Данной командой устанавливается вручную системное время и дата.

Синтаксис

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

Параметры

<TIME> – устанавливаемое системное время, задаётся в виде HH:MM:SS, где:

HH – часы, принимает значение [0..23];

MM – минуты, принимает значение [0 .. 59];

SS – секунды, принимает значение [0 .. 59].

<DAY> – день месяца, принимает значения [1..31];

<MONTH> – месяц, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];

<YEAR> – год, принимает значения [2001..2037].

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

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

7.2 *clock timezone*

Данной командой устанавливается часовой пояс.

Использование отрицательной формы команды (no) устанавливает часовой пояс по умолчанию.

Синтаксис

clock timezone <OFFSET>

no clock timezone

Параметры

<OFFSET> – обозначение зоны, содержащее сдвиг в часах относительно Greenwich Mean Time, принимает значения [gmt -12 .. gmt +12].

Значение по умолчанию

gmt 0

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

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

7.3 show date

Данная команда позволяет посмотреть текущие системное время и дату.

Синтаксис

show date

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

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

7.4 ntp enable

Данной командой включается синхронизация системных часов с удаленными серверами по протоколу NTP.

Использование отрицательной формы команды (no) выключает синхронизацию по протоколу NTP.

Синтаксис

```
[no] ntp enable [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать синхронизация системных часов с удаленными серверами по протоколу NTP.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

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

7.5 ntp broadcast-client enable

Данной командой включается режим приёма широковещательных сообщений NTP-серверов. Маршрутизатор работает в качестве NTP-клиента. Если в конфигурации устройства заданы NTP пиры и серверы, то в широковещательном режиме они игнорируются.

Использование отрицательной формы команды (no) выключает широковещательный режим.

Синтаксис

```
[no] ntp broadcast-client enable [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет включен режим приёма широковещательных сообщений NTP-серверов.

Значение по умолчанию

Выключен

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

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

7.6 ntp dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов NTP-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

```
ntp dscp <DSCP>
```

```
no ntp dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

46

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

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

7.7 ntp peer

Данная команда используется для установления партнерских отношений между NTP-серверами и перехода в командный режим CONFIG-NTP.

NTP-сервер на маршрутизаторе работает в режиме двусторонней активности с удаленным NTP-сервером, указанным в команде. В случае потери связи одного из партнеров с вышестоящим NTP-сервером, он сможет синхронизировать время по серверу-партнеру.

Использование отрицательной формы команды (no) удаляет заданного NTP-партнера.

Синтаксис

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

Параметры

<ADDR> – IP-адрес партнера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес партнера, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать NTP-партнер.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

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

7.8 ntp server

Данная команда используется для создания NTP-сервера и перехода в командный режим CONFIG-NTP.

Маршрутизатор работает с указанным NTP-сервером в режиме односторонней активности. В данном режиме локальные часы маршрутизатора могут синхронизироваться с удаленным NTP-сервером.

Использование отрицательной формы команды (no) удаляет заданный NTP-сервер.

Синтаксис

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

Параметры

<ADDR> – IP-адрес сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPv6-ADDR> – IPv6-адрес сервера, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать NTP-сервер.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

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

7.9 maxpoll

Данная команда устанавливает максимальное значение интервала времени между отправкой сообщений NTP-серверу.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

maxpoll <INTERVAL>

no maxpoll

Параметры

<INTERVAL> – максимальное значение интервала опроса. Параметр команды используется как показатель степени двойки при вычислении длительности интервала в секундах, вычисляется путем возведения двойки в степень, заданную параметром команды, принимает значение [10..17].

Значение по умолчанию

10 (2^{10} = 1024 секунды или 17 минут 4 секунды)

Необходимый уровень привилегий

10

Командный режим

CONFIG-NTP

Пример

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

7.10 minpoll

Данная команда устанавливает минимальное значение интервала времени между отправкой сообщений NTP-серверу.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

minpoll <INTERVAL>

no minpoll

Параметры

<INTERVAL> – минимальное значение интервала опроса в секундах, вычисляется путем возведения двойки в степень, заданную параметром команды, принимает значение [4..6].

Значение по умолчанию

6 ($2^6 = 64$ секунды или 1 минута 4 секунды)

Необходимый уровень привилегий

10

Командный режим

CONFIG-NTP

Пример

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

7.11 prefer

Данная команда отмечает данный NTP-сервер как предпочтительный. При прочих равных условиях данный NTP-сервер будет выбран для синхронизации среди всех рабочих NTP-серверов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

[no] prefer

Параметры

Команда не содержит аргументов.

Значение по умолчанию

Выключено

Необходимый уровень привилегий

10

Командный режим

CONFIG-NTP

Пример

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

7.12 version

Данной командой устанавливается версия NTP-протокола.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

version <VERSION>

no version

Параметры

<VERSION> – версия NTP-протокола, принимает значения [1..4].

Значение по умолчанию

4

Необходимый уровень привилегий

10

Командный режим

CONFIG-NTP

Пример

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

7.13 show ntp configuration

Данная команда отображает действующую (RUNNING) конфигурацию протокола NTP.

Синтаксис

show ntp configuration [vrf <VRF>]

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будет отображена конфигурация протокола NTP в указанном VRF.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show ntp configuration
NTP status: Enabled
NTP mode:   client/server
```

Address	Type	Prefer	NTP version	Min poll	Max poll
10.100.100.1	peer	yes	NTPv4	6 (1m 4s)	11 (34m 8s)
10.100.100.2	peer	yes	NTPv4	6 (1m 4s)	10 (17m 4s)
1.2.3.4	server	no	NTPv4	6 (1m 4s)	10 (17m 4s)

7.14 show ntp peers

Данная команда позволяет посмотреть текущее состояние NTP-серверов (пиров). В следующих таблицах приведен перечень отображаемых параметров и их описание.

Таблица 7.1 – Состояние удаленного сервера (пира)

Параметр	Описание
remote	DNS-имя или IP-адрес сервера (пира). Первый символ в таблице используется для обозначения состояния сервера (пира), состояния описаны в таблице 7.2.
refid	Идентификатор связи или IP-адрес того, с кем синхронизирован удаленный сервер (пир). Типы идентификаторов связи описаны в таблице 7.3.
st	Статус.
t	Отношение маршрутизатора к удаленному серверу (пиру), типы описаны в таблице 7.4.
when	Период времени с момента, когда сервер (пир) последний раз опрашивался, в секундах (“h” часы, “d” дни).
poll	Частота опроса сервера (пира).
reach	Восьмибитный сдвигаемый влево регистр, содержащий результаты опросов (1 = успешно, 0 = неуспешно), отображается в восьмеричной системе счисления.
delay	Время прохождения пакета до сервера (пира) и обратно, в миллисекундах.
offset	Среднее постоянное смещение времени маршрутизатора относительно сервера (пира).
jitter	Средний разброс отклонения времени (джиттер).

Таблица 7.2 – Состояние удаленного сервера (пира)

Тип	Описание
пробел	Указывает на то, что: - не было ответов от удаленного сервера (пира); - сервер не используется, так как статус имеет большое значение; - сервер (пир) использует данный маршрутизатор для синхронизации своих часов.
x	Сервер (пир) не используется для синхронизации времени, отброшен алгоритмом пересечения.
-	Сервер (пир) не используется для синхронизации времени, отброшен кластерным алгоритмом.
#	Рабочий удаленный сервер (пир), но не используется, так как не вошел в число первых шести серверов (пиров), отсортированных по расстоянию синхронизации, является резервным.
+	Рабочий и предпочитаемый удаленный сервер (пир), включен алгоритмом объединения.
*	Сервер (пир), который в настоящее время является первичным источником времени.

Таблица 7.3 – Типы идентификаторов соединения с удаленным сервером (пиром)

Тип	Описание
.ACST.	Manycast-сервер.
.AUTH.	Ошибка аутентификации.
.AUTO.	Ошибка последовательности автоматического ключа.
.BCST.	Broadcast-сервер.
.CRYPT.	Ошибка протокола автоматического ключа.
.DENY.	Сервер отказал в доступе.
.INIT.	Инициализация соединения с сервером.
.MCST.	Multicast-сервер.
.TIME.	Таймаут соединения с сервером.
.STEP.	Ступенчатое изменение времени, смещение меньше предельного порога (1000 миллисекунд), но больше, чем шаг порога (125 миллисекунд).
.RATE.	Превышение частоты опросов.

Таблица 7.4 – Типы отношений маршрутизатора к удаленному серверу (пиру)

Тип	Описание
u	Unicast или manycast-клиент.
b	Broadcast или unicast-клиент.
s	Двусторонняя связь (пир).
A	Manycast-сервер.
B	Broadcast-сервер.
M	Multicast-сервер.

Синтаксис

```
show ntp peers [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будет отображено текущее состояние NTP-серверов (пиров) в указанном VRF.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

esr# show ntp peers									
remote	refid	st	t	when	poll	reach	delay	offset	jitter
-----	-----	-	-	-	-	-	-	-	-
1.2.3.4	.INIT.	16	-	-	64	0	0.000	0.000	0.000
*10.100.100.1	46.8.40.31	2	u	2	16	377	0.135	2.859	0.293
10.100.100.2	.INIT.	16	u	-	256	0	0.000	0.000	0.000

8 НАСТРОЙКА AAA

8.1 *username*

Данной командой выполняется добавление пользователя в локальную базу пользователей и осуществляется переход в режим настройки параметров пользователя.

Использование отрицательной формы команды (no) удаляет пользователя из системы.

Синтаксис

```
[no] username <NAME>
```

Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа. Если использовать команду для удаления, то при указании значения «all» будут удалены все пользователи.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
esr(config)# username test
esr(config-user)#
```

Добавлен пользователь с именем test.

8.2 *password*

Команда для установки пароля определённому пользователю для входа в систему. Пароль может быть задан как в открытом виде, так и в виде хеш sha512.

Использование отрицательной формы команды (no) удаляет пароль пользователя из системы.

Синтаксис

```
password { <CLEAR-TEXT> | encrypted <HASH_SHA512> }
no password
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой [8 .. 31] символов, принимает значения [0-9a-fA-F];

<HASH_SHA512> – хеш пароля по алгоритму sha512, задаётся строкой из 110 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-USER

Пример

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

8.3 privilege

Данной командой производится установка уровня привилегий пользователя. Набор команд, который доступен пользователю, зависит от уровня привилегий. Пользователям с уровнями привилегий от 1 до 9 доступен только просмотр информации. Пользователям с уровнем привилегий от 10 до 15 доступна большая часть команд конфигурирования. Пользователям с уровнем привилегий 15 доступен полный набор команд. Требуемый необходимый уровень привилегий команд может быть изменен, описание в разделе 8.37.

Использование отрицательной формы команды (no) устанавливает необходимый уровень привилегий по умолчанию.

Назначение начального уровня привилегий пользователям происходит следующим образом:

- необходимый уровень привилегий пользователям из локальной базы назначается указанной командой;
- необходимый уровень привилегий для пользователей, авторизовавшихся по протоколу RADIUS, берется из атрибута cisco-avpair = "shell:priv-lvl=<PRIV>";
- необходимый уровень привилегий для пользователей, авторизовавшихся по протоколу TACACS, берется из атрибута priv-lvl=<PRIV>;
- уровень привилегии для пользователей авторизовавшихся по протоколу LDAP берется из атрибута заданного командой **privilege-level-attribute**, описанной в разделе 8.29, по умолчанию priv-lvl=<PRIV>;

Если при аутентификации пользователя через протоколы TACACS/RADIUS/LDAP не была получена вышеуказанная опция или была получена опция с некорректным значением, то пользователю будут назначены привилегии пользователя «remote», по умолчанию 1. Необходимый уровень привилегий пользователя «remote» можно изменить аналогично любому другому пользователю из локальной базы с помощью указанной команды.

Синтаксис

```
privilege <PRIV>
```

```
no privilege
```

Параметры

<PRIV> – необходимый уровень привилегий, принимает значение [1..15].

Значение по умолчанию

1

Необходимый уровень привилегий

15

Командный режим

CONFIG-USER

Пример

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

8.4 enable

Данной командой производится повышение уровня привилегий пользователя. Способы аутентификации повышения привилегий пользователей задаются с помощью команды, описанной в разделе 8.8.



По умолчанию в конфигурации установлен метод аутентификации по паролю «enable». При этом пароли не заданы, то есть любой системный пользователь может получить 15 необходимый уровень привилегий.

Для аутентификации повышения привилегий по протоколам TACACS/RADIUS/LDAP на сервере должны быть созданы пользователи \$enab<PRIV>\$, где <PRIV> – необходимый уровень привилегий пользователя, который должен быть аутентифицирован.

Синтаксис

```
enable [ <PRIV> ]
```

Параметры

<PRIV> – необходимый уровень привилегий, принимает значение [2..15], значение по умолчанию 15.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

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

8.5 disable

Данной командой производится понижение уровня привилегий пользователя до первоначальных.

Синтаксис

```
disable
```

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# disable
esr>
```

8.6 enable password

Данной командой устанавливается пароль, который будет запрашиваться при повышении уровня привилегий пользователя.



По умолчанию в конфигурации пароли не заданы, то есть любой системный пользователь может получить 15 необходимый уровень привилегий.

Использование отрицательной формы команды (no) удаляет пароль из системы.

Синтаксис

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

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой [8 .. 31] символов, принимает значения [0-9a-fA-F];

<HASH_SHA512> – хеш пароля по алгоритму sha512, задаётся строкой из 110 символов;

<PRIV> – необходимый уровень привилегий, принимает значение [2..15], значение по умолчанию 15.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.7 aaa authentication login

Данной командой создаются списки способов аутентификации входа пользователей в систему. При неудачной попытке аутентификации по одному способу происходит попытка аутентификации по следующему в списке.

В конфигурации по умолчанию существует список с именем «default», данный список содержит один способ аутентификации – «local». Чтобы использовать список для аутентификации входа пользователей, необходимо выполнить его активацию командой, описанной в разделе 8.12.

Использование отрицательной формы команды (no) удаляет список способов аутентификации.

Синтаксис

```
aaa authentication login { default | <NAME> } <METHOD 1> [ <METHOD 2> ] [ <METHOD 3> ]  
[ <METHOD 4> ]  
no aaa authentication login { default | <NAME> }
```

Параметры

<NAME> – имя списка, задаётся строкой до 31 символа;

Способы аутентификации:

local – аутентификация с помощью локальной базы пользователей;

tacacs – аутентификация по списку TACACS-серверов;

radius – аутентификация по списку RADIUS-серверов;

ldap – аутентификация по списку LDAP-серверов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.8 aaa authentication enable

Данной командой создаются списки способов аутентификации повышения привилегий пользователей. При неудачной попытке аутентификации по одному способу происходит попытка аутентификации по следующему в списке.

В конфигурации по умолчанию существует список с именем «default». Список «default» содержит один способ аутентификации – «enable». Чтобы использовать список для аутентификации повышения привилегий пользователей, необходимо выполнить его привязку командой, описанной в разделе 8.13.

Использование отрицательной формы команды (no) удаляет список способов аутентификации.

Синтаксис

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

Параметры

<NAME> – имя списка:

строка до 31 символа;

default – имя списка «default».

<METHOD> – способы аутентификации:

enable – аутентификация с помощью enable-паролей;

tacacs – аутентификация по протоколу TACACS;

radius – аутентификация по протоколу RADIUS;

ldap – аутентификация по протоколу LDAP.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.9 aaa accounting commands

Данной командой конфигурируется список способов учета команд, введенных в CLI.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

aaa accounting commands stop-only <METHOD>

no aaa accounting commands stop-only

Параметры

<METHOD> – способы учета:

tacacs – учет введенных команд по протоколу TACACS.

Значение по умолчанию

Учет не ведется

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.10 aaa accounting login

Данной командой конфигурируется список способов учета сессий пользователей. Ведение учета активируется и прекращается, когда пользователь входит и отключается от системы, что соответствует значениям «start» и «stop» в сообщениях протоколов RADIUS и TACACS.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

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

no aaa accounting login start-stop

Параметры

<METHOD> – способы учета:

tacacs – учет сессий по протоколу TACACS;

radius – учет сессий по протоколу RADIUS.

Значение по умолчанию

Учет сессий ведется в локальный журнал

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.11 line

Данной командой осуществляется переход в режим конфигурирования соответствующего терминала: локальная консоль, удаленная консоль (Telnet), удаленная защищенная консоль (SSH).

Использование отрицательной формы команды (no) устанавливает параметры терминала по умолчанию. Параметры по умолчанию описаны в разделах 8.12 и 8.13.

Синтаксис

[no] line <TYPE>

Параметры

<TYPE> – тип консоли:

console – локальная консоль;

telnet – удаленная консоль;

ssh – защищенная удаленная консоль;

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.12 login authentication

Данной командой осуществляется активация списка аутентификации входа пользователей в систему, который будет использоваться в конфигурируемом терминале.

В конфигурации по умолчанию активным является список с именем «default», данный список содержит один способ аутентификации – «local».

Использование отрицательной формы команды (no) делает список с именем «default» активным.

Синтаксис

login authentication <NAME>

no login authentication

Параметры

<NAME> – имя списка, задаётся строкой до 31 символа.

Значение по умолчанию

default

Необходимый уровень привилегий

15

Командный режим

CONFIG-LINE-CONSOLE

CONFIG-LINE-TELNET

CONFIG-LINE-SSH

Пример

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

8.13 enable authentication

Данной командой осуществляется активация списка аутентификации повышения привилегий пользователей, который будет использоваться в конфигурируемом терминале.

В конфигурации по умолчанию активным является список с именем «default», данный список содержит один способ аутентификации – «enable».

Использование отрицательной формы команды (no) делает список с именем «default» активным.

Синтаксис

enable authentication <NAME>

no enable authentication

Параметры

<NAME> – имя списка, задаётся строкой до 31 символа.

Значение по умолчанию

default

Необходимый уровень привилегий

15

Командный режим

CONFIG-LINE-CONSOLE

CONFIG-LINE-TELNET

CONFIG-LINE-SSH

Пример

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

8.14 tacacs-server timeout

Данной командой задаётся интервал, по истечении которого устройство считает, что TACACS-сервер недоступен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
tacacs-server timeout <SEC>
```

```
no tacacs-server timeout
```

Параметры

<SEC> – период времени в секундах, принимает значения [1..30].

Значение по умолчанию

3 секунды

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.15 tacacs-server dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов TACACS-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

```
tacacs-server dscp <DSCP>
```

```
no tacacs-server dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

61

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

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

8.16 tacacs-server host

Данная команда используется для добавления TACACS-сервера в список используемых серверов и перехода в командный режим TACACS SERVER.

Использование отрицательной формы команды (no) удаляет заданный TACACS-сервер.

Синтаксис

```
[no] tacacs-server host { <ADDR> | <IPv6-ADDR> } [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

<ADDR> – IP-адрес TACACS -сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPv6-ADDR> – IPv6-адрес TACACS -сервера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.17 radius-server timeout

Данной командой задаётся интервал, по истечении которого устройство считает, что RADIUS-сервер недоступен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
radius-server timeout <SEC>
```

```
no radius-server timeout
```

Параметры

<SEC> – период времени в секундах, принимает значения [1..30].

Значение по умолчанию

3 секунды

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.18 radius-server retransmit

Данной командой задаётся количество перезапросов к последнему активному RADIUS-серверу, которое будет выполнено перед выполнением запросов к следующим RADIUS-серверам в списке.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

radius-server retransmit <COUNT>

no radius-server retransmit

Параметры

<COUNT> – количество перезапросов к RADIUS-серверу, принимает значения [1..10].

Значение по умолчанию

1

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.19 radius-server dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов RADIUS-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

radius-server dscp <DSCP>

no radius-server dscp

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

61

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

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

8.20 radius-server host

Данная команда используется для добавления RADIUS-сервера в список используемых серверов и перехода в командный режим RADIUS SERVER.

Использование отрицательной формы команды (no) удаляет заданный RADIUS-сервер.

Синтаксис

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

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

<ADDR> – IP-адрес RADIUS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес RADIUS-сервера, задаётся в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.21 ldap-server base-dn

Данной командой задаётся базовый DN (Distinguished name), который будет использоваться при поиске пользователей.

Использование отрицательной формы команды (no) удаляет заданный базовый DN.

Синтаксис

```
ldap-server base-dn <NAME>
```

```
no ldap-server base-dn
```

Параметры

<NAME> – базовый DN, задается строкой до 255 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.22 ldap-server bind timeout

Данной командой задаётся интервал, по истечении которого устройство считает, что LDAP-сервер недоступен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ldap-server bind timeout <SEC>

no ldap-server bind timeout

Параметры

<SEC> – период времени в секундах, принимает значения [1..30].

Значение по умолчанию

3 секунды

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.23 ldap-server bind authenticate root-dn

Данной командой задаётся DN (Distinguished name) пользователя с правами администратора, под которым будет происходить авторизация на LDAP-сервере при поиске пользователей.

Использование отрицательной формы команды (no) удаляет заданный DN пользователя.

Синтаксис

ldap-server bind authenticate root-dn <NAME>

no bind authenticate root-dn

Параметры

<NAME> – DN пользователя с правами администратора, задается строкой до 255 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.24 ldap-server bind authenticate root-password

Данной командой задаётся пароль пользователя с правами администратора, под которым будет происходить авторизация на LDAP-сервере при поиске пользователей.

Использование отрицательной формы команды (no) удаляет заданный пароль пользователя.

Синтаксис

```
ldap-server bind authenticate root-password ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> }  
no bind authenticate root-password
```

Параметры

<TEXT> – строка [8..16] ASCII-символов;

<ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задаётся строкой [16..32] символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.25 ldap-server search filter user-object-class

Данной командой задаётся имя класса объектов, среди которых необходимо выполнять поиск пользователей на LDAP-сервере.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ldap-server search filter user-object-class <NAME>  
no ldap-server search filter user-object-class
```

Параметры

<NAME> – имя класса объектов, задаётся строкой до 127 символов.

Значение по умолчанию

posixAccount

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.26 ldap-server search scope

Данной командой задаётся область поиска пользователей в дереве LDAP-сервера.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ldap-server search scope <SCOPE>

no ldap-server search scope

Параметры

<SCOPE> – область поиска пользователей на LDAP-сервере, принимает следующие значения:

onelevel – выполнять поиск в объектах на следующем уровне после базового DN в дереве LDAP-сервера;

subtree – выполнять поиск во всех объектах поддерева базового DN в дереве LDAP сервера.

Значение по умолчанию

subtree

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.27 ldap-server search timeout

Данной командой задаётся интервал, по истечении которого устройство считает, что LDAP-сервер не нашел записей пользователей, подходящих под условие поиска.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ldap-server search timeout <SEC>

no ldap-server search timeout

Параметры

<SEC> – период времени в секундах, принимает значения [0..30].

Значение по умолчанию

0 – устройство ожидает завершения поиска и получения ответа от LDAP-сервера.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.28 ldap-server naming-attribute

Данной командой задаётся имя атрибута объекта, со значением которого идет сравнение имени искомого пользователя на LDAP-сервере.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ldap-server naming-attribute <NAME>

no ldap-server naming-attribute

Параметры

<NAME> – имя атрибута объекта, задаётся строкой до 127 символов.

Значение по умолчанию

uid

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.29 ldap-server privilege-level-attribute

Данной командой задаётся имя атрибута объекта, значение которого будет определять начальные привилегии пользователя на устройстве. Атрибут должен принимать значения [1..15]. Если указанный атрибут отсутствует или содержит недопустимое значение, то начальные привилегии пользователя будут соответствовать привилегиям пользователя «remote».

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ldap-server privilege-level-attribute <NAME>

no ldap-server privilege-level-attribute

Параметры

<NAME> – имя атрибута объекта, задаётся строкой до 127 символов.

Значение по умолчанию

priv-lvl

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.30 *ldap-server dscp*

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов LDAP-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

ldap-server dscp <DSCP>

no ldap-server dscp

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

61

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

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

8.31 *ldap-server host*

Данная команда используется для добавления LDAP-сервера в список используемых серверов и перехода в командный режим LDAP SERVER.

Использование отрицательной формы команды (no) удаляет заданный LDAP-сервер.

Синтаксис

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

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

<ADDR> – IP-адрес LDAP-сервера, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес LDAP-сервера, задается в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

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

8.32 usage

Данная команда определяет тип соединений для аутентификации которых будет использоваться RADIUS-сервера.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

usage { all | aaa | pptp | l2tp }

no usage

Параметры

all – все типы соединений;

aaa – RADIUS-сервер будет использоваться для аутентификации и авторизации telnet, ssh console сессий;

pptp – RADIUS-сервер будет использоваться для аутентификации удаленных пользователей, подключающихся по протоколу PPTP;

l2tp – RADIUS-сервер будет использоваться для аутентификации удаленных пользователей, подключающихся по протоколу L2TP over IPsec.

Значение по умолчанию

all

Необходимый уровень привилегий

15

Командный режим

CONFIG-RADIUS-SERVER

Пример

```
esr(config-radius-server)# usage pptp
```

8.33 key

Данной командой задаётся пароль для аутентификации на удаленном сервере.

Использование отрицательной формы команды (no) удаляет заданный пароль для аутентификации на удаленном сервере.

Синтаксис

key ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> }

no key

Параметры

<TEXT> – строка [8..16] ASCII-символов;

<ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задаётся строкой [16..32] символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-TACACS-SERVER

CONFIG-RADIUS-SERVER

Пример

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

8.34 port

Данной командой задаётся номер порта для обмена данными с удалённым сервером.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

port <PORT>

no port

Параметры

<PORT> – номер TCP-порта для обмена данными с удалённым сервером, принимает значения [1..65535].

Значение по умолчанию

49 для TACACS-сервера

389 для LDAP-сервера

Необходимый уровень привилегий

15

Командный режим

CONFIG-TACACS-SERVER

CONFIG-LDAP-SERVER

Пример

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

8.35 priority

Данной командой задаётся приоритет использования удаленного сервера. Чем ниже значение, тем приоритетнее сервер.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

priority <PRIORITY>

no priority

Параметры

<PRIORITY> – приоритет использования удаленного сервера, принимает значения [1..65535].

Значение по умолчанию

1

Необходимый уровень привилегий

15

Командный режим

CONFIG-TACACS-SERVER

CONFIG-RADIUS-SERVER

CONFIG-LDAP-SERVER

Пример

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

8.36 timeout

Данной командой задаётся интервал, по истечении которого устройство считает, что RADIUS-сервер недоступен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

timeout <SEC>

no timeout

Параметры

<SEC> – период времени в секундах, принимает значения [1..30].

Значение по умолчанию

Не задан, используется значение глобального таймера, описанного в разделе 8.17.

Необходимый уровень привилегий

15

Командный режим

CONFIG-RADIUS-SERVER

Пример

```
esr(config-radius-server)# timeout 7
```

8.37 source-address

Данной командой определяется IP/IPv6-адрес маршрутизатора, который будет использоваться в качестве IP/IPv6-адреса источника в отправляемых RADIUS пакетах.

Использование отрицательной формы команды (no) удаляет указанный IP/IPv6-адрес источника.

Синтаксис

```
source-address { <ADDR> | <IPV6-ADDR> }
```

```
no source-address
```

Параметры

<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес источника, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

10

Командный режим

CONFIG-RADIUS-SERVER

Пример

```
esr(config-radius-server)# source-address 220::71
```

8.38 privilege

Данной командой производится установка минимального уровня привилегий пользователя, необходимого для выполнения команды из указанного поддерева команд.

Использование отрицательной формы команды (no) устанавливает необходимый уровень привилегий по умолчанию.

Синтаксис

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

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

Параметры

<COMMAND-MODE> – командный режим, описание режимов приведено в таблице 3.2;

<PRIV> – необходимый уровень привилегий поддерева команд, принимает значение [1..15];

<COMMAND> – поддерево команд, задается строкой до 255 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

Установить для поддрева команд «show» корневого командного режима необходимый уровень привилегий 2. Команды поддрева «show interfaces» оставить с уровнем привилегий 1.

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

8.39 tech-support login enable privilege

Данной командой включается низкоуровневый удаленный доступ к системе с помощью пользователя «techsupport». Низкоуровневый доступ к системе позволит получить технической поддержке всю необходимую информацию, когда это необходимо.

Использование отрицательной формы команды (no) выключает низкоуровневый удаленный доступ к системе с помощью пользователя «techsupport».

Синтаксис

[no] tech-support login enable

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# tech-support login enable
```

8.40 show users accounts

Данная команда позволяет просмотреть конфигурацию пользователей системы.

Синтаксис

show users accounts

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show user accounts  
Name                               Password                               Privilege  
-----
```

admin	\$6\$1sXrvGaV8Za8oX/K\$YNe15xYPZ4cj bemYWYNpQBQKDxWE9v0aoKgQ kRCEb0EMNuusO9Kmg7UBs7nA3buEM87e Eu.rA6tZq0	15
techsupport	\$6\$YfwntIwU\$ah7UxPZTemKhjpSWvVsV 9jHcp. 9lweQaSldw7ZtUr uH66uZx9.EBASff//hUj8ObUaC484TNR x.	15
remote	\$6\$YfwntIwU\$ah7UxPZTemKhjpSWvVsV 9jHcp.kqFAK.vmvY9lweQaSldw7ZtUr uH66uZx9.EBASff//hUj8ObUaC484TNR x.	1
operator	\$6\$eILpbbYRxedCzvVD\$4RHP08mjXvNf urX7V/ULCZ1oHIWMwE6h5f zgwZQUZcPoZCEyaqQQcicRMRuPwhxrQ bvGChWreWl	1

8.41 show users

Данная команда позволяет просмотреть активные сессии пользователей системы.

Синтаксис

show users

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show users
```

User name	Logged in at	Host	Timers Login/Priv	level
-----	-----	-----	-----	-----
admin	13/02/15 01:14:25	Console	00:29:57/00:00:00	15

1 user sessions.

8.42 show aaa authentication

Данная команда позволяет просмотреть списки способов аутентификации пользователей, а также активные списки каждого типа терминалов.

Синтаксис

show aaa authentication

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show aaa authentication

  Login Authentication Method Lists
  ~~~~~
List           Methods
-----
default        local

  Enable Authentication Method Lists
  ~~~~~
List           Methods
-----
default        enable

  Lines configuration
  ~~~~~
Line           Login method list           Enable method list
-----
console        default                     default
telnet         default                     default
ssh            default                     default
```

8.43 show aaa accounting

Данная команда позволяет просмотреть настроенные параметры учета.

Синтаксис

```
show aaa accounting
```

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show aaa accounting
Login :      radius
Commands :   tacacs
```

8.44 show aaa ldap-servers

Данная команда позволяет просмотреть параметры LDAP-серверов.

Синтаксис

```
show aaa ldap-servers
```

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show aaa ldap-servers
```

Base DN:	dc=example,dc=com	
Root DN:	cn=admin,dc=example,dc=com	
Root password:	CDE65039E5591FA3	
Naming attribute:	uid	
Privilege level attribute:	priv-lvl	
User object class:	posixAccount	
DSCP:	63	
Bind timeout:	3	
Search timeout:	0	
Search scope:	subtree	

IP Address	Port	Priority
-----	-----	-----
10.100.100.1	389	1

8.45 show aaa radius-servers

Данная команда позволяет просмотреть параметры RADIUS-серверов.

Синтаксис

show aaa radius-servers

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show aaa radius-servers
```

Timeout:	3
Retransmit:	1
DSCP:	63

IP Address	Timeout	Priority	Usage	Key
-----	-----	-----	-----	-----
2.2.2.2	--	1	all	9DA7076CA30B5FFE0DC9C4
2.4.4.4	--	1	all	9DA7076BA30B4EFCE5

8.46 show aaa tacacs-servers

Данная команда позволяет просмотреть параметры TACACS-серверов.

Синтаксис

show aaa tacacs-servers

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show aaa tacacs-servers
Timeout :      3
DSCP:         63
```

IP Address	Port	Priority	Key
10.100.100.1	49	1	CDE65039E5591FA3
10.100.100.5	49	10	CDE65039E5591FA3

9 НАСТРОЙКА И МОНИТОРИНГ ИНТЕРФЕЙСОВ

Порядок именования интерфейсов маршрутизатора описан в разделе 3.3.

Команды, введенные в режиме конфигурирования интерфейса (группы интерфейсов), применяются к выбранному интерфейсу (группе интерфейсов).

9.1 Ethernet-интерфейсы

9.1.1 *interface*

Данная команда позволяет перейти в режим конфигурирования одного или более интерфейсов.

Использование отрицательной формы команды (no) восстанавливает настройки интерфейса по умолчанию.

Синтаксис

[no] interface <IF>

Параметры

<IF> – наименование интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе 3.3.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример 1

Переход в режим конфигурирования Ethernet-интерфейса gi 1/0/2:

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

Пример 2

Переход в режим конфигурирования виртуального интерфейса:

```
esr(config)# interface loopback 5
esr(config-loopback) #
```

Пример 3

Переход в режим конфигурирования субинтерфейса:

```
esr(config)# interface gigabitethernet 1/0/2.20
esr(config-subif) #
```

Пример 4

Переход в режим конфигурирования интерфейса port-channel 2:

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

Пример 5

Переход в режим конфигурирования интерфейса e1 1/0/1:

```
esr(config)# interface e1 1/0/1
esr(config-e1)#
```

Пример 6

Переход в режим конфигурирования интерфейса multilink 1:

```
esr(config)# interface multilink 1
esr(config-multilink)#
```

9.1.2 *description*

Данная команда используется для изменения описания конфигурируемого интерфейса.

Использование отрицательной формы команды (no) удаляет установленное описание.

Синтаксис

description <DESCRIPTION>

no description

Параметры

<DESCRIPTION> – описание интерфейса, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-LOOPBACK

CONFIG-E1

CONFIG-MULTILINK

Пример

```
esr(config-if-gi)# description "Uplink interface"
```

9.1.3 *load-average*

Данной командой устанавливается интервал времени, в течение которого собирается статистика о нагрузке на интерфейс.

Использование отрицательной формы команды устанавливает значение по умолчанию.

Синтаксис

load-average <TIME>

no load-average

Параметры

<TIME> – интервал в секундах, принимает значения [5..300].

Значение по умолчанию

5 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-E1

CONFIG-MULTILINK

CONFIG-VTI

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-L2TPv3

CONFIG-BRIDGE

CONFIG-LOOPBACK

Пример

```
esr(config-if-gi)# load-average 30
```

9.1.4 *speed*

Данной командой устанавливается значение скорости для конфигурируемого интерфейса, группы интерфейсов. Командой могут быть установлены следующие режимы: 10Мбит/с, 100Мбит/с, 1000 Мбит/с, 10Гбит/с или auto.

Использование отрицательной (no) формы команды устанавливает значение по умолчанию.

Синтаксис

speed <SPEED> <DUPLEX>

no speed

Параметры

<SPEED> – значение скорости:

10M – значение скорости 10Мбит/с;

100M – значение скорости 100Мбит/с;

1000M – значение скорости 1000Мбит/с;

auto – автоматический выбор режима.

<DUPLEX> – режим работы приемопередатчика, принимает значения:

full-duplex – дуплекс;

half-duplex – полудуплекс.

Значение по умолчанию

auto

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

CONFIG-PORT-CHANNEL

Пример 1

```
esr(config-if-gi)# speed 10M full-duplex
```

Установлен скоростной режим интерфейса 10 Мбит/с, дуплекс.

9.1.5 *system jumbo-frames*

Данной командой включается поддержка Jumbo-фреймов.

Использование отрицательной формы команды (no) выключает поддержку Jumbo-фреймов.

Синтаксис

[no] system jumbo-frames

Параметры

Команда не содержит аргументов.

Значение по умолчанию

Выключена

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# system jumbo-frames
```

9.1.6 *mtu*

Данной командой указывается размер MTU (Maximum Transmission Unit) для физических интерфейсов.

Использование отрицательной формы команды (no) устанавливает значение MTU по умолчанию.

Синтаксис

mtu <MTU>

no mtu

Параметры

<MTU> – значение MTU, принимает значения в диапазоне [128..10000]. Значения MTU более 1500 можно выставить только при включенной поддержке Jumbo-фреймов, описанной в разделе 9.1.5.

Значение по умолчанию

1500

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-E1

Пример

```
esr(config-if-gi)# mtu 1400
```

9.1.7 *shutdown*

Данной командой отключается конфигурируемый интерфейс.

Использование отрицательной формы команды (no) включает конфигурируемый интерфейс.

Синтаксис

[no] shutdown

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-LOOPBACK

CONFIG-E1

Пример

```
esr(config-if-gi)# shutdown
```

Конфигурируемый интерфейс отключен.

9.1.8 *switchport mode*

Данная команда используется для задания режима работы интерфейса с VLAN.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

switchport mode <MODE>

no switchport mode

Параметры

<MODE> – режим работы:

- access¹ – интерфейс доступа, нетегированный интерфейс для одной VLAN;
- trunk – интерфейс, принимающий только тегированный трафик за исключением одного VLAN, который может быть добавлен с помощью команды **switchport trunk native vlan**, описанной в 12.2.7;
- general² – физический интерфейс переключается в режим general;
- e1 – физический интерфейс переключается в режим E1.

Значение по умолчанию

access

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

Пример

```
esr-100(config-if-gi)# switchport mode trunk
```

9.1.9 *show interfaces switch-port status*

Команда используется для просмотра состояния физических интерфейсов.

Синтаксис

show interfaces switch-port status [<IF>]

Параметры

<IF> – имя физического интерфейса устройства, задаётся в виде, описанном в разделе 3.3.

Можно указать несколько интерфейсов перечислением через запятую «,» либо указать диапазон интерфейсов через дефис «-». Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы. Если задан определенный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех физических интерфейсов.

Необходимый уровень привилегий

1

¹ В текущей версии ПО данная команда поддерживается только на маршрутизаторах ESR-100/ESR-200

² В текущей версии ПО данная команда поддерживается только на маршрутизаторе ESR-1000

Командный режим

ROOT

Пример

```
esr# show interfaces switch-port status gigabitethernet 1/0/1-6
Port          Media      Duplex    Speed      Neg        Flow      Link      Back
              control   State     Pressure
-----
gil/0/1       N/A       N/A       N/A        N/A        N/A       Down     N/A
gil/0/2       copper    Full      100 Mbps   Enabled    Off       Up       Disabled
gil/0/3       N/A       N/A       N/A        N/A        N/A       Down     N/A
gil/0/4       N/A       N/A       N/A        N/A        N/A       Down     N/A
gil/0/5       N/A       N/A       N/A        N/A        N/A       Down     N/A
gil/0/6       N/A       N/A       N/A        N/A        N/A       Down     N/A

esr# show interfaces switch-port status gigabitethernet 1/0/2
Interface      gigabitethernet 1/0/2
Status:        up
Media:         copper
Speed:         100 Mbps
Duplex:        full
Flow control:  no
MAC address:   a8:f9:b5:00:00:25
MAC status:
Buffers full:  no
Doing back pressure: no
Sending PAUSE frames: no
Receiving PAUSE frames: no
Auto-Negotiation done: yes
Sync fail:     no
```

9.1.10 *show interfaces utilization*

Команда используется для просмотра текущей нагрузки на физических интерфейсах.

Синтаксис

show interfaces utilization [<IF>]

Параметры

<IF> – имя физического интерфейса устройства, задаётся в виде, описанном в разделе 3.3.

Можно указать несколько интерфейсов перечислением через запятую «,» либо указать диапазон интерфейсов через дефис «-». Если не указывать индексы интерфейсов, то будет отображена текущая нагрузка для всех интерфейсов заданной группы. При выполнении команды без параметра будет показана текущая нагрузка для всех физических интерфейсов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show interfaces utilization gigabitethernet 1/0/3-5
Port          Period, s      Sent,      Recv,      Frames Sent  Frames Recv
              Kbit/s       Kbit/s
-----
gil/0/3       5             0          0          0            0
```

gil/0/4	5	0	0	0	0
gil/0/5	5	0	0	0	0

9.1.11 *show interfaces counters*

Командой выполняется просмотр счетчиков на системных интерфейсах – портах, субинтерфейсах, группах агрегации, сетевых мостах.

Синтаксис

show interfaces counters [<IF>]

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе 3.3.

Команда отображает счётчики для портов маршрутизатора, субинтерфейсов и туннельных интерфейсов.

Можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены счетчики всех интерфейсов заданной группы. Если задан определённый интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны счетчики всех системных интерфейсов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

esr# show interfaces counters gigabitethernet 1/0/4-6				
Interface	UC recv	Bytes recv	Errors recv	MC recv
-----	-----	-----	-----	-----
gil/0/4	0	0	0	0
gil/0/5	0	0	0	0
gil/0/6	0	0	0	0
Interface	UC sent	Bytes sent	Errors sent	
-----	-----	-----	-----	
gil/0/4	0	0	0	
gil/0/5	1138	393748	0	
gil/0/6	0	0	0	
esr# show interfaces counters gigabitethernet 1/0/4				
Packets received:	0			
Bytes received:	0			
Dropped on receive:	0			
Receive errors:	0			
Multicasts received:	0			
Receive length errors:	0			
Receive buffer overflow errors:	0			
Receive CRC errors:	0			
Receive frame errors:	0			
Receive FIFO errors:	0			
Receive missed errors:	0			
Receive compressed:	0			
Packets transmitted:	0			
Bytes transmitted:	0			
Dropped on transmit:	0			

```
Transmit errors:          0
Transmit aborted errors:  0
Transmit carrier errors:  0
Transmit FIFO errors:     0
Transmit heartbeat errors: 0
Transmit window errors:   0
Transmit compressed:      0
Collisions:               0
```

9.1.12 *show interfaces description*

Команда используется для просмотра описания системных интерфейсов.

Синтаксис

```
show interfaces description [ <IF> ]
```

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе 3.3.

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы. При выполнении команды без параметра будут показаны описания всех системных интерфейсов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show interfaces description gigabitethernet 1/0/4-5
Interface      Admin   Link   Description
               State   State
-----
gil1/0/4       Up      Down   Link to NSK
gil1/0/5       Up      Down   Link to MSK
```

9.1.13 *show interfaces status*

Команда используется для просмотра состояния системных интерфейсов.

Синтаксис

```
show interfaces status [<IF>]
```

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе 3.3.

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы. Если задан конкретный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех системных интерфейсов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show interfaces status gigabitethernet 1/0/1-2
```

Interface	Admin state	Link state	MTU	MAC address	Uptime
-----	-----	-----	-----	-----	-----
-					
gil/0/1	Up	Down	1500	a8:f9:4b:aa:53:fc	--
gil/0/2	Up	Up	1500	a8:f9:4b:aa:53:fd	15 hours, 17 minutes and 52 seconds

9.1.14 *show interfaces sfp*

Команда используется для просмотра информации об SFP-трансиверах.

Синтаксис

show interfaces sfp [<IF>]

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе 3.3. В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы. Если задан определенный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех системных интерфейсов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show interfaces sfp
```

Interface 'gil/0/1':	
SFP present:	Yes
Connector Type:	LC
Type:	SFP/SFP+
Compliance code:	10G BASE-SR
Laser wavelength:	850 nm
Transfer distance:	300.00 m
Vendor OUI:	24:00:00
Vendor name:	Modultech
Vendor PN:	MT-PP-85192-SR
Vendor SN:	M1204011007
Vendor date:	04.05.12
Vendor revision:	1.0
DDM supported:	Yes
Temperature:	40.562 C
Voltage:	3.3364 V
Current:	6.004 mA
RX Power:	0.0001 mW / -40.0000 dBm
TX Power:	0.4398 mW / -3.5674 dBm

```

RX LOS:                Yes
TX Fault:              No
TX Disable:            No
Soft TX Disable:       No
Interface 'gil/0/2':
SFP present:           Yes
Connector Type:        SC
Type:                  SFP/SFP+
Compliance code:       1000BASE-LX
Laser wavelength:      1310 nm
Transfer distance:     20.00 km
Vendor OUI:            --
Vendor name:           OEM
Vendor PN:             APSB35123CXS20
Vendor SN:             SG35224701333
Vendor date:           12.12.12
Vendor revision:       1.00
DDM supported:         No

```

9.1.15 *clear interfaces counters*

Данной командой осуществляется сброс счетчиков заданного системного интерфейса или группы интерфейсов.

Синтаксис

```
clear interfaces counters [<IF>]
```

Параметры

<IF> – наименование системного интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе 3.3.

Можно указать несколько интерфейсов перечислением через «,» либо указать диапазон интерфейсов через «-». Если не указывать индексы интерфейсов, то будут очищены счетчики всех интерфейсов заданной группы. При выполнении команды без параметра будут очищены счетчики всех системных интерфейсов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# clear interfaces counters gigabitethernet 1/0/5
```

9.2 TDM (E1)

9.2.1 *ppp authentication chap*

Данной командой включается CHAP-аутентификация.

Использование отрицательной формы команды (no) отключает аутентификацию.

Синтаксис

[no] ppp authentication chap

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-E1

CONFIG-MULTILINK

Пример

```
esr(config-e1)# ppp authentication chap
```

9.2.2 *ppp chap refuse*

Данной командой включается игнорирование аутентификации.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

[no] ppp chap refuse

Параметры

Команда не содержит аргументов.

Значение по умолчанию

Игнорирование аутентификации выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-E1

CONFIG-MULTILINK

Пример

```
esr(config-e1)# ppp chap refuse
```

9.2.3 *ppp chap hostname*

Данной командой указывается имя маршрутизатора, которое отправляется удаленной стороне для прохождения CHAP-аутентификации. Использование отрицательной формы команды (no) устанавливает значение по умолчанию (системное имя устройства).

Синтаксис

ppp chap hostname <NAME>

no ppp chap hostname

Параметры

<NAME> – имя маршрутизатора, задаётся строкой до 31 символа.

Командный режим

CONFIG-E1

CONFIG-MULTILINK

Пример

```
esr(config-e1)# ppp chap hostname esr1
```

9.2.4 *ppp chap password*

Данной командой указывается пароль, который отправляется удаленной стороне вместе с именем маршрутизатора для прохождения CHAP-аутентификации. Использование отрицательной формы команды (no) удаляет пароль.

Синтаксис

ppp chap password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }

no ppp chap password

Параметры

<CLEAR-TEXT> – пароль в открытой форме, задаётся строкой [8 .. 64] символов, может включать символы [0-9a-fA-F];

<ENCRYPTED-TEXT> – пароль в зашифрованной форме, задаётся строкой [16..128] символов.



Пароль хранится в конфигурации в зашифрованной форме независимо от формата, использованного при вводе команды.

Необходимый уровень привилегий

10

Командный режим

CONFIG-E1

CONFIG-MULTILINK

Пример

```
esr(config-e1)# ppp chap password ascii-text 01234567
```

9.2.5 *ppp chap username*

Данной командой указывается пользователь для аутентификации удаленной стороны и осуществляется переход в режим конфигурирования пользователя.

Использование отрицательной формы команды (no) удаляет указанного пользователя.

Синтаксис

ppp chap username <NAME>

no ppp chap username

Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа.

Командный режим

CONFIG-E1

CONFIG-MULTILINK

Пример

```
esr(config-e1)# ppp chap username хар
```

9.2.6 *password*

Команда для установки пароля в открытой или зашифрованной форме определенному пользователю для аутентификации удаленной стороны. Пароль пользователя хранится в конфигурации в зашифрованной форме. При конфигурировании можно задать пароль в открытой форме либо скопировать пароль в зашифрованной форме с другого устройства.

Использование отрицательной формы команды (no) удаляет пароль пользователя.

Синтаксис

password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }

no password

Параметры

<CLEAR-TEXT> – пароль в открытой форме, задаётся строкой [8 .. 64] символов, может включать символы [0-9a-fA-F];

<ENCRYPTED-TEXT> – пароль в зашифрованной форме, задаётся строкой [16..128] символов.



Пароли хранятся в конфигурации в зашифрованной форме независимо от формата, использованного при вводе команды.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPP-USER

Пример

```
esr(config-ppp-user)# password ascii-text 01234567
```

9.2.7 *ppp ipcp accept-address*

Данной командой разрешается принимать от соседа любой ненулевой IP-адрес в качестве локального IP-адреса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

[no] ppp ipcp accept-address

Параметры

Команда не содержит аргументов.

Значение по умолчанию

Прием IP-адреса запрещен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-E1

CONFIG-MULTILINK

Пример

```
esr(config-e1)# ppp ipcp accept-address
```

9.2.8 *ppp ipcp remote-address*

Данной командой устанавливается IP-адрес, который отправляется удаленной стороне для последующего его присвоения.

Использование отрицательной формы команды (no) удаляет IP-адрес удаленной стороны.

Синтаксис

ppp ipcp remote-address <ADDR>

no ppp ipcp remote-address

Параметры

<ADDR> – IP-адрес удаленного шлюза.

Необходимый уровень привилегий

10

Командный режим

CONFIG-E1

CONFIG-MULTILINK

Пример

```
esr(config-e1)# ppp ipcp remote-address 192.168.1.2
```

9.2.9 *ppp max-configure*

Данной командой устанавливается количество попыток отправки Configure-Request пакетов, прежде чем удаленный пир будет признан неспособным ответить. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ppp max-configure <VALUE>

no ppp max-configure

Параметры

<VALUE> – время в секундах, принимает значения [1..255].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-E1

CONFIG-MULTILINK

Пример

```
esr(config-if-gi)#i ppp max-configure 4
```

9.2.10 *ppp max-failure*

Данной командой устанавливается количество попыток выслать Configure-NAK пакеты, прежде чем будут подтверждены все опции. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ppp max-failure <VALUE>

no ppp max-failure

Параметры

<VALUE> – время в секундах, принимает значения [1..255]. Значение по умолчанию 5

Необходимый уровень привилегий

10

Командный режим

CONFIG-E1

CONFIG-MULTILINK

Пример

```
esr(config-if-gi)#i ppp max-failure 3
```

9.2.11 *ppp max-terminate*

Данной командой устанавливается количество попыток выслать Terminate-Request пакеты, прежде чем сессия будет прервана. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ppp max-terminate <VALUE>

no ppp max-terminate

Параметры

<VALUE> – время в секундах, принимает значения [1..255].

Значение по умолчанию

2

Необходимый уровень привилегий

10

Командный режим

CONFIG-E1

CONFIG-MULTILINK

Пример

```
esr(config-if-gi)#i ppp max-terminate 4
```

9.2.12 *ppp mru*

Данной командой указывается размер MRU (Maximum Receive Unit) для интерфейса. Использование отрицательной формы команды (no) устанавливает значение MRU по умолчанию.

Синтаксис

```
ppp mru <MRU>
no ppp mru
```

Параметры

<MRU> – значение MRU, принимает значения в диапазоне [128..1485].

Значение по умолчанию

1485

Необходимый уровень привилегий

10

Командный режим

CONFIG-E1

CONFIG-MULTILINK

Пример

```
esr(config-if-gi)# mru 1400
```

9.2.13 *ppp timeout keepalive*

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор отправляет keepalive-сообщение.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ppp timeout keepalive [ <TIME >]
no ppp timeout keepalive
```

Параметры

<TIME> – время в секундах, принимает значения [1..32767].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-E1

CONFIG-MULTILINK

Пример

```
esr(config-if-gi)# ppp timeout keepalive 200
```

9.2.14 *ppp timeout retry*

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор повторяет запрос на установление сессии.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ppp timeout retry <TIME>
no ppp timeout retry
```

Параметры

<TIME> – время в секундах, принимает значения [1..255].

Значение по умолчанию

3

Необходимый уровень привилегий

10

Командный режим

CONFIG-E1

CONFIG-MULTILINK

Пример

```
esr(config-if-gi)# ppp timeout retry 3
```

9.2.15 *switchport e1 slot*

Данной командой порт e1 привязывается к физическому интерфейсу. Использование отрицательной формы команды (no) переходит в стандартный режим.

Синтаксис

```
[no] switchport mode e1 < SLOT>
```

Параметры

<SLOT> – идентификатор слота, принимает значение в диапазоне [0..3].

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

Пример

```
esr(config-if-gi)# switchport mode e1
```

10 УПРАВЛЕНИЕ ГРУППАМИ АГРЕГАЦИИ КАНАЛОВ

10.1 *Link Agregation Group (LAG)*

10.1.1 *channel-group*

Данной командой физический интерфейс включается в группу агрегации каналов.

Использование отрицательной формы команды (no) удаляет интерфейс из группы агрегации каналов.

Синтаксис

```
channel-group <ID> mode <MODE>
no channel-group
```

Параметры

<ID> – порядковый номер группы агрегации каналов, принимает значения [1..12].

<MODE> – режим формирования группы агрегации каналов:

- auto – добавить интерфейс в динамическую группу агрегации с поддержкой протокола LACP;
- on – добавить интерфейс в статическую группу агрегации.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

Пример

```
esr(config-if-gi)# channel-group 6 mode auto
```

10.1.2 *lacp system-priority*

Данной командой устанавливается приоритет системы для протокола LACP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
lacp system-priority <PRIORITY>
no lacp system-priority
```

Параметры

<PRIORITY> – приоритет, указывается в диапазоне [1..65535].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# lacp system-priority 5000
```

10.1.3 *port-channel load-balance*

Данной командой устанавливается механизм балансировки нагрузки для групп агрегации каналов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
port-channel load-balance {src-dst-mac-ip|src-dst-mac|src-dst-ip|src-dst-mac-ip-port}
no port-channel load-balance
```

Параметры

- src-dst-mac-ip – механизм балансировки основывается на MAC-адресе и IP-адресе отправителя и получателя;
- src-dst-mac – механизм балансировки основывается на MAC-адресе отправителя и получателя;
- src-dst-ip – механизм балансировки основывается на IP-адресе отправителя и получателя;
- src-dst-mac-ip-port – механизм балансировки основывается на MAC-адресе, IP-адресе и порте отправителя и получателя.

Значение по умолчанию

src-dst-mac

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# port-channel load-balance src-dst-mac-ip
```

10.1.4 *lacp timeout*

Данной командой устанавливается административный таймаут протокола LACP.

Использование отрицательной формы команды устанавливает значение по умолчанию.

Синтаксис

```
lacp timeout { short | long }
no lacp timeout
```

Параметры

- long – длительное время таймаута;
- short – короткое время таймаута.

Значение по умолчанию

long

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

Пример

```
esr(config-if-gi)# lacp timeout short
```

10.1.5 ***lacp port-priority***

Данной командой устанавливается LACP-приоритет интерфейса Ethernet.

Использование отрицательной формы команды устанавливает значение по умолчанию.

Синтаксис

lacp port-priority <PRIORITY>

no lacp port-priority

Параметры

<PRIORITY> – приоритет, указывается в диапазоне [1..65535].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

Пример

```
esr(config-if-gi)# lacp port-priority 5000
```

10.1.6 ***show lacp statistics***

Данная команда используется для просмотра статистики работы LACP-протокола для интерфейса Ethernet.

Синтаксис

show lacp statistics <IF>

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе 3.3.

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будет отображена статистика всех интерфейсов заданной группы. Если задан определенный интерфейс, то будет отображена детальная информация по данному интерфейсу.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show lacp statistics gigabitethernet 1/0/2
gil/0/21 LACP statistics:
    LACP Pdus sent:                0
    LACP Pdus received:            0
    Link failure count:             0
```

10.1.7 *show lacp parameters*

Данная команда используется для просмотра параметров настройки протокола LACP для интерфейса Ethernet.

Синтаксис

show lacp parameters <IF>

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе 3.3.

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены параметры всех интерфейсов заданной группы. Если задан конкретный интерфейс, то будет отображена детальная информация по данному интерфейсу.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show lacp parameters gigabitethernet 1/0/2
    LACP parameters
    ~~~~~
Interface  Port Priority  Timeout  Mode
-----
gil/0/2    32768          Short    Active
```

10.1.8 *show lacp*

Данная команда используется для просмотра информации о протоколе LACP.

Синтаксис

```
show lacp <IF>
```

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе 3.3.

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будет отображена информация о LACP-протоколе для всех интерфейсов заданной группы.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show lacp port-channel 1
Active Aggregator: none
Channel group 1 (Aggregator 1)      Number of ports: 1
      Actor System      Partner System
System Priority: 50000      65535
System MAC: a8:f9:4b:aa:03:00 00:00:00:00:00:00
Key: 0x0044      0x0001
esr# show lacp gigabitethernet 1/0/6
Port gigabitethernet 1/0/6: [backup], link down
      Actor Port      Partner Port
Port Number: 12      1
Port Priority: 32768      255
LACP Activity: active      passive
```

10.1.9 *show interfaces port-channel*

Данная команда используется для просмотра информации о членах группы агрегации каналов.

Синтаксис

```
show interfaces port-channel [<ID>]
```

Параметры

<ID> – порядковый номер группы агрегации каналов, принимает значения в диапазоне [1..12].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show interfaces port-channel 1
load-balance: src-dst-mac

Channels  Ports
-----  -
po1      gi1/0/2
```

10.1.10 *show lacp port-channel*

Данная команда показывает информацию о протоколе LACP для группы портов.

Синтаксис

`show interfaces port-channel [<ID>]`

Параметры

<ID> – порядковый номер группы агрегации каналов, принимает значения в диапазоне [1..12].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show lacp port-channel
Active Aggregator: none

Channel group 1 (Aggregator 1)    Number of ports: 1
      Actor System      Partner System
System Priority:    32768      65535
System MAC:        02:20:03:a0:04:90  00:00:00:00:00:00
Key:                0x0044      0x0001

Channel group 1 (Aggregator 2)    Number of ports: 1
      Actor System      Partner System
System Priority:    32768      65535
System MAC:        02:20:03:a0:04:90  00:00:00:00:00:00
Key:                0x0044      0x0001
```

10.2 MLPPP – Multilink PPP

10.2.1 *ppp multilink*

Данной командой включается режим MLPPP на E1-интерфейсе.

Использование отрицательной формы команды (no) отключает режим MLPPP.

Синтаксис

`[no] ppp multilink`

Параметры

Команда не содержит аргументов

Необходимый уровень привилегий

10

Командный режим

CONFIG-E1

Пример

```
esr(config-e1)# ppp multilink
```

10.2.2 *ppp multilink-group*

Данной командой E1-интерфейс включается в группу агрегации.

Использование отрицательной формы команды (no) исключает интерфейс из группы агрегации.

Синтаксис

ppp multilink-group <GROUP-ID>

Параметры

<GROUP-ID> – идентификатор группы, принимает значение [1..4].

Необходимый уровень привилегий

10

Командный режим

CONFIG-E1

Пример

```
esr(config-e1)# ppp multilink-group 1
```

11 УПРАВЛЕНИЕ ТУННЕЛЯМИ

Порядок именования туннелей в маршрутизаторе описан в разделе 3.3.



SoftGRE туннели используются при работе маршрутизатора в качестве Wi-Fi контроллера, данный режим доступен только для ESR-1000 при наличии соответствующей лицензии.

11.1 tunnel

Данная команда позволяет перейти в режим конфигурирования туннеля.

Использование отрицательной формы команды (no) удаляет туннель.

Синтаксис

```
[no] tunnel { gre | ip4ip4 | l2tpv3 | vti } <INDEX>
```

```
[no] tunnel softgre <INDEX>[.<VLAN>]
```

Параметры

<INDEX> – идентификатор туннеля, задается в виде, описанном в разделе 3.4;

<VLAN> – идентификатор VLAN виртуального интерфейса SoftGRE-туннеля.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример 1

Переход в режим конфигурирования туннеля l2tpv3 10:

```
esr(config)# tunnel l2tpv3 10
esr(config-l2tpv3) #
```

Пример 2

Переход в режим конфигурирования туннеля ip4ip4 10:

```
esr(config)# tunnel ip4ip4 10
esr(config-ip4ip4) #
```

Пример 3

Переход в режим конфигурирования туннеля gre 10:

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

Пример 6

Переход в режим конфигурирования туннеля vti 10:

```
esr(config)# tunnel vti 125
esr(config-vti) #
```

11.2 enable

Данной командой включается туннель.

Использование отрицательной формы команды (no) отключает туннель.

Синтаксис

[no] enable

Параметры

Команда не содержит аргументов.

Значение по умолчанию

Туннель выключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-SOFTGRE

CONFIG-SUBTUNNEL

CONFIG-L2TPV3

CONFIG-VTI

Пример

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

11.3 description

Данная команда используется для изменения описания конфигурируемого туннеля.

Использование отрицательной формы команды (no) удаляет установленное описание.

Синтаксис

description <DESCRIPTION>

no description

Параметры

<DESCRIPTION> – описание туннеля, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IP4IP4

CONFIG-SOFTGRE

CONFIG-SUBTUNNEL

CONFIG-GRE
CONFIG-L2TPV3
CONFIG-VTI

Пример

```
esr(config-gre)# description "tunnel to branch"
```

11.4 local address

Данной командой устанавливается IP-адрес локального шлюза туннеля.

Использование отрицательной формы команды (no) удаляет IP-адрес локального шлюза.

Синтаксис

local address <ADDR>
no local address

Параметры

<ADDR> – IP-адрес локального шлюза.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IP4IP4
CONFIG-GRE
CONFIG-SOFTGRE
CONFIG-L2TPV3
CONFIG-VTI

Пример

```
esr(config-ip4ip4)# local address 192.168.1.1
```

11.5 remote address

Данной командой устанавливается IP-адрес удаленного шлюза туннеля.

Использование отрицательной формы команды (no) удаляет IP-адрес удаленного шлюза.

Синтаксис

remote address <ADDR>
no remote address

Параметры

<ADDR> – IP-адрес удаленного шлюза.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IP4IP4
CONFIG-GRE
CONFIG-SOFTGRE
CONFIG-L2TPV3
CONFIG-VTI

Пример

```
esr(config-ip4ip4)# remote address 192.168.1.2
```

11.6 mtu

Данной командой указывается размер MTU (Maximum Transmission Unit) для туннелей.

Использование отрицательной формы команды (no) устанавливает значение MTU по умолчанию.

Синтаксис

mtu <MTU>
no mtu

Параметры

<MTU> – значение MTU, принимает значения в диапазоне [1280..1500].

Значение по умолчанию

1500

Необходимый уровень привилегий

10

Командный режим

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-L2TPV3

CONFIG-VTI

Пример

```
esr(config-l2tpv3)# mtu 1400
```

11.7 ttl

Команда задаёт значение времени жизни TTL для туннельных пакетов.

Использование отрицательной формы команды (no) устанавливает значение TTL по умолчанию.

Синтаксис

ttl <TTL>

no ttl

Параметры

<TTL> – значение TTL, принимает значения в диапазоне [1..255].

Значение по умолчанию

Наследуется от инкапсулируемого пакета.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IP4IP4

CONFIG-GRE

Пример

```
esr(config-ip4ip4)# ttl 10
```

11.8 dscp

Команда задаёт значение кода DSCP для использования в IP заголовке инкапсулирующего пакета.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

dscp <DSCP>

no dscp

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

Наследуется от инкапсулируемого пакета

Необходимый уровень привилегий

10

Командный режим

CONFIG-IP4IP4

CONFIG-GRE

Пример

```
esr(config-ip4ip4)# dscp 40
```

11.9 key

Данная команда разрешает передачу ключа (Key) в туннельном заголовке GRE (в соответствии с RFC 2890) и устанавливает значение ключа. Ключ может быть использован для идентификации потоков трафика в GRE туннеле.

Использование отрицательной формы команды (no) запрещает передачу ключа.

Синтаксис

key <KEY>

no key

Параметры

<KEY> – значение KEY, принимает значения в диапазоне [1..2000000].

Значение по умолчанию

Ключ не передаётся.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
esr(config-gre)# key 40
```

11.10 local checksum

Данная команда включает вычисление контрольной суммы и занесение её в GRE заголовки отправляемых пакетов.

Использование отрицательной формы команды (no) отключает процесс вычисления и отправки контрольной суммы.

Синтаксис

[no] local checksum

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

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

11.11 *remote checksum*

Команда включает проверку наличия и соответствия значений контрольной суммы в заголовках принимаемых GRE-пакетов.

Использование отрицательной формы команды (no) отключает проверку контрольной суммы.

Синтаксис

[no] remote checksum

Параметры

Команда не содержит аргументов.

Значение по умолчанию

По умолчанию проверка контрольной суммы выключена.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

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

11.12 *protocol*

Выбор метода инкапсуляции для туннеля L2TPv3.

Синтаксис

protocol <TYPE>

no protocol

Параметры

<TYPE> – тип инкапсуляции, возможные значения:

IP-инкапсуляция в IP-пакет;

UDP-инкапсуляция в UDP-дейтаграммы.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TPV3

Пример

```
esr(config-l2tpv3)# protocol ip
```

11.13 local cookie

Данная команда определяет значение cookie для дополнительной проверки соответствия между передаваемыми данными и сессией.

Использование отрицательной формы команды (no) удаляет локальный cookie.

Синтаксис

local cookie <COOKIE>

no local cookie

Параметры

<COOKIE> – значение COOKIE, параметр принимает значения длиной восемь или шестнадцать символов в шестнадцатеричном виде [8 or 16].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TPV3

Пример

```
esr(config-l2tpv3)# local cookie 8FB51B8FB
```

11.14 local port

Команда определяет локальный UDP-порт, если в качестве метода инкапсуляции был выбран UDP протокол.

Использование отрицательной формы команды (no) удаляет локальный номер UDP-порта.

Синтаксис

local port <UDP>

no local port

Параметры

<UDP> – номер UDP-порта в диапазоне [1..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TPV3

Пример

```
esr(config-l2tpv3)# local port 1501
```

11.15 *local session-id*

Установить локальный идентификатор сессии.

Использование отрицательной формы команды (no) удаляет локальный идентификатор сессии.

Синтаксис

local session-id <SESSION-ID>

no local session-id

Параметры

<SESSION-ID> – идентификатор сессии, принимает значения [1..200000].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TPV3

Пример

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

11.16 *local tunnel-id*

Данной командой устанавливается локальный идентификатор туннеля.

Использование отрицательной формы команды (no) удаляет локальный идентификатор туннеля.

Синтаксис

local tunnel-id <TUNNEL-ID>

no local tunnel-id

Параметры

<TUNNEL-ID> – идентификатор сессии, принимает значения [1..200000].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TPV3

Пример

```
esr(config-l2tpv3)# local tunnel-id 215
```

11.17 *remote cookie*

Данная команда определяет значение cookie для дополнительной проверки соответствия между передаваемыми данными и сессией.

Использование отрицательной формы команды (no) удаляет удаленный cookie.

Синтаксис

remote cookie <COOKIE>

no remote cookie

Параметры

<COOKIE> – значение COOKIE, принимает значения длиной восемь или шестнадцать символов в шестнадцатеричном виде [8 or 16].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TPV3

Пример

```
esr(config-l2tpv3)# remote cookie 8FB51B8FB
```

11.18 remote port

Данная команда определяет удаленный UDP-порт, если в качестве метода инкапсуляции был выбран UDP.

Использование отрицательной формы команды (no) удаляет удаленный номер UDP-порта.

Синтаксис

remote port <UDP>

no remote port

Параметры

<UDP> – номер UDP порта в диапазоне [1..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TPV3

Пример

```
esr(config-l2tpv3)# remote port 65000
```

11.19 remote session-id

Данной командой устанавливается удаленный идентификатор сессии.

Использование отрицательной формы команды (no) удаляет удаленный идентификатор сессии.

Синтаксис

remote session-id <SESSION-ID>

no remote session-id

Параметры

<SESSION-ID> – идентификатор сессии, принимает значение в диапазоне [1..200000].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TPV3

Пример

```
esr(config-l2tpv3)# remote session-id 2
```

11.20 *remote tunnel-id*

Данной командой устанавливается удаленный идентификатор туннеля.

Использование отрицательной формы команды (no) удаляет удаленный идентификатор туннеля.

Синтаксис

remote tunnel-id <TUNNEL-ID>

no remote tunnel-id

Параметры

<TUNNEL-ID> – идентификатор туннеля, принимает значение в диапазоне [1..200000].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TPV3

Пример

```
esr(config-l2tpv3)# remote tunnel-id 192
```

11.21 *mode*

Данной командой задается режим работы SoftGRE-туннеля.

Использование отрицательной формы команды (no) снимает установленный режим.

Синтаксис

mode <MODE>

no mode

Параметры

<MODE> – режим работы туннеля, возможные значения:

data – режим данных;

management – режим управления.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SOFTGRE

Пример

```
esr(config-softgre)# mode data
```

11.22 *default-profile*

Данная команда позволяет использовать конфигурацию данного SoftGRE-туннеля для автоматического создания туннелей с такими же mode и local address.

Использование отрицательной формы команды (no) запрещает использования конфигурации туннеля для автоматического создания туннелей.

Синтаксис

[no] default-profile

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SOFTGRE

Пример

```
esr(config-softgre)# default-profile
```

11.23 *show tunnels configuration*

Командой выполняется просмотр конфигурации туннеля.

Синтаксис

show tunnels configuration { gre | ip4ip4 | l2tpv3 | softgre | vti } [<INDEX>]

Параметры

<INDEX> – идентификатор туннеля, задается в виде, описанном в разделе 3.4.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show tunnels configuration gre 25
State:                               enabled
Description:
Local address:                       14.0.0.2
Remote address:                      14.0.0.1
Calculates checksums for outgoing GRE packets: no
Requires that all input GRE packets were checksum: no
key:                                  -
TTL:                                  Inherit
```


DSCP:	0
MTU:	1500
Security zone:	remote

11.24 *show tunnels status*

Команда используется для просмотра состояния системных интерфейсов.

Синтаксис

```
show tunnels status [ { gre | ip4ip4 | l2tpv3 | vti } [ <INDEX> ] ]
show tunnels status softgre <INDEX>[.<VLAN>]
```

Параметры

<INDEX> – идентификатор туннеля, задается в виде, описанном в разделе 3.4.

В команде можно указать несколько туннелей. Если не указывать индексы туннелей, то будут отображены статусы всех туннелей заданной группы. Если задан конкретный туннель, то будет отображена детальная информация по данному туннелю.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

esr# show tunnels status					
Tunnel	Admin state	MTU	Local IP	Remote IP	Uptime
-----	-----	-----	-----	-----	-----
ip4ip4 4	Up	1500	115.0.0.100	115.0.0.30	1 minute and 4 seconds

11.25 *show tunnels counters*

Командой выполняется просмотр счетчиков на туннелях.

Синтаксис

```
show tunnels counters [ { gre | ip4ip4 | l2tpv3 | vti } [ <INDEX> ] ]
show tunnels counters softgre <INDEX>[.<VLAN>]
```

Параметры

<INDEX> – идентификатор туннеля, задается в виде, описанном в разделе 3.4.

Можно указать несколько туннелей. Если не указывать индексы туннелей, то будут отображены счетчики всех туннелей заданной группы. Если задан определённый туннель, то будет отображена детальная информация по данному туннелю.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show tunnels counters l2tpv3 1
Tunnel 'l2tpv3 1' counters:
Packets received:          0
Bytes received:            0
Dropped on receive:       0
Receive errors:            0
Multicasts received:      0
Receive length errors:    0
Receive buffer overflow errors: 0
Receive CRC errors:       0
Receive frame errors:     0
Receive FIFO errors:      0
Receive missed errors:    0
Receive compressed:       0
Packets transmitted:      658
Bytes transmitted:        56588
Dropped on transmit:      0
Transmit errors:          0
Transmit aborted errors:  0
Transmit carrier errors:  0
Transmit FIFO errors:     0
Transmit heartbeat errors: 0
Transmit window errors:   0
Transmit compressed:      0
Collisions:               0
```

11.26 *show tunnels utilization*

Команда используется для просмотра средней нагрузки в туннелях за указанный период.

Синтаксис

```
show tunnels utilization [ { gre | ip4ip4 | l2tpv3 | vti } [<INDEX>] ]
```

```
show tunnels utilization softgre <INDEX>[.<VLAN>]
```

Параметры

<INDEX> – идентификатор туннеля, задается в виде, описанном в разделе 3.4.

Можно указать несколько туннелей перечислением через «,» либо указать диапазон интерфейсов через «-». Если не указывать индексы туннелей, то будут очищены счетчики всех туннелей заданной группы.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
esr-200# show tunnels utilization gre 2
Tunnel      Period, s      Sent,      Recv,      Frames Sent  Frames Recv
-----
gre 2       15              0          0          0            0
```

11.27 *clear tunnels counters*

Данной командой осуществляется сброс счетчиков заданного туннеля или группы туннелей.

Синтаксис

```
clear tunnels counters [ { gre | ip4ip4 | l2tpv3 | vti } [<INDEX>] ]
```

```
clear tunnels counters softgre <INDEX>[.<VLAN>]
```

Параметры

<INDEX> – идентификатор туннеля, задается в виде, описанном в разделе 3.4.

<VLAN> – идентификатор VLAN виртуального интерфейса SoftGRE-туннеля.

Можно указать несколько туннелей перечислением через «,» либо указать диапазон интерфейсов через «-». Если не указывать индексы туннелей, то будут очищены счетчики всех туннелей заданной группы.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
esr# clear tunnels counters gre 25
```

12 УПРАВЛЕНИЕ L2 ФУНКЦИЯМИ

12.1 Управление L2 маршрутизацией

12.1.1 *bridge*

Данной командой добавляется сетевой мост в систему и осуществляется переход в режим настройки параметров его параметров.

Использование отрицательной формы команды (no) удаляет мост.

Синтаксис

[no] bridge <BRIDGE-ID>

Параметры

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе 303.3.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

Переход в режим конфигурирования сетевого моста *bridge 10*:

```
esr(config)# bridge 10
esr(config-bridge)#
```

12.1.2 *enable*

Данной командой разрешается маршрутизация данных между интерфейсами, включенными в сетевой мост.

Использование отрицательной формы команды (no) выключает маршрутизацию данных.

Синтаксис

[no] enable

Параметры

Команда не содержит аргументов.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BRIDGE

Пример

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

12.1.3 *description*

Данная команда используется для назначения описания конфигурируемому сетевому мосту. Использование отрицательной формы команды (no) удаляет установленное описание.

Синтаксис

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

Параметры

<DESCRIPTION> – описание сетевого моста, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BRIDGE

Пример

```
esr(config-bridge)# description "broadway"
```

12.1.4 *vlan*

Данная команда используется для связывания текущего сетевого моста с VLAN. Все порты, являющиеся членами назначаемого VLAN, автоматически включаются в сетевой мост и становятся участниками общего L2 домена. Для управления членством сетевых интерфейсов в VLAN используются команды, описанные в разделе 12.2.3.

Использование отрицательной формы команды (no) удаляет привязку VLAN и отключает соответствующие интерфейсы от сетевого моста.

Синтаксис

```
vlan <VID>
no vlan
```

Параметры

<VID> – идентификатор VLAN, задаётся в диапазоне [1..4095].

Необходимый уровень привилегий

10

Командный режим

CONFIG-BRIDGE

Пример

```
esr(config-bridge)# vlan 40
```

12.1.5 *bridge-group*

Данная команда используется для добавления в текущего сетевого интерфейса в L2 домен.

Использование отрицательной формы команды (no) удаляет интерфейс из L2 домена.

Синтаксис

```
bridge-group <BRIDGE-ID>
no bridge-group
```

Параметры

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе 303.3.

Необходимый уровень привилегий

10

Командный режим

```
CONFIG-SUBIF
CONFIG-L2TPV3
CONFIG-SUBTUNNEL
```

Пример

```
esr(config-if-gi)# bridge-group 15
```

12.1.6 *mac-address*

Данная команда позволяет задать MAC-адрес сетевого моста, отличный от системного.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
mac-address <ADDR>
no mac-address
```

Параметры

<ADDR> – MAC-адрес сетевого моста, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

Значение по умолчанию

Системный MAC-адрес

Необходимый уровень привилегий

10

Командный режим

```
CONFIG-BRIDGE
```

Пример

```
esr(config-bridge)# mac-address A8:F9:B0:00:00:04
```

12.1.7 *show interfaces bridge*

Данная команда используется для просмотра информации о VLAN, саб-интерфейсах, туннелях объединенных мостом.

Синтаксис

```
show interfaces bridge [<BRIDGE-ID>]
```

Параметры

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе 303.3.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show interfaces bridge 1
Bridges      Interfaces
-----
bridge 1     vlan 1,gi1/0/1.10
```

12.2 *Настройка и мониторинг VLAN*

12.2.1 *vlan*

Данной командой добавляется VLAN в систему и осуществляется переход в режим настройки параметров VLAN. На маршрутизаторе всегда существует VLAN с идентификатором 1, все интерфейсы по умолчанию включены в данный VLAN.

Использование отрицательной формы команды (no) удаляет VLAN.

Синтаксис

```
[no] vlan <VID>
```

Параметры

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094].

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# vlan 40
```

12.2.2 *name*

Данная команда используется для добавления описания VLAN.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

name <NAME>

no name

Параметры

<NAME> – описание VLAN, задаётся строкой до 255 символов.

Значение по умолчанию

Описание не задано.

Необходимый уровень привилегий

10

Командный режим

CONFIG-VLAN

Пример

```
esr(config)# name L2-ACCESS
```

Пример

```
esr(config-if-gi)# ip internal-usage-vlan 1500
```

12.2.3 *switchport default-vlan tagged*

Данная команда используется для изменения членства интерфейса во VLAN по умолчанию на тегированное.

Использование отрицательной формы команды (no) изменяет членство интерфейса во VLAN по умолчанию на нетегированное.

Синтаксис

[no] switchport default-vlan tagged

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

Пример

```
esr(config-if-gi)# switchport default-vlan tagged
```

12.2.4 *switchport forbidden default-vlan*

Данная команда используется для удаления интерфейса из VLAN по умолчанию.

Использование отрицательной формы команды (no) разрешает добавление vlan на порту.

Синтаксис

[no] switchport forbidden default-vlan

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

Пример

```
esr(config-if-gi)# no switchport forbidden default-vlan
```

12.2.5 *switchport access vlan*

Данная команда используется для включения/исключения интерфейса в/из VLAN в режиме работы access.

Синтаксис

switchport access vlan <VID>

no switchport access vlan

Параметры

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094].

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

Пример

```
esr-100(config-if-gi)# switchport access vlan 50
```

12.2.6 *switchport trunk allowed vlan*

Данная команда используется для включения/исключения интерфейса в/из VLAN в режиме работы trunk.

Синтаксис

switchport trunk allowed vlan <ACT> <VID>

Параметры

<ACT> – назначаемое действие:

- add – включение интерфейса во VLAN;
- remove – исключение интерфейса из VLAN;

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094]. Можно задать диапазоном через «-» или перечислением через «,».

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

Пример

```
esr-100(config-if-gi)# switchport trunk allowed vlan add 10-50
```

12.2.7 *switchport trunk native vlan*

Данная команда используется для настройки VLAN по умолчанию для интерфейса в режиме работы trunk. Весь нетегированный трафик, поступающий на данный интерфейс, направляется в данную VLAN.

Синтаксис

switchport trunk native vlan <VID>

no switchport trunk native vlan

Параметры

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

Пример

```
esr-100(config-if-gi)# switchport trunk native vlan 55
```

12.2.8 *show vlans*

Данная команда используется для просмотра информации об определенной VLAN.

Синтаксис

show vlans [<VID>]

Параметры

<VID> – идентификационный номер VLAN, диапазон допустимых значений [1 .. 4094].

Можно указать несколько VLAN перечислением через запятую «,» либо указать диапазон VLAN через дефис «-». При выполнении команды без параметра будут показаны все созданные VLAN.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

esr# show vlans			
VID	Name	Tagged	Untagged
----	-----	-----	-----
1	default		gi1/0/3-4, po1
2	--		gi1/0/1

13 РАБОТА С АДРЕСНЫМИ ТАБЛИЦАМИ

13.1 *ip arp reachable-time*

Данной командой устанавливается время жизни записи в ARP-таблице. Использование отрицательной формы команды (no) устанавливает значение параметра arp reachable-time по умолчанию.

Синтаксис

```
ip arp reachable-time <TIME>  
no ip arp reachable-time
```

Параметры

< TIME > – время жизни динамических MAC-адресов, в миллисекундах. Допустимые значения от 5000 до 100000000 миллисекунд. Реальное время обновления записи варьируется от [0,5;1,5]*< TIME >.

Необходимый уровень привилегий

10

Значение по умолчанию

160000

Командный режим

```
CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-LOOPBACK  
CONFIG-BRIDGE
```

Пример

```
esr(config-if-gi)# ip arp reachable-time 6000
```

13.2 *ipv6 nd reachable-time*

Данной командой устанавливается время, в течение которого удаленный узел IPv6 считается доступным при отсутствии активности узла.

Использование отрицательной формы команды (no) устанавливает значение параметра nd reachable-time по умолчанию.

Синтаксис

```
ipv6 nd reachable-time <TIME>  
no ipv6 nd arp reachable-time
```

Параметры

< TIME > – время жизни записи об удаленном узле IPv6 в таблице ND протокола, в миллисекундах. Допустимые значения от 5000 до 100000000 миллисекунд. Реальное время обновления записи варьируется от [0,5;1,5]*< TIME >.

Значение по умолчанию

30000

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-LOOPBACK

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
esr(config-if-gi)# ipv6 nd reachable-time 27000
```

13.3 mac address-table aging time¹

Командой устанавливается время жизни динамических MAC-адресов в forwarding-таблице.

Использование отрицательной формы команды (no) устанавливает значение «aging time» по умолчанию.

Синтаксис

mac address-table aging-time <AGING TIME>

[no] mac address-table aging time

Параметры

<AGING TIME> – время жизни динамических MAC-адресов, в секундах. Допустимые значения от 10 до 630 секунд. При значении 0 таймер выключен.

Значение по умолчанию

300 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# mac address-table aging-time 30
```

¹ В текущей версии ПО данный функционал поддерживается только на маршрутизаторе ESR-1000

13.4 mac address-table save-secure-freq

Данной командой устанавливается частота сохранения списка статических (secure) MAC-адресов.

Использование отрицательной формы команды (no) устанавливает значение «mac address-table save-secure-freq» по умолчанию.

Синтаксис

```
mac address-table save-secure-freq <SAVE-SECURE-FREQ>
```

```
[no] mac address-table save-secure-freq
```

Параметры

<SAVE-SECURE-FREQ> – частота сохранения списка статических (secure) MAC-адресов, принимает значение [600..86400] секунд.

Значение по умолчанию

1200 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# mac address-table save-secure-freq 650
```

13.5 show mac address-table

Команда используется для просмотра информации об изученных MAC-адресах.

Синтаксис

```
show mac address-table [<OPTIONS>]
```

Параметры

<OPTIONS> – параметры команды для просмотра дополнительной информации и детализации запроса, опциональный параметр. Возможные варианты параметров команды:

count – просмотр количества записей в таблице MAC-адресов. Список MAC-адресов не отображается;

exclude mac <ADDR> [MASK] – просмотреть информацию в таблице по всем MAC-адресам, исключая заданный;

include mac <ADDR> [MASK] – просмотреть определенную запись в таблице по заданному MAC-адресу, где:

<ADDR> – MAC-адрес, по которому ведется поиск, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

[MASK] – маска MAC-адреса, опциональный параметр, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF]. Биты маски, установленные в 0, задают биты MAC-адреса, исключаемые из сравнения при поиске. Значение маски по умолчанию FF:FF:FF:FF:FF:FF.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show mac address-table
VID      MAC address      Port
-----
1         02:01:02:03:04:55  0/CPU
4         02:01:02:03:04:55  0/CPU
5         02:01:02:03:04:55  0/CPU
6         02:01:02:03:04:55  0/CPU
100       02:01:02:03:04:55  0/CPU
101       02:01:02:03:04:55  0/CPU
6 valid mac entries
```

13.6 clear mac address-table

Команда используется для удаления информации об изученных MAC-адресах.

Синтаксис

```
clear mac address-table [ <IF> ]
```

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе 3.3.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
esr# clear mac address-table
```

13.7 show arp

Команда используется для просмотра ARP-таблицы.

Синтаксис

```
show arp [<options>]
```

Параметры

<options> – параметры команды для детализации запрашиваемой информации, опциональный параметр:

vrf <VRF> – имя экземпляра VRF, задается строкой до 31 символа. При указании данного параметра будет отображена ARP-таблица в указанном VRF;

<IF> – наименование системного интерфейса или списка интерфейсов, задаётся в виде, описанном в разделе 3.3 и 3.4. Отображается только информация по указанным интерфейсам;

mac-address <MAC> – MAC-адрес, по которому ведется поиск, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

ip-address <ADDR> – IP-адрес, по которому ведется поиск, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

esr# show arp				
Interface	IP address	MAC address	State	Age (min)
-----	-----	-----	-----	-----
bridge 1	192.168.1.1	a8:f9:4b:aa:00:40	--	--
gi1/0/5	10.255.100.1	d8:50:e6:d2:f0:46	reachable	2
gi1/0/5	10.255.100.5	a8:f9:4b:aa:00:45	--	--

13.8 clear arp-cache

Команда используется для очистки содержимого ARP-таблицы.

Синтаксис

```
clear arp-cache [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа. Опциональный параметр, при указании которого будет очищена ARP-таблица в указанном VRF.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
esr# clear arp-cache
```

13.9 show ipv6 neighbors

Команда используется для просмотра IPv6 Neighbor Discovery таблицы.

Синтаксис

```
show ipv6 neighbors [<OPTIONS>]
```

Параметры

<OPTIONS> – параметры команды для детализации запрашиваемой информации, опциональный параметр:

vrf <VRF> – имя экземпляра VRF, задается строкой до 31 символа. При указании данного параметра будет отображена IPv6 Neighbor Discovery таблица в указанном VRF;

<IF> – наименование системного интерфейса или списка интерфейсов, задается в виде, описанном в разделе 3.3 и 3.4. Отображается только информация по указанным интерфейсам;

mac-address <MAC> – указывается MAC-адрес, по которому ведется поиск, задается в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

ipv6-address <IPv6-ADDR> – указывается IPv6-адрес, по которому ведется поиск, задается в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

esr# show ipv6 neighbors				
Interface	IPv6 address	MAC address	State	Age (min)
-----	-----	-----	-----	-----
-				
gil1/0/5	fc00::1	d8:50:e6:d2:f0:46	reachable	1
gil1/0/5	fc00::2	a8:f9:4b:aa:00:45	--	--
bridge 1	fe80::aaf9:4bff:feaa:40	a8:f9:4b:aa:00:40	--	--
bridge 2	fe80::aaf9:4bff:feaa:40	a8:f9:4b:aa:00:40	--	--
gil1/0/5	fe80::aaf9:4bff:feaa:45	a8:f9:4b:aa:00:45	--	--
gil1/0/5	ff02::16	33:33:00:00:00:16	norarp	--
gil1/0/5	ff02::fb	33:33:00:00:00:fb	norarp	--
gil1/0/5	ff02::1:ff00:1	33:33:ff:00:00:01	norarp	--
gil1/0/5	ff02::1:ff00:2	33:33:ff:00:00:02	norarp	--

13.10 clear ipv6 neighbors

Команда используется для очистки содержимого IPv6 Neighbor Discovery таблицы.

Синтаксис

clear ipv6 neighbors [vrf <VRF>]

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа. При указании данного параметра будет очищена IPv6 Neighbor Discovery таблица в указанном VRF.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
esr# clear ipv6 neighbors
```

13.11 *show arp configuration*

Команда используется для просмотра значений времени жизни записей в ARP таблице.

Синтаксис

show arp configuration <IF>

Параметры

<IF> – наименования системных интерфейсов, задаются в виде, описанном в разделе 3.3;

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
Esr# sh arp configuration gigabitethernet 1/0/1-5
Globally configured ARP reachable time is 6000 msec
Interface          ARP reachable time, msec
-----
gil/0/1            6000
gil/0/2            6000
gil/0/3            6000
gil/0/4            6000
gil/0/4            6000
```

13.12 *show ipv6 neighbors configuration*

Команда используется для просмотра значений времени жизни записи об удаленном узле в таблице ND протокола.

Синтаксис

show ipv6 neighbors configuration <IF>

Параметры

<IF> – наименования системных интерфейсов, задаются в виде, описанном в разделе [3.3](#);

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
esr# sh ipv6 neighbors configuration gigabitethernet 1/0/1-2
Globally configured NDP reachable time is 30000 msec
Interface          ND reachable time, msec
-----
gil/0/1            30000
gil/0/2            30000
```

14 НАСТРОЙКА VRF

14.1 ip vrf

Данной командой в системе создается экземпляр VRF и осуществляется переход в режим настройки параметров экземпляра VRF.

Использование отрицательной формы команды (no) удаляет экземпляр VRF из системы.

Синтаксис

```
[no] ip vrf <VRF>
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# ip vrf VRF1  
esr(config-vrf)#
```

14.2 description

Данная команда используется для назначения описания конфигурируемому экземпляру VRF.

Использование отрицательной формы команды (no) удаляет установленное описание.

Синтаксис

```
description <DESCRIPTION>
```

```
no description
```

Параметры

<DESCRIPTION> – описание экземпляра VRF, задается строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRF

Пример

```
esr(config-vrf)# description "VRF1"
```

14.3 ip vrf forwarding

Данной командой задается имя экземпляра VRF, в котором будут использоваться указанные сетевой интерфейс, мост, зона безопасности или группа правил NAT.

Использование отрицательной формы команды (no) удаляет привязку сетевого интерфейса, моста, зоны безопасности или группы правил NAT к экземпляру VRF.

Синтаксис

```
ip vrf forwarding <VRF>
no ip vrf forwarding
```

Параметры

<VRF> – имя VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

```
CONFIG-GI
CONFIG-SUBIF
CONFIG-PORT-CHANNEL
CONFIG-BRIDGE
CONFIG-LOOPBACK
CONFIG-GRE
CONFIG-ZONE
CONFIG-DNAT-RULESET
CONFIG-SNAT-RULESET
```

Пример

```
esr(config-snat-ruleset)# ip vrf forwarding VRF1
```

14.4 ip source-vrf

Данной командой осуществляется перевод системы копирования по TFTP, SCP, FTP в VRF.

Использование отрицательной формы команды (no) системы копирования в стандартный режим.

Синтаксис

```
ip { tftp | ftp | ssh } source-vrf <VRF>
no ip { tftp | ftp | ssh } source-vrf
```

Параметры

<VRF> – имя VRF, задается строкой до 31 символа.

tftp – копирование по протоколу TFTP;

ftp – копирование по протоколу FTP;

ssh – копирование по протоколу SCP.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# ip tftp source-vrf vrf1
```

14.5 show ip vrf

Команда используется для просмотра информации о существующих в системе экземплярах VRF.

Синтаксис

```
show ip vrf [ <VRF> ]
```

Параметры

<VRF> – имя VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show ip vrf
Name                               Interfaces
-----
VRF1                               gi1/0/8
VRF2                               gi1/0/10.22
test                               gi1/0/2
```

15 НАСТРОЙКА IP АДРЕСАЦИИ

15.1 ip address

Данной командой создаётся IP-интерфейс и добавляется IP-адрес и маска подсети для конфигурируемого интерфейса (физического интерфейса, группы агрегации каналов, туннеля или сетевого моста).

Использование отрицательной формы команды (no) удаляет IP-адрес с интерфейса. При удалении последнего адреса IP-интерфейс уничтожается.



При создании IP-интерфейса система резервирует наибольший незанятый VLAN ID, который будет использоваться внутри системы. Для каждого IP-интерфейса на Ethernet-порту резервируется VLAN. Если VLAN уже был зарезервирован для IPv6-интерфейса, то для IP-интерфейса VLAN резервироваться не будет.

Можно зарезервировать VLAN ID для внутреннего использования явно с помощью команды `ip internal-usage vlan <VLAN_ID>`.

Синтаксис

`ip address <ADDR/LEN>`

`no ip address {<ADDR/LEN>| all}`

Параметры

<ADDR/LEN> – IP-адрес и префикс подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32]. Можно указать несколько IP-адресов перечислением через запятую. Может быть назначено до 8 IP-адресов (включая IPv6-адреса) на интерфейс. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IP-адреса;

- all – команда удаляет все IP-адреса на интерфейсе.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-E1

CONFIG-MULTILINK

CONFIG-VTI

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-BRIDGE

CONFIG-LOOPBACK

Пример

```
esr(config-if-gi)# ip address 192.168.25.25/24
```

15.2 show ip interfaces

Команда используется для просмотра информации о существующих в системе IP-интерфейсах.

Синтаксис

```
show ip interfaces [ vrf <VRF> ] [ { <IF> | <TUN> | ip-address <ADDR> } ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа. При указании данного параметра будут отображены IP-интерфейсы в указанном VRF;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть принимает значения [0..255]. При указании данного параметра будет отображен IP-интерфейс с указанным IP-адресом;

<IF> – наименования системных интерфейсов, задаются в виде, описанном в разделе 3.3;

<TUN> – наименования туннелей, задаются в виде, описанном в разделе 3.4.

В команде можно указать несколько системных интерфейсов. Если не указывать индексы интерфейсов, то будут отображены все IP-интерфейсы, относящиеся к системным интерфейсам указанного типа.

Если в команде указан определенный системный интерфейс, получающий IP-параметры по протоколу DHCP, то будут отображены настройки DHCP-клиента и состояние текущей аренды IP-параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show ip interfaces

IP address      Interface      Type
-----
155.0.0.60/24   br10           static
16.0.0.2/24     gil/0/16       DHCP
10.0.0.1/8      gil/0/18       static
180.0.0.1/24    gil/0/18       static
192.168.1.1/24  gil/0/2        static
25.0.0.2/30     gre 25         static
10.1.0.2/24     ip4ip4 10      static

esr# show ip interfaces gigabitethernet 1/0/16
IP address      Interface      Type
-----
16.0.0.2/24     gil/0/16       DHCP

DHCP Client settings:

DHCP Server:      N/A
Lease time(dd:hh:mm): 00:02:00
Reboot time:      10 seconds
Retry time:       300 seconds
Timeout:          60 seconds
Select timeout:   0 seconds
```

Vendor class ID: N/A

Ignore options:

router

Latest lease contents:

Lease time(dd:hh:mm): 00:02:00

DHCP message type: DHCPACK

Renew at: Wednesday2015/02/25 12:22:24 2015/02/25 12:22:24

Rebind at: Wednesday2015/02/25 13:14:09 2015/02/25 13:14:09

Expires at: Wednesday2015/02/25 13:29:09 2015/02/25 13:29:09

16 НАСТРОЙКА IPV6 АДРЕСАЦИИ

16.1 *ipv6 address*

Данной командой создаётся IPv6-интерфейс и добавляется IPv6-адрес и маска подсети для конфигурируемого интерфейса: физического интерфейса, группы агрегации каналов, туннеля или сетевого моста.

Использование отрицательной формы команды (no) удаляет IPv6-адрес с интерфейса. При удалении последнего адреса IPv6-интерфейс уничтожается.



При создании IPv6-интерфейса система резервирует наибольший незанятый VLAN ID, который будет использоваться внутри системы. Для каждого IPv6-интерфейса на Ethernet-порту резервируется VLAN. Если VLAN уже был зарезервирован для IP-интерфейса, то для IPv6-интерфейса VLAN резервироваться не будет.

Можно зарезервировать VLAN ID для внутреннего использования явно с помощью команды `ip internal-usage vlan <VLAN_ID>`.

Синтаксис

```
ipv6 address <IPV6-ADDR/LEN>
```

```
no ipv6 address {<IPV6-ADDR/LEN>|all}
```

Параметры

<IPV6-ADDR/LEN> – IP-адрес и префикс подсети, задаётся в виде X:X:X:X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128]. Можно указать несколько IPv6-адресов перечислением через запятую. Может быть назначено до 8 IPv6-адресов (включая IP-адреса) на интерфейс. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IPv6-адреса.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

Пример

```
esr(config-if-gi)# ipv6 address fc00::1/120
```

16.2 show ipv6 interfaces

Команда используется для просмотра информации о существующих в системе IPv6-интерфейсах.

Синтаксис

```
show ip interfaces [ vrf <VRF> ] [ { <IF> | <TUN> | ipv6-address <IPv6-ADDR> } ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа. При указании данного параметра будут отображены IPv6-интерфейсы в указанном VRF;

<IF> – наименования системных интерфейсов, задаются в виде, описанном в разделе 3.3;

<TUN> – наименования туннелей, задаются в виде, описанном в разделе 3.4;

<IPv6-ADDR> – IP-адрес, задаётся в виде X:X:X:X:X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128]. При указании данного параметра будут отображен IPv6-интерфейс с указанным IPv6-адресом.

В команде можно указать несколько системных интерфейсов. Если не указывать индексы интерфейсов, то будут отображены все IPv6-интерфейсы, относящиеся к системным интерфейсам указанного типа.

Если в команде указан определенный системный интерфейс, получающий IPv6 параметры по протоколу DHCP, то будут отображены настройки DHCP-клиента и состояние текущей аренды IPv6-параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show ipv6 interfaces
IPv6 address
-----
fc00::2/120
fe80::aaf9:4bff:feaa:45/64
esr# show ip interfaces gigabitethernet 1/0/16
IP address      Interface      Type
-----
16.0.0.2/24     gil/0/16      DHCP
DHCP Client settings:
DHCP Server:    N/A
Lease time(dd:hh:mm): 00:02:00
Reboot time:    10 seconds
Retry time:     300 seconds
Timeout:        60 seconds
Select timeout: 0 seconds
Vendor class ID: N/A
Ignore options:
router
Latest lease contents:
Lease time(dd:hh:mm): 00:02:00
DHCP message type: DHCPACK
Renew at:       Wednesday2015/02/25 12:22:24 2015/02/25 12:22:24
Rebind at:      Wednesday2015/02/25 13:14:09 2015/02/25 13:14:09
Expires at:     Wednesday2015/02/25 13:29:09 2015/02/25 13:29:09
```

17 УПРАВЛЕНИЕ ПРОФИЛЯМИ IP-АДРЕСОВ И ПОРТОВ

17.1 *object-group network*

Команда предназначена для создания профиля IP-адресов. Профили используются при настройке сервисов, работающих с пулами IP-адресов – например, NAT, Firewall, Remote-Access, также для создания списка префиксов.

Использование отрицательной формы команды (no) удаляет профиль IP-адресов.

Синтаксис

[no] object-group network <NAME>

Параметры

<NAME> – имя конфигурируемого профиля IP-адресов, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все профили IP-адресов.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

Создание профиля IP-адресов с именем *remote* и переход в режим конфигурирования профиля:

```
esr(config)# object-group network remote
```

17.2 *object-group service*

Команда предназначена для создания профиля TCP/UDP портов. Данный профиль используется в правилах сервисов NAT и Firewall.

Использование отрицательной формы команды (no) удаляет профиль.

Синтаксис

[no] object-group service <NAME>

Параметры

<NAME> – наименование профиля портов, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все профили TCP/UDP-портов.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# object-group service ssh
```

17.3 description

Команда используется для изменения описания профиля IP-адресов и профиля портов.

Использование отрицательной формы команды (no) удаляет описание профиля.

Синтаксис

description <DESCRIPTION>

no description

Параметры

<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-NETWORK

CONFIG-OBJECT-GROUP-SERVICE

Пример

Установить описание для профиля IP-адресов:

```
esr(config-object-group-network)# description "Internal addresses"
```

17.4 ip prefix

Команда используется для задания подсети.

Использование отрицательной формы команды (no) удаляет заданную подсеть.

Синтаксис

[no] ip prefix <ADDR/LEN>

Параметры

<ADDR/LEN> – подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-NETWORK

Пример

```
esr(config-object-group-network)# ip prefix 10.10.10.0/24
```

17.5 ip address-range

Команда используется для задания диапазона IP-адресов.

Использование отрицательной формы команды (no) удаляет запись из конфигурируемого профиля.

Синтаксис

[no] ip address-range <FROM-ADDR>[-<TO-ADDR>]

Параметры

<FROM-ADDR> – начальный IP-адрес диапазона адресов;

<TO-ADDR> – конечный IP-адрес диапазона адресов, опциональный параметр. Если параметр не указан, то командой задаётся одиночный IP-адрес.

Адреса задаются в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-NETWORK

Пример

```
esr(config-object-group-network)# ip address 192.168.1.1 192.168.1.25
```

17.6 ipv6 prefix

Команда используется для задания IPv6-подсети.

Использование отрицательной формы команды (no) удаляет заданную подсеть.

Синтаксис

[no] ipv6 prefix <IPV6-ADDR/LEN>

Параметры

<IPV6-ADDR/LEN> – IP-адрес и маска подсети, задаётся в виде X:X:X:X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-NETWORK

Пример

```
esr(config-object-group-network)# ipv6 prefix fc00::/126
```

17.7 ipv6 address-range

Команда используется для задания диапазона IPv6-адресов. Использование отрицательной формы команды (no) удаляет запись из конфигурируемого профиля.

Синтаксис

[no] ipv6 address-range <FROM-ADDR>[-<TO-ADDR>]

Параметры

<FROM-ADDR> – начальный IPv6-адрес диапазона адресов;

<TO-ADDR> – конечный IPv6-адрес диапазона адресов, опциональный параметр. Если параметр не указан, то командой задаётся одиночный IPv6-адрес.

Адреса задаются в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-NETWORK

Пример

```
esr(config-object-group-network)# ipv6 address-range fc00::1:1-fc00:1::32
```

17.8 port-range

Командой задаётся диапазон TCP/UDP-портов, относящихся к профилю.

Использование отрицательной формы команды (no) удаляет запись из конфигурируемого профиля.

Синтаксис

port-range <PORT>

no port-range [<PORT>]

Параметры

<PORT> – номер порта, принимает значение [1..65535].

Можно указать несколько портов перечислением через запятую «,» либо указать диапазон портов через «-». Пример записи: <PORT>, <PORT> или <PORT>-<PORT> или <PORT>-<PORT>, <PORT>-<PORT>.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-SERVICE

Пример

```
esr(config-object-group-service)# port-range 22
```

17.9 show object-group

Данная команда используется для просмотра информации о профилях IP-адресов и TCP/UDP-портов.

Синтаксис

```
show object-group <PROFILE_TYPE> [<NAME>]
```

Параметры

<PROFILE_TYPE> – тип профиля:

network – профиль IP-адресов;

service – профиль TCP/UDP-портов;

<NAME> – имя профиля, задаётся строкой до 31 символа, опциональный параметр. Если имя профиля не задано, то будет выведена информация по всем профилям IP-адресов и TCP/UDP-портов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show object-group network
Network                               Description
-----                               -
remote                               --
local                                --
tunnel                               --
esr# show object-group network remote
IP Addresses
-----
10.102.0.0/16

esr# show object-group service
Service                               Description
-----                               -
telnet                               --
ssh                                  --
dhcp_server                          --
dhcp_client                          --
ntp                                  --

esr# show object-group service ssh
Port ranges
-----
22
```

18 УПРАВЛЕНИЕ СПИСКАМИ КОНТРОЛЯ ДОСТУПА (ACL)

18.1 ip access-list extended

Данная команда используется для создания списка контроля доступа и перехода в режим конфигурирования списка.

Использование отрицательной формы команды (no) удаляет заданный список контроля доступа.

Синтаксис

```
[no] ip access-list extended <NAME>
```

Параметры

<NAME> – имя создаваемого списка контроля доступа, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# ip access-list extended acl-ssh-drop  
esr(config-acl)#
```

18.2 description

Данная команда используется для изменения описания конфигурируемого списка контроля доступа.

Использование отрицательной формы команды (no) удаляет установленное описание.

Синтаксис

```
description <DESCRIPTION>
```

```
no description
```

Параметры

<DESCRIPTION> – описание списка контроля доступа, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL

Пример

```
esr(config-acl)# description "Drop SSH traffic"
```

18.3 service-acl input

Данная команда используется для привязки указанного списка контроля доступа к конфигурируемому интерфейсу для фильтрации входящего трафика.

Использование отрицательной формы команды (no) удаляет привязку списка контроля доступа к данному интерфейсу.

Синтаксис

```
service-acl input <NAME>
no service-acl input
```

Параметры

<NAME> – имя списка контроля доступа, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

```
CONFIG-GI
CONFIG-PORT-CHANNEL
```

Пример

```
esr(config-if-gi)# service-acl input acl-ssh-drop
```

18.4 rule

Данная команда используется для создания правила и перехода в режим конфигурирования CONFIG-ACL-RULE. Правила обрабатываются устройством в порядке возрастания их номеров.

Использование отрицательной формы команды (no) удаляет указанное правило.

Синтаксис

```
[no] rule <ORDER>
```

Параметры

<ORDER> – номер правила, принимает значения [1..2000000].

Необходимый уровень привилегий

10

Командный режим

```
CONFIG-ACL
```

Пример

```
esr(config-acl)# rule 10
esr(config-acl-rule)#
```

18.5 enable

Данная команда используется для активирования правила.

Использование отрицательной формы команды (no) деактивирует правило.

Синтаксис

[no] enable

Параметры

Команда не содержит аргументов.

Значение по умолчанию

Правило выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
esr(config-acl-rule)# enable
```

18.6 match source-address

Данной командой устанавливаются IP-адреса отправителя, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

[no] match source-address { <ADDR> <WILDCARD> | any }

Параметры

<ADDR> – IP-адрес отправителя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<WILDCARD> – маска IP-адреса, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Биты маски, установленные в 0, задают биты IP-адреса, исключаемые из сравнения при поиске.

При указании значения «any» правило будет срабатывать для любого IP-адреса отправителя.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
esr(config-acl-rule)# match source-address 10.100.100.0 0.0.0.255
```

18.7 match source-mac

Данной командой устанавливаются MAC-адреса отправителя, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

[no] match source-mac <ADDR> <WILDCARD>

Параметры

<ADDR> – MAC-адрес отправителя, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

<WILDCARD> – маска MAC-адреса, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF]. Биты маски, установленные в 0, задают биты MAC-адреса, исключаемые из сравнения при поиске.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
esr(config-acl-rule)# match source-mac A8:F9:4B:AA:00:40 00:00:00:FF:FF:FF
```

18.8 match source-port

Данной командой устанавливается номер TCP/UDP-порта отправителя, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

match source-port { <PORT> | any }

no match source-port

Параметры

<PORT> – номер TCP/UDP-порта отправителя, принимает значения [1..65535]. При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта отправителя.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
esr(config-acl-rule)# match source-port any
```

18.9 match destination-address

Данной командой устанавливаются IP-адреса получателя, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

match destination-address { <ADDR> <WILDCARD> | any }

no match destination-address

Параметры

<ADDR> – IP-адрес получателя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<WILDCARD> – маска IP-адреса, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Биты маски, установленные в 0, задают биты IP-адреса, исключаемые из сравнения при поиске.

При указании значения «any» правило будет срабатывать для любого IP-адреса получателя.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
esr(config-acl-rule)# match destination-address 10.10.10.0 0.0.0.255
```

18.10 match destination-mac

Данной командой устанавливаются MAC-адреса получателя, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

[no] match destination-mac <ADDR> <WILDCARD>

Параметры

<ADDR> – MAC-адрес получателя, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

<WILDCARD> – маска MAC-адреса, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF]. Биты маски, установленные в 0, задают биты MAC-адреса, исключаемые из сравнения при поиске.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
esr(config-acl-rule)# match destination-mac A8:F9:4B:AA:00:41 00:00:00:00:00:FF
```

18.11 match destination-port

Данной командой устанавливается номер TCP/UDP-порта получателя, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) удаляет назначение.

Синтаксис

```
match destination-port <PORT>
no match destination-port
```

Параметры

<PORT> – номер TCP/UDP-порта получателя, принимает значения [1..65535]. При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта получателя.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
esr(config-acl-rule)# match destination-port 22
```

18.12 match protocol

Данной командой устанавливается имя IP-протокола, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match protocol <TYPE>
no match protocol
match protocol-id <ID>
no match protocol-id
```

Параметры

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre. При указании значения «any» правило будет срабатывать для любых протоколов;

<ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
esr(config-acl-rule)# match protocol tcp
```

18.13 match cos

Данной командой устанавливается значение 802.1p приоритета, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

match cos <COS>

no match cos

Параметры

<COS> – значение 802.1p приоритета, принимает значения [0..7].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
esr(config-acl-rule)# match cos 2
```

18.14 match dscp

Данной командой устанавливается значение кода DSCP, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

match dscp <DSCP>

no match dscp

Параметры

<DSCP> – значение кода DSCP, принимает значения [0..63].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
esr(config-acl-rule)# match dscp 55
```

18.15 match ip-precedence

Данной командой устанавливается значение кода IP Precedence, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

match ip-precedence <IPP>

no match ip-precedence

Параметры

<IPP> – значение кода IP Precedence, принимает значения [0..7].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
esr(config-acl-rule)# match ip-precedence 5
```

18.16 match vlan

Данной командой устанавливается значение идентификационного номера VLAN, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

match vlan <VID>

no match vlan

Параметры

<VID> – идентификационный номер VLAN, принимает значения [1...4094].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
esr(config-acl-rule)# match vlan 100
```

18.17 action

Данная команда используется для указания действия, которое должно быть применено для трафика, удовлетворяющего заданным критериям.

Использование отрицательной формы команды (no) удаляет назначенное действие.

Синтаксис

action <ACT>

no action

Параметры

<ACT> – назначаемое действие:

2. permit – прохождение трафика разрешается;
3. deny – прохождение трафика запрещается.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
esr(config-acl-rule)# action permit
```

18.18 *show ip access-list*

Данная команда используется для просмотра списков управления доступом.

Синтаксис

```
show ip access-list [ <NAME> [ <ORDER> ] ]
```

Параметры

<NAME> – имя списка управления доступом, задаётся строкой до 31 символа;

<ORDER> – номер правила, принимает значения [1..2000000]. При указании номера правила будет показана информация только по данному правилу.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show ip access-list
Name                                     Description
-----
acl-telnet-drop                         --
acl-ssh-drop                           Drop SSH traffic

esr# show ip access-list acl-ssh-drop
Index:                                  1
Matching pattern:
  Protocol:                            TCP(6)
  Source MAC address:                  any
  Source IP address:                   any
  Source port:                         any
  Destination MAC address:             any
  Destination IP address:              any
  Destination port:                    22
Action:                                Deny
Status:                                Enabled
-----
Index:                                  2
Matching pattern:
  Protocol:                            any
  Source MAC address:                  any
  Source IP address:                   any
  Destination MAC address:             any
  Destination IP address:              any
Action:                                Permit
```


Status:	Enabled

19 УПРАВЛЕНИЕ FIREWALL

19.1 security zone

Данная команда используется для создания зон безопасности и перехода в режим конфигурирования зоны.

Использование отрицательной формы команды (no) удаляет заданную зону безопасности.

Синтаксис

```
[no] security zone <NAME>
```

Параметры

<NAME> – имя создаваемой зоны безопасности, задаётся строкой до 12 символов. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все зоны безопасности.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

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

19.2 description

Данная команда используется для изменения описания конфигурируемой зоны безопасности.

Использование отрицательной формы команды (no) удаляет установленное описание.

Синтаксис

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

Параметры

<DESCRIPTION> – описание зоны безопасности, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZONE

Пример

```
esr(config-zone)# description "Trusted interfaces"
```

19.3 security-zone

Данная команда используется для добавления выбранного сетевого интерфейса в зону безопасности. Использование отрицательной формы команды (no) удаляет интерфейс из зоны.

Синтаксис

```
security-zone <NAME>  
no security-zone
```

Параметры

<NAME> – имя зоны безопасности, задаётся строкой до 12 символов.

Необходимый уровень привилегий

10

Командный режим

```
CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-E1  
CONFIG-MULTILINK  
CONFIG-VTI  
CONFIG-GRE  
CONFIG-IP4IP4  
CONFIG-BRIDGE
```

Пример

```
esr(config-if-gi)# security-zone trusted
```

19.4 ip firewall disable

Данная команда используется для отключения функции Firewall на сетевом интерфейсе.

Использование отрицательной формы команды (no) включает функцию Firewall на сетевом интерфейсе.

Синтаксис

```
[no] ip firewall disable
```

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GI
CONFIG-SUBIF
CONFIG-PORT-CHANNEL
CONFIG-VTI
CONFIG-GRE
CONFIG-IP4IP4
CONFIG-BRIDGE

Пример

```
esr(config-if-gi)# ip firewall disable
```

19.5 security zone-pair

Данная команда используется для создания группы правил для пары зон безопасности. Использование отрицательной формы команды (no) удаляет указанную группу правил.

Синтаксис

```
[no] security zone-pair <SOURCE-ZONE> <DESTINATION-ZONE>
```

Параметры

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик. На маршрутизаторе всегда существует зона безопасности с именем «self». Если в качестве получателя трафика выступает сам маршрутизатор, то есть трафик не является транзитным, то в качестве параметра указывается зона «self». Если при удалении используется значение параметра «all», то будут удалены все конфигурируемые пары зон безопасности.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# security zone-pair trusted self
```

19.6 rearrange

Данная команда меняет шаг между созданными правилами.

Синтаксис

```
rearrange <VALUE>
```

Параметры

<VALUE> – шаг между правилами, принимает значения [1..50].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZONE-PAIR
CONFIG-SNAT-RULESET
CONFIG-DNAT-RULESET

Пример

```
esr(config-zone-pair)# rearrange 10
```

19.7 renumber

Данная команда меняет номер правила.

Синтаксис

```
renumber rule <CUR_ORDER> <NEW_ORDER>
```

Параметры

<CUR_ORDER> – текущий номер правила, принимает значения [1..10000];

<NEW_ORDER> – новый номер правила, принимает значения [1..10000].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZONE-PAIR
CONFIG-SNAT-RULESET
CONFIG-DNAT-RULESET

Пример

```
esr(config-zone-pair)# renumber rule 13 100
```

19.8 rule

Данная команда используется для создания правила и перехода в командный режим SECURITY ZONE PAIR RULE. Правила обрабатываются устройством в порядке возрастания их номеров.

Использование отрицательной формы команды (no) удаляет указанное правило.

Синтаксис

```
[no] rule <ORDER>
```

Параметры

<ORDER> – номер правила, принимает значения [1..10000]. Если при удалении используется значение параметра «all», то будут удалены все правила для конфигурируемой пары зон безопасности.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZONE-PAIR

Пример

```
esr(config-zone-pair)# rule 10
esr(config-zone-rule)#
```

19.9 description

Данная команда используется для изменения описания.

Использование отрицательной формы команды (no) удаляет описание.

Синтаксис

description <DESCRIPTION>

no description

Параметры

<DESCRIPTION> – описание, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZONE-PAIR-RULE

Пример

```
esr(config-zone-rule)# description " Telnet traffic filtering"
```

19.10 enable

Данная команда используется для активирования правила.

Использование отрицательной формы команды (no) деактивирует правило.

Синтаксис

[no] enable

Параметры

Команда не содержит аргументов.

Значение по умолчанию

Правило выключено

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZONE-PAIR-RULE

Пример

```
esr(config-zone-rule)# enable
```

19.11 *match source-address*

Данной командой устанавливается профиль IP-адресов отправителя, для которых должно срабатывать правило.

При использовании параметра «not» (match not) правило будет срабатывать для IP-адресов отправителя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
[no] match [not] source-address <OBJ-GROUP-NETWORK-NAME>
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа. При указании значения «any» правило будет срабатывать для любого IP-адреса отправителя.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZONE-PAIR-RULE

Пример

```
esr(config-zone-rule)# match source-address remote
```

19.12 *match source-mac*

Данной командой устанавливается MAC-адрес отправителя, для которого должно срабатывать правило.

При использовании параметра «not» (match not) правило будет срабатывать для MAC-адресов отправителя, отличных от указанного.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
[no] match [not] source-mac <ADDR>
```

Параметры

<ADDR> – MAC-адрес отправителя, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZONE-PAIR-RULE

Пример

```
esr(config-zone-rule)# match source-mac A8:F9:4B:AA:00:40
```

19.13 *match source-port*

Данной командой устанавливается профиль TCP/UDP-портов отправителя, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для TCP/UDP-портов отправителя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] source-port <PORT-SET-NAME>
no match source-port
```

Параметры

<PORT-SET-NAME> – имя профиля TCP/UDP-портов, задаётся строкой до 31 символа. При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта отправителя.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZONE-PAIR-RULE

Пример

```
esr(config-zone-rule)# match source-port telnet
```

19.14 *match destination-address*

Данной командой устанавливается профиль IP-адресов получателя, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для IP-адресов получателя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] destination-address <OBJ-GROUP-NETWORK-NAME>
no match destination-address
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа. При указании значения «any» правило будет срабатывать для любого IP-адреса получателя.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZONE-PAIR-RULE

Пример

```
esr(config-zone-rule)# match destination-address local
```


19.15 *match destination-mac*

Данной командой устанавливается MAC-адрес получателя, для которого должно срабатывать правило.

При использовании параметра «not» (match not) правило будет срабатывать для MAC-адресов получателя, отличных от указанного.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
[no] match [not] destination-mac <ADDR>
```

Параметры

<ADDR> – MAC-адрес получателя, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZONE-PAIR-RULE

Пример

```
esr(config-zone-rule)# match destination-mac A8:F9:4B:AA:00:40
```

19.16 *match destination-port*

Данной командой устанавливается профиль TCP/UDP-портов получателя, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для TCP/UDP-портов получателя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) удаляет назначение.

Синтаксис

```
match [not] destination-port <PORT-SET-NAME>
```

```
no match destination-port
```

Параметры

<PORT-SET-NAME> – имя профиля TCP/UDP -портов, задаётся строка до 31 символа. При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта получателя.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZONE-PAIR-RULE

Пример

```
esr(config-zone-rule)# match destination-port ssh
```

19.17 *match protocol*

Данной командой устанавливается имя или номер IP-протокола, для которого должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для всех протоколов, кроме указанного.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] protocol <TYPE>
```

```
no match protocol
```

```
match [not] protocol-id <ID>
```

```
no match protocol-id
```

Параметры

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre.

При указании значения «any» правило будет срабатывать для любых протоколов;

<ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZONE-PAIR-RULE

Пример

```
esr(config-zone-rule)# match protocol udp
```

19.18 *match icmp*

Данная команда используется для настройки параметров протокола ICMP, если он выбран командой «match protocol». Данной командой устанавливается тип и код сообщений протокола ICMP, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для всех типов и кодов сообщений протокола ICMP, кроме указанных.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] icmp <ICMP_TYPE> <ICMP_CODE>
```

```
no match icmp
```

Параметры

<ICMP_TYPE> – тип сообщения протокола ICMP, принимает значения [0..255];

<ICMP_CODE> – код сообщения протокола ICMP, принимает значения [0..255]. При указании значения «any» правило будет срабатывать для любого кода сообщения протокола ICMP.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZONE-PAIR-RULE

Пример

```
esr(config-zone-rule)# match icmp 2 any
```

19.19 match destination-nat

Данной командой устанавливается ограничение, при котором правило будет срабатывать только для трафика, измененного сервисом трансляции IP-адресов и портов получателя.

При использовании параметра «not» правило будет срабатывать для трафика, не измененного сервисом трансляции IP-адресов и портов получателя. Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

[no] match [not] destination-nat

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZONE-PAIR-RULE

Пример

```
esr(config-zone-rule)# match destination-nat
```

19.20 action

Данная команда используется для указания действия, которое должно быть применено для трафика, удовлетворяющего заданным критериям.

Использование отрицательной формы команды (no) удаляет назначенное действие.

Синтаксис

action <АСТ>

no action

Параметры

<АСТ> – назначаемое действие:

4. permit – прохождение трафика разрешается;
5. deny – прохождение трафика запрещается;
6. reject – прохождение трафика запрещается, а также посылается отправителю ответ об ошибке;
7. netflow-sample – прохождение трафика разрешается и осуществляется экспорт статистики по протоколу Netflow;

8. sflow-sample – прохождение трафика разрешается и осуществляется экспорт статистики по протоколу sFlow.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZONE-PAIR-RULE

Пример

```
esr(config-zone-rule)# action permit
```

19.21 *ip firewall sessions counters*

Командой выполняется включение счетчиков сессий для NAT и Firewall. Счетчики увеличиваются только тогда, когда устанавливается новая сессия. Для установленных сессий увеличения значений счетчиков не происходит при прохождении пакетов. Включение счетчиков снижает производительность маршрутизатора.

Команды для просмотра счетчиков и сессий описаны в разделе 19.37 и 19.38.

Использование отрицательной формы команды (no) отключает счетчики сессий.

Синтаксис

[no] ip firewall sessions counters

Параметры

Команда не содержит аргументов.

Значение по умолчанию

Счетчики сессий отключены.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# ip firewall sessions counters
```

19.22 *ip firewall sessions max-expect*

Данной командой определяется размер таблицы сессий ожидающих обработки.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ip firewall sessions max-expect <COUNT>

no ip firewall sessions max-expect

Параметры

<COUNT> – размер таблицы, принимает значения [1..8553600].

Значение по умолчанию

256

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# ip firewall sessions max-expect 512
```

19.23 *ip firewall sessions max-tracking*

Данной командой определяется размер таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ip firewall sessions max-tracking <COUNT>

no ip firewall sessions max-tracking

Параметры

<COUNT> – размер таблицы, принимает значения [1..8553600].

Значение по умолчанию

512000

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# ip firewall sessions max-tracking 256000
```

19.24 *ip firewall sessions icmp-timeout*

Данной командой определяется время жизни ICMP-сессии, по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ip firewall sessions icmp-timeout <TIME>

no ip firewall sessions icmp-timeout

Параметры

<TIME> – время жизни ICMP-сессии, принимает значения в секундах [1..8553600].

Значение по умолчанию

30 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# ip firewall sessions icmp-timeout 60
```

19.25 *ip firewall sessions icmpv6-timeout*

Данной командой определяется время жизни ICMPv6-сессии, по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ip firewall sessions icmpv6-timeout <TIME>

no ip firewall sessions icmpv6-timeout

Параметры

<TIME> – время жизни ICMPv6-сессии, принимает значения в секундах [1..8553600].

Значение по умолчанию

30 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# ip firewall sessions icmpv6-timeout 60
```

19.26 *ip firewall sessions tcp-connect-timeout*

Данной командой определяется время жизни TCP-сессии в состоянии «соединение устанавливается», по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ip firewall sessions tcp-connect-timeout <TIME>

no ip firewall sessions tcp-connect-timeout

Параметры

<TIME> – время жизни TCP-сессии в состоянии "соединение устанавливается", принимает значения в секундах [1..8553600].

Значение по умолчанию

60 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# ip firewall sessions tcp-connect-timeout 120
```

19.27 *ip firewall sessions tcp-disconnect-timeout*

Данной командой определяется время жизни TCP-сессии в состоянии "соединение закрывается", по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ip firewall sessions tcp-disconnect-timeout <TIME>

no ip firewall sessions tcp-disconnect-timeout

Параметры

<TIME> – время жизни TCP-сессии в состоянии "соединение закрывается", принимает значения в секундах [1..8553600].

Значение по умолчанию

30 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# ip firewall sessions tcp-disconnect-timeout 10
```

19.28 *ip firewall sessions tcp-established-timeout*

Данной командой определяется время жизни TCP-сессии в состоянии "соединение установлено", по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ip firewall sessions tcp-established-timeout <TIME>

no ip firewall sessions tcp-established-timeout

Параметры

<TIME> – время жизни TCP-сессии в состоянии "соединение установлено", принимает значения в секундах [1..8553600].

Значение по умолчанию

120 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# ip firewall sessions tcp-established-timeout 3600
```

19.29 *ip firewall sessions tcp-latecome-timeout*

Данной командой определяется время ожидания, по истечении которого происходит фактическое удаление закрытой TCP-сессии из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ip firewall sessions tcp-latecome-timeout <TIME>

no ip firewall sessions tcp-latecome-timeout

Параметры

<TIME> – время ожидания, принимает значения в секундах [1..8553600].

Значение по умолчанию

120 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# ip firewall sessions tcp-latecome-timeout 10
```

19.30 *ip firewall sessions udp-assured-timeout*

Данной командой определяется время жизни UDP-сессии в состоянии "соединение подтверждено", по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ip firewall sessions udp-assured-timeout <TIME>

no ip firewall sessions udp-assured-timeout

Параметры

<TIME> – время жизни UDP-сессии в состоянии "соединение подтверждено ", принимает значения в секундах [1..8553600].

Значение по умолчанию

180 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# ip firewall sessions udp-assured-timeout 3600
```

19.31 *ip firewall sessions udp-wait-timeout*

Данной командой определяется время жизни UDP-сессии в состоянии «соединение не подтверждено», по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ip firewall sessions udp-wait-timeout <TIME>

no ip firewall sessions udp-wait-timeout

Параметры

<TIME> – время жизни UDP-сессии в состоянии «соединение не подтверждено», принимает значения в секундах [1..8553600].

Значение по умолчанию

30 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# ip firewall sessions udp-wait-timeout 60
```

19.32 *ip firewall sessions generic-timeout*

Данной командой определяется время жизни сессии для неподдерживаемых протоколов, по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

ip firewall sessions generic-timeout <TIME>

no ip firewall sessions generic-timeout

Параметры

<TIME> – время жизни сессии для неподдерживаемых протоколов, принимает значения в секундах [1..8553600].

Значение по умолчанию

60 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# ip firewall sessions generic-timeout 60
```

19.33 *show security zone*

Данная команда используется для просмотра интерфейсов, входящих в зону безопасности.

Синтаксис

show security zone [<NAME>]

Параметры

<NAME> – имя зоны, задаётся строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show security zone
Zone name      Interfaces
-----
trusted        gil/0/2-5, bridge 1
untrusted      gil/0/1, bridge 2
```

19.34 *show security zone-pair*

Данная команда используется для просмотра списка пар зон.

Синтаксис

```
show security zone-pair
```

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show security zone-pair
From zone      To zone
-----
trusted        untrusted
trusted        trusted
trusted        self
untrusted      self
```

19.35 *show security zone-pair configuration*

Данная команда используется для просмотра правил для пары зон безопасности.

Синтаксис

```
show security zone-pair configuration <DESTINATION-ZONE> <SOURCE-ZONE> [<ORDER>]
```

Параметры

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будет показана информация только по данному правилу.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show security zone-pair configuration trusted self
Order:          1
Description:    --
Matching pattern:
  Protocol:     tcp(6)
  Src-addr:     any
  src-port:     any
  Dest-addr:    any
  dest-port:    23
0              0
```

19.36 *show ip firewall counters*

Данная команда используется для просмотра статистики по пакетам, проходящим между зонами, для которых не установлена сессия.

Синтаксис

```
show ip firewall counters [ vrf <VRF> ] [ <DESTINATION-ZONE> <SOURCE-ZONE> [ <ORDER> ] ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены счетчики правил в указанном VRF;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик;

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будет показана информация только по данному правилу.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

esr# show ip firewall counters trusted self				
Zone-pair	Rule	Action	Pkts	Bytes
-----	-----	-----	-----	-----
any/any	default	deny	0	0
trusted/self	1	permit	0	0
trusted/trusted	1	permit	0	0

19.37 *show ip firewall sessions*

Данная команда используется для просмотра активных IP-сессий.

Синтаксис

```
show ip firewall sessions [ vrf <VRF> ] [ protocol <TYPE> ] [ inside-source <ADDR> ] [ outside-source <ADDR> ] [ inside-destination <ADDR> ] [ outside-destination <ADDR> ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены активные сессии в указанном VRF;

summary – выводит суммарную статистику по IP-сессиям;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

inside-source – команда для указания IP-адреса источника, приходящих пакетов;

inside-destination – команда для указания IP-адреса назначения, приходящих пакетов;

outside-source – команда для указания IP-адреса источника, отправляемых пакетов;

outside-destination – команда для указания IP-адреса назначения отправляемых пакетов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show ip firewall sessions
Prot Inside source   Inside destination  Outside source  Outside destination  Pkts  Bytes
-----
vrrp   4.4.4.4            224.0.0.18         4.4.4.4         224.0.0.18         --    --
```

19.38 show ipv6 firewall counters

Данная команда используется для просмотра статистики по пакетам, проходящим между зонами, для которых не установлена сессия.

Синтаксис

```
show ipv6 firewall counters [ vrf <VRF> ] [ <DESTINATION-ZONE> <SOURCE-ZONE> [ <ORDER> ] ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены счетчики правил в указанном VRF;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик;

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будет показана информация только по данному правилу.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show ipv6 firewall counters trusted self
Zone-pair      Rule      Action      Pkts      Bytes
-----
any/any        default   deny        0         0
trusted/self   1         permit      0         0
trusted/trusted 1         permit      0         0
```

19.39 show ipv6 firewall sessions

Данная команда используется для просмотра активных IPv6-сессий.

Синтаксис

```
show ipv6 firewall sessions [ vrf <VRF> ] [ summary ] [ protocol <TYPE> ] [ inside-source <IPV6-ADDR> ] [ outside-source <IPV6-ADDR> ] [ inside-destination <IPV6-ADDR> ] [ outside-destination <IPV6-ADDR> ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены активные сессии в указанном VRF;

summary – выводит суммарную статистику по IPv6-сессиям;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<IPv6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

inside-source – команда для указания IPv6-адреса источника, приходящих пакетов;

inside-destination – команда для указания IPv6-адреса назначения, приходящих пакетов;

outside-source – команда для указания IPv6-адреса источника, отправляемых пакетов;

outside-destination – команда для указания IPv6-адреса назначения отправляемых пакетов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

esr# show ipv6 firewall sessions						
Prot	Inside source	Inside destination	Outside source	Outside destination	Pkts	Bytes
icmp6	fc00::2	fc00::2	fc00::2	fc00::2	--	--
icmp6	fc00::2	fc00::1	fc00::2	fc00::1	--	--

19.40 clear ip firewall counters

Данной командой осуществляется сброс счетчиков правил Firewall.

Синтаксис

clear ip firewall counters [vrf <VRF>] [<DESTINATION-ZONE> <SOURCE-ZONE> [<ORDER>]]

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут очищены счетчики правил в указанном VRF;

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будут очищены счетчики только по данному правилу.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
esr# clear ip firewall counters trusted self
```

19.41 *clear ip firewall sessions*

Данной командой осуществляется удаление активных IP-сессий.

Синтаксис

```
clear ip firewall sessions [ vrf <VRF> ] [ protocol <TYPE> ] [ inside-source <ADDR> ] [ outside-source <ADDR> ] [ inside-destination <ADDR> ] [ outside-destination <ADDR> ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут удалены активные сессии в указанном VRF;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

inside-source – команда для указания IPv6-адреса источника, приходящих пакетов;

inside-destination – команда для указания IPv6-адреса назначения, приходящих пакетов;

outside-source – команда для указания IPv6-адреса источника, отправляемых пакетов;

outside-destination – команда для указания IPv6-адреса назначения отправляемых пакетов.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
esr# clear ip firewall sessions vrf VRF1
```

19.42 *clear ipv6 firewall counters*

Данной командой осуществляется сброс счетчиков правил Firewall.

Синтаксис

```
clear ipv6 firewall counters [ vrf <VRF> ] [ <DESTINATION-ZONE> <SOURCE-ZONE> [<ORDER>] ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут очищены счетчики правил в указанном VRF;

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будут очищены счетчики только по данному правилу.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
esr# clear ipv6 firewall counters trusted self
```

19.43 *clear ipv6 firewall sessions*

Данной командой осуществляется удаление активных IPv6-сессий.

Синтаксис

```
clear ipv6 firewall sessions [ vrf <VRF> ] [ protocol <TYPE> ] [ inside-source <IPv6-ADDR> ] [ outside-source <IPv6-ADDR> ] [ inside-destination <IPv6-ADDR> ] [ outside-destination <IPv6-ADDR> ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут удалены активные сессии в указанном VRF;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<IPv6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

inside-source – команда для указания IPv6-адреса источника, приходящих пакетов;

inside-destination – команда для указания IPv6-адреса назначения, приходящих пакетов;

outside-source – команда для указания IPv6-адреса источника, отправляемых пакетов;

outside-destination – команда для указания IPv6-адреса назначения отправляемых пакетов.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
esr# clear ipv6 firewall sessions vrf VRF1
```


20 УПРАВЛЕНИЕ NAT

20.1 *nat source*

Данная команда позволяет войти в режим настройки сервиса трансляции адресов отправителя (SNAT, Source NAT).

Использование отрицательной формы команды (no) удаляет настройки сервиса трансляции адресов отправителя (SNAT, Source NAT).

Синтаксис

[no] nat source

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# nat source
esr(config-snat)#
```

20.2 *nat destination*

Данная команда позволяет войти в режим настройки сервиса трансляции адресов получателя (DNAT, Destination NAT).

Использование отрицательной формы команды (no) удаляет настройки сервиса трансляции адресов получателя (DNAT, Destination NAT).

Синтаксис

[no] nat destination

Параметры

Команда не содержит аргументов.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# nat destination
esr(config-dnat)#
```

20.3 ruleset

Данная команда используется для создания группы правил с определённым именем и перехода в командный режим SNAT RULESET или DNAT RULESET.

Использование отрицательной формы команды (no) удаляет заданную группу правил.

Синтаксис

[no] ruleset <NAME>

Параметры

<NAME> – имя группы правил, задаётся строкой до 31 символа. Если использовать команду для удаления, то при указании значения «all» будут удалены все группы правил.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT

CONFIG-SNAT

Пример

```
esr(config-snat)# ruleset wan
esr(config-snat-ruleset)#
```

20.4 pool

Команда создаёт и назначает пул IP-адресов и TCP/UDP-портов с определённым именем для сервиса NAT и меняет командный режим на SNAT POOL или DNAT POOL.

Примечание. Если пул используется в какой-либо группе правил, то его удалять нельзя.

Использование отрицательной формы команды (no) удаляет заданный пул NAT-адресов.

Синтаксис

[no] pool <NAME>

Параметры

<NAME> – имя пула NAT-адресов, задаётся строкой до 31 символа. Если использовать команду для удаления, то при указании значения «all» будут удалены все пулы IP-адресов и TCP/UDP-портов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT

CONFIG-SNAT

Пример

```
esr(config-snat)# pool nat
esr(config-snat-pool)#
```

20.5 ip nat proxy-arp

Данная команда позволяет маршрутизатору отвечать на ARP-запросы IP-адресов из указанного пула. Функция необходима для того, чтобы не назначать все IP-адреса из пула трансляции на интерфейсе.

Синтаксис

```
ip nat proxy-arp <OBJ-GROUP-NETWORK-NAME>
no ip nat proxy-arp
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

Значение по умолчанию

Функция NAT Proxy ARP отключена.

Необходимый уровень привилегий

10

Командный режим

```
CONFIG-GI
CONFIG-SUBIF
CONFIG-PORT-CHANNEL
CONFIG-BRIDGE
```

Пример

```
esr(config-if-gi)# ip nat proxy-arp nat-pool
```

20.6 from

Данной командой ограничивается область применения группы правил. Правила будут применяться только для трафика, идущего из определенной зоны или интерфейса.

Использование отрицательной формы команды (no) удаляет ограничение области применения группы правил.

Синтаксис

```
from { zone <NAME> | interface <IF> | default }
no from
```

Параметры

<NAME> – имя зоны изоляции;

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе 3.3.

default – обозначает группу правил для всего трафика, источник которого не попал под критерии других групп правил.

Примечание. Группа правил со значением «default» параметра «from» может быть только одна.

Значение по умолчанию

none

Необходимый уровень привилегий

Командный режим

CONFIG-DNAT-RULESET

Пример

```
esr(config-dnat-ruleset)# from zone untrusted
```

20.7 to

Данной командой ограничивается область применения группы правил. Правила будут применяться только для трафика, идущего в определенную зону или интерфейс.

Использование отрицательной формы команды (no) удаляет ограничение области применения группы правил.

Синтаксис

to { zone <NAME> | interface <IF> | default }

no to

Параметры

<NAME> – имя зоны изоляции;

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе 3.3.

default – обозначает группу правил для всего трафика, место назначения которого не попало под критерии других групп правил.

Примечание. Группа правил со значением «default» параметра «to» может быть только одна.

Значение по умолчанию

None

Необходимый уровень привилегий

10

Командный режим

CONFIG-SNAT-RULESET

Пример

```
esr(config-snat)# ruleset test
esr(config-snat-ruleset)# to interface gigabitethernet 1/0/1
```

20.8 description

Данной командой задаётся описание.

Использование отрицательной формы команды (no) удаляет описание.

Синтаксис

description <DESCRIPTION>

no description

Параметры

<DESCRIPTION> – описание, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULESET

CONFIG-SNAT-RULESET

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

CONFIG-DNAT-POOL

CONFIG-SNAT-POOL

Пример

```
esr(config-snat-ruleset)# description "test ruleset"
```

20.9 rule

Данной командой создается правило с определённым номером и устанавливается режим командного интерфейса SNAT RULE или DNAT RULE. Правила обрабатываются устройством в порядке возрастания номеров правил.

Использование отрицательной формы команды (no) удаляет правило по номеру либо все правила.

Синтаксис

[no] rule <ORDER>

Параметры

<ORDER> – номер правила, принимает значения [1 .. 10000]. Если использовать команду для удаления, то при указании значения «all» будут удалены все правила.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULESET

CONFIG-SNAT-RULESET

Пример

```
esr(config-snat-ruleset)# rule 10
esr(config-snat-rule)#
```

20.10 enable

Данной командой активируется конфигурируемое правило.

Использование отрицательной формы команды (no) деактивирует использование правила.

Синтаксис

[no] enable

Параметры

Команда не содержит аргументов.

Значение по умолчанию

Правило выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Пример

```
esr(config-snat-rule)# enable
```

20.11 *match source-address*

Данной командой устанавливается профиль IP-адресов отправителя, для которых должно срабатывать правило.

При использовании команды «not» правило будет срабатывать для IP-адресов отправителя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет установленное действие.

Синтаксис

```
match [not] source-address <OBJ-GROUP-NETWORK-NAME>
```

```
no match source-address
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого IP-адреса отправителя.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Пример

```
esr(config-snat-rule)# match source-address local
```

20.12 *match source-port*

Данной командой устанавливается профиль TCP/UDP портов отправителя, для которых должно срабатывать правило.

При использовании команды «not» правило будет срабатывать для TCP/UDP портов отправителя, которые не входят в указанный профиль. Использование отрицательной формы команды (no) отменяет установленное действие.

Синтаксис

```
match [not] source-port <PORT-SET-NAME>
no match source-port
```

Параметры

<PORT-SET-NAME> – имя профиля порта, задаётся строкой до 31 символа. При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта отправителя.

Необходимый уровень привилегий

10

Командный режим

```
CONFIG-DNAT-RULE
CONFIG-SNAT-RULE
```

Пример

```
esr(config-snat-rule)# match source-port telnet
```

20.13 *match destination-address*

Данной командой устанавливается профиль IP-адресов получателя, для которых должно срабатывать правило.

При использовании команды «not» правило будет срабатывать для IP-адресов получателя, которые не входят в указанный профиль. Использование отрицательной формы команды (no) отменяет установленное действие.

Синтаксис

```
match [not] destination-address <OBJ-GROUP-NETWORK-NAME>
no match destination-address
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адреса, задаётся строкой до 31 символа. При указании значения «any» правило будет срабатывать для любого IP-адреса получателя.

Необходимый уровень привилегий

10

Командный режим

```
CONFIG-DNAT-RULE
CONFIG-SNAT-RULE
```

Пример

```
esr(config-snat-rule)# match destination-address remote
```

20.14 *match destination-port*

Данной командой устанавливается профиль TCP/UDP портов получателя, для которых должно срабатывать правило.

При использовании команды «not» правило будет срабатывать для TCP/UDP портов получателя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет установленное действие.

Синтаксис

```
match [not] destination-port <PORT-SET-NAME>
```

```
no match destination-port
```

Параметры

<PORT-SET-NAME> – имя профиля порта, задаётся строкой до 31 символа. При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта получателя.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Пример

```
esr(config-snat-rule)# match destination-port ssh
```

20.15 *match protocol*

Данной командой устанавливается имя или номер IP-протокола, для которого должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для всех протоколов, кроме указанного.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] protocol <TYPE>
```

```
no match protocol
```

```
match [not] protocol-id <ID>
```

```
no match protocol-id
```

Параметры

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre.

При указании значения «any» правило будет срабатывать для любых протоколов;

<ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Пример

```
esr(config-snat-rule)# match protocol udp
```

20.16 *match icmp*

Данная команда используется для настройки параметров протокола ICMP, если он выбран командой «match protocol». Командой устанавливается тип и код сообщений протокола ICMP, для которых должно срабатывать правило.

При использовании команды «not» правило будет срабатывать для всех типов и кодов сообщений протокола ICMP, кроме указанных.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

match [not] icmp <ICMP_TYPE> <ICMP_CODE>

no match icmp

Параметры

<ICMP_TYPE> – тип сообщения протокола ICMP, принимает значения [0 ..255];

<ICMP_CODE> – код сообщения протокола ICMP, принимает значения [0 ..255]. При указании значения «any» правило будет срабатывать для любого кода сообщения протокола ICMP.

Значение по умолчанию

any any

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Пример

```
esr(config-snat-rule)# match icmp 2 any
```

20.17 *action source-nat*

Данной командой назначается тип действия «трансляция адреса и порта отправителя» и параметры трансляции для трафика, удовлетворяющего критериям, заданным командами «match».

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

action source-nat { off | pool <NAME> | netmap <ADDR/LEN> | interface [FIRST_PORT – LAST_PORT] }

no action source-nat

Параметры

`off` – трансляция отключена. Трафик, попадающий под заданные критерии, не будет изменен;

`pool <NAME>` – задаёт пул IP-адресов и/или TCP/UDP портов. У трафика, попадающего под заданные критерии, будет изменен IP-адрес и/или TCP/UDP порт отправителя на значения, выбранные из пула;

`netmap <ADDR/LEN>` – задаёт префикс адреса и маску подсети для трансляции. У трафика, попадающего под заданные критерии, будет изменен IP-адрес отправителя на IP-адрес из указанной подсети. Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

`interface [FIRST_PORT – LAST_PORT]` – задаёт трансляцию в IP-адрес интерфейса. У трафика, попадающего под заданные критерии, будет изменён IP-адрес отправителя на IP-адрес интерфейса, через который данный трафик был получен. Если дополнительно задан диапазон TCP/UDP портов, то трансляция будет происходить только для TCP/UDP портов отправителя, входящих в указанный диапазон портов.

Значение по умолчанию

`none`

Необходимый уровень привилегий

10

Командный режим

`CONFIG-SNAT-RULE`

Пример

```
esr(config-snat-rule)# action source-nat netmap 10.10.10.0/24
```

20.18 *action destination-nat*

Данной командой выполняется трансляция адреса и порта получателя для трафика, удовлетворяющего заданным критериям.

Использование отрицательной формы команды (`no`) устанавливает значение по умолчанию.

Синтаксис

`action destination-nat { off | pool <NAME> | netmap <ADDR/LEN> }`

`no action destination-nat`

Параметры

`off` – трансляция отключена. Трафик, попадающий под заданные критерии, не будет изменен;

`pool <NAME>` – имя пула, содержащего набор IP-адресов и/или TCP/UDP портов. У трафика, попадающего под заданные критерии, будет изменен IP-адрес и TCP/UDP порт получателя на значения, выбранные из пула;

`netmap <ADDR/LEN>` – IP-адрес и маска подсети, используемые при трансляции. У трафика, попадающего под заданные критерии, будет изменен IP-адрес получателя на IP-адрес из указанной подсети. Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

Значение по умолчанию

`none`

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULE

Пример

```
esr(config-dnat-rule)# action destination-nat netmap 10.10.10.0/24
```

20.19 *ip address*

Данной командой устанавливается внутренний IP-адрес, на который будет заменяться IP-адрес получателя.

Использование отрицательной формы команды (no) удаляет заданный IP-адрес.

Синтаксис

ip address <ADDR>

no ip address

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-POOL

Пример

```
esr(config-dnat-pool)# ip address 10.10.10.10
```

20.20 *ip port*

Данной командой устанавливается внутренний TCP/UDP порт, на который будет заменяться TCP/UDP порт получателя.

Использование отрицательной формы команды (no) удаляет заданный TCP/UDP порт.

Синтаксис

ip port <PORT>

no ip port

Параметры

<PORT> – TCP/UDP порт, принимает значения [1..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-POOL

Пример

```
esr(config-dnat-pool)# ip port 5000
```

20.21 *ip address-range*

Данной командой задаётся диапазон внешних IP-адресов, на которые будет заменяться IP-адрес отправителя.

Использование отрицательной формы команды (no) удаляет заданный диапазон адресов.

Синтаксис

```
ip address-range <IP>[-<ENDIP>]
```

```
no ip address-range
```

Параметры

<IP> – IP-адрес начала диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ENDIP> – IP-адрес конца диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Если не указывать IP-адрес конца диапазона, то в качестве IP-адреса для трансляции используется только IP-адрес начала диапазона.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SNAT-POOL

Пример

```
esr(config-snat-pool)# ip address-range 10.10.10.1-10.10.10.20
```

20.22 *ip port-range*

Данной командой задаётся диапазон внешних TCP/UDP портов, на которые будет заменяться TCP/UDP порт отправителя.

Использование отрицательной формы команды (no) удаляет заданный диапазон портов.

Синтаксис

```
ip port-range <PORT>[-<ENDPORT>]
```

```
no ip port-range
```

Параметры

<PORT> – TCP/UDP порт начала диапазона, принимает значения [1..65535];

<ENDPORT> – TCP/UDP порт конца диапазона, принимает значения [1..65535]. Если не указывать TCP/UDP порт конца диапазона, то в качестве TCP/UDP порта для трансляции используется только TCP/UDP порт начала диапазона.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SNAT-POOL

Пример

```
esr(config-snat-pool)# ip port-range 20-100
```

20.23 *persistent*

Командой выполняется включение функции NAT persistent.

NAT persistent позволяет приложениям использовать STUN (session traversal utilities for NAT – утилиты проброса сессий для NAT) для установления соединения с устройствами, находящимися за шлюзом NAT. При этом гарантируется, что запросы от одного и того же внутреннего адреса транслируются в один и тот же внешний адрес.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

[no] persistent

Параметры

Команда не содержит аргументов.

Значение по умолчанию

Функция NAT persistent отключена.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SNAT-POOL

Пример

```
esr(config-snat-pool)# persistent
```

20.24 *show ip nat ruleset*

Данной командой выполняется просмотр всех или выбранных групп правил, используемых функцией NAT.

Синтаксис

show ip nat <TYPE> ruleset [<NAME>]

Параметры

<TYPE> – тип группы правил:

source – группа правил для трансляции IP-адреса и TCP/UDP порта отправителя;

destination – группа правил для трансляции IP-адреса и TCP/UDP порта получателя;

[NAME] – имя группы правил, опциональный параметр. Если имя не задано – будет выведен список всех групп правил.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show ip nat source rulesets
Rulesets
~~~~~
ID      Name                               To                               Description
-----
0       factory                             zone 'untrusted'
1       test                                gigabitethernet
                                      1/0/1
esr# show ip nat source rulesets factory
Ruleset:          factory
Description:
To:               none
Rules:
-----
Order:           10
Description:      replace 'source ip' by outgoing interface ip address
Matching pattern:
  Protocol:       any(0)
  Src-addr:       any
  Dest-addr:      any
Action:           interface port any
Status:           Enabled
-----
```

20.25 *show ip nat pool*

Данная команда используется для просмотра пулов внутренних и внешних IP-адресов и TCP/UDP портов.

Синтаксис

show ip nat <TYPE> pools

Параметры

<TYPE> – тип пулов, для просмотра:

source – внешние IP-адреса и TCP/UDP порты;

destination – внутренние IP-адреса и TCP/UDP порты.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show nat source pools
Pools
~~~~~
```

ID	Name	Ip address	Port range	Description	Persistent
0	outside	25.56.48.11	2000 – 3000	outside-pool	false

20.26 *show nat proxy-arp*

Данная команда используется для просмотра настроек NAT Proxy ARP.

Синтаксис

```
show nat proxy-arp
```

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

esr# show nat proxy-arp	
Interface	IP address range
-----	-----
gi1/0/15	115.0.0.15-115.0.0.100

20.27 *show ip nat translations*

Данная команда используется для просмотра сессий трансляции. Для просмотра информации о статистике необходимо включить счетчики (раздел 19.21).

Синтаксис

```
show ip nat translations [ vrf <VRF> ] [ protocol <TYPE> ] [ inside-source <ADDR> ] [ outside-source <ADDR> ] [ inside-destination <ADDR> ] [ outside-destination <ADDR> ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены сессии трансляций в указанном VRF;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

Для Source NAT:

inside-source – команда для указания IP-адреса источника до трансляции;

inside-destination – команда для указания IP-адреса назначения на входе в маршрутизатор;

outside-source – команда для указания IP-адреса источника после трансляции;

outside-destination – команда для указания IP-адреса назначения на выходе из маршрутизатора.

Для Destination NAT

inside-source – команда для указания IP-адреса источника на выходе из маршрутизатора;

inside-destination – команда для указания IP-адреса назначения после трансляции;

outside-source – команда для указания IP-адреса источника на входе в маршрутизатор;

outside-destination – команда для указания IP-адреса назначения до трансляции.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример 1

Source NAT

esr#	show ip nat translations					
Prot	Inside source	Inside destination	Outside source	Outside destination	Pkts	Bytes
----	-----	-----	-----	-----	-----	-----
icmp	115.0.0.10	1.1.0.2	1.1.0.24	1.1.0.2	3	252

Пример 2

Destination NAT

esr#	show ip nat translations					
Prot	Inside source	Inside destination	Outside source	Outside destination	Pkts	Bytes
----	-----	-----	-----	-----	-----	-----
icmp	1.1.0.2	115.0.0.10	1.1.0.2	1.1.0.16	--	--

21 НАСТРОЙКИ IPSEC VPN

21.1 Управление VPN. Настройки IKE

21.1.1 *security ike proposal*

Данной командой создается профиль протокола IKE (Internet Key Exchange), который включает в себя параметры алгоритмов шифрования и аутентификации, метода Диффи-Хеллмана, которые будут использоваться при согласовании параметров IKE со встречной стороной VPN соединения при создании Security Association (SA). Кроме того, профиль задаёт предельное время действия SA. Использование отрицательной формы команды (no) удаляет заданный профиль.

Синтаксис

[no] security ike proposal <NAME>

Параметры

<NAME> – имя профиля протокола IKE, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IKE профили.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# security ike proposal ike_prop1
esr(config-ike-proposal)#
```

21.1.2 *security ike policy*

Данной командой создается политика IKE, которая включает в себя профили протокола IKE, общий секретный ключ для аутентификации и режим согласования первой фазы протокола IKE. Использование отрицательной формы команды (no) удаляет указанную политику. Команда устанавливает режим командной строки SECURITY IKE POLICY.

Синтаксис

[no] security ike policy <NAME>

Параметры

<NAME> – имя политики IKE, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IKE политики.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# security ike policy ike_poll  
esr(config-ike-policy)#
```

21.1.3 *security ike gateway*

Данной командой осуществляется переход в командный режим конфигурирования шлюза IKE SECURITY IKE GATEWAY. Если шлюз IKE с указанным именем не существует в конфигурации, то он будет создан. Параметры шлюза включают в себя VTI-интерфейс, в который будет направляться трафик, политика и версия протокола IKE, а также режим перенаправления трафика в туннель.

Использование отрицательной формы команды (no) удаляет шлюз протокола IKE.

Синтаксис

[no] security ike gateway <NAME>

Параметры

<NAME> – имя шлюза протокола IKE, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IKE шлюзы.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# security ike gateway ike_gw1  
esr(config-ike-gw)#
```

21.1.4 *description*

Команда используется для изменения описания профиля, политики или шлюза протокола IKE.

Использование отрицательной формы команды (no) удаляет описание.

Синтаксис

description <DESCRIPTION>

no description

Параметры

<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-PROPOSAL

CONFIG-IKE-POLICY

CONFIG-IKE-GATEWAY

Пример

```
esr(config-ike-proposal)# description "my proposal"
```

21.1.5 authentication method

Данной командой выбирается метод аутентификации по ключу для IKE-соединения. Аутентификация сообщений по ключу используется при установлении IKE-соединения, ключ задаётся в политике (pre-shared key, см п.п. 21.1.11). После установления IKE-соединения аутентификация сообщений осуществляется при помощи алгоритма хеширования.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

authentication method pre-shared-key

no authentication method

Параметры

pre-shared-key – метод аутентификации, использующий предварительно полученные ключи шифрования.

Значение по умолчанию

pre-shared-key

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-PROPOSAL

Пример

```
esr(config-ike-proposal)# authentication method pre-shared-key
```

21.1.6 encryption algorithm

Данной командой выбирается алгоритм шифрования, используемый при установлении IKE-соединения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

encryption algorithm <ALGORITHM>

no encryption algorithm

Параметры

<ALGORITHM> – идентификатор протокола шифрования, принимает значения: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.

Значение по умолчанию

3des

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-PROPOSAL

Пример

```
esr(config-ike-proposal)# encryption algorithm aes128
```

21.1.7 *dh-group*

Данной командой устанавливается номер группы метода Диффи-Хеллмана. Номер группы определяет уровень защищенности IKE-соединения при обмене ключами – защищенность возрастает с ростом номера группы, но увеличивается и время установления соединения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

dh-group <DH-GROUP>

no dh-group

Параметры

<DH-GROUP> – номер группы Диффи-Хеллмана, принимает значения [1, 2, 5, 14, 15, 16, 17, 18].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-PROPOSAL

Пример

```
esr(config-ike-proposal)# dh-group 5
```

21.1.8 *authentication algorithm*

Данной командой устанавливается алгоритм аутентификации, который используется для аутентификации сообщений установленного IKE-соединения. При установлении IKE-соединения используется аутентификация сообщений по ключу (authentication, см. п. 21.1.5).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

authentication algorithm <ALGORITHM>

no authentication algorithm

Параметры

<ALGORITHM> – алгоритм аутентификации, принимает значения: md5, sha1, sha2-256, sha2-384, sha2-512.

Значение по умолчанию

sha1

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-PROPOSAL

Пример

```
esr(config-ike-proposal)# authentication algorithm md5
```

21.1.9 *lifetime*

Данной командой задаётся время жизни соединения протокола IKE.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

lifetime <SEC>

no lifetime

Параметры

<SEC> – период времени, принимает значения [1140 ..86400] секунд.

Значение по умолчанию

3600 сек.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-PROPOSAL

Пример

```
esr(config-ike-proposal)# lifetime 21600
```

21.1.10 *mode*

Данной командой устанавливается режим согласования первой фазы протокола IKE.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

mode <MODE>

no mode

Параметры

<MODE> – режим первой фазы IKE, принимает значения:

main – состоит из трех двусторонних обменов между отправителем и получателем:

1. Во время первого обмена согласуются алгоритмы аутентификации и шифрования, которые будут использоваться для защиты IKE-соединения, посредством сопоставления профилей протокола IKE каждого узла;
2. Используя алгоритм Диффи-Хеллмана, стороны обмениваются общим секретным ключом. Также узлы проверяют идентификацию друг друга путем передачи и подтверждения последовательности псевдослучайных чисел;
3. Проверяется идентичность противоположной стороны. В результате выполнения основного режима создается безопасный канал для второй фазы протокола IKE;

aggressive – этот режим обходится меньшим числом обменов и, соответственно, числом пакетов.

- В первом сообщении (от инициатора) отправляется следующая информация, которая используется для установления IKE-соединения: предложение параметров SA, инициирование обмена Диффи-Хеллмана, отправление псевдослучайного числа и идентификатора пакета.
- Во втором сообщении ответчик принимает SA, аутентифицирует инициатора, отправляет псевдослучайное число и свой IKE -идентификатор.
- В третьем сообщении инициатор аутентифицирует ответчика и подтверждает обмен.

Значение по умолчанию

main

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-POLICY

Пример

```
esr(config-ike-policy)# mode aggressive
```

21.1.11 *pre-shared-key*

Данной командой устанавливается общий секретный ключ для аутентификации, должен совпадать у обеих сторон, устанавливающих туннель.

Использование отрицательной формы команды (no) удаляет установленный ключ.

Синтаксис

```
pre-shared-key { ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> } | hexadecimal { <HEX> | encrypted <ENCRYPTED-HEX> } }
```

```
no pre-shared-key
```

Параметры

<TEXT> – строка [1..64] ASCII символов;

<HEX> – число размером [1..32] байт задаётся строкой [2..128] символов в шестнадцатеричном формате(0xYYYY...) или (YYYY...).

<ENCRYPTED-TEXT> – зашифрованный пароль размером [1..32] байт, задаётся строкой [2..128] символов;

<ENCRYPTED-HEX> – зашифрованное число размером [2..64] байт, задаётся строкой [2..256] символов.

Значение по умолчанию

none

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-POLICY

Пример

```
esr(config-ike-policy)# pre-shared-key hexadecimal abc123
```

21.1.12 *proposal*

Данной командой устанавливается привязка профиля протокола IKE к политике.

Использование отрицательной формы команды (no) удаляет привязку профиля протокола IKE.

Синтаксис

[no] proposal <NAME>

Параметры

<NAME> – имя профиля протокола IKE, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-POLICY

Пример

```
esr(config-ike-policy)# proposal ike_prop1
```

21.1.13 *version*

Данной командой задаётся версия протокола IKE.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

version <VERSION>

no version

Параметры

<version> – версия IKE-протокола: v1-only или v2-only.

Значение по умолчанию

v1-only

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
esr(config-ike-gw)# version v2-only
```

21.1.14 *ike-policy*

Данной командой устанавливается привязка политики протокола IKE к шлюзу.

Использование отрицательной формы команды (no) удаляет привязку политики.

Синтаксис

[no] ike-policy <NAME>

Параметры

<NAME> – имя политики протокола IKE, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
esr(config-ike-gw)# ike-policy ike_poll
```

21.1.15 *dead-peer-detection action*

Данной командой устанавливается действие, которое должно предпринять устройство, в случае обнаружения недоступности IPSec соседа, механизмом Dead Peer Detection.

Dead Peer Detection (DPD) – это механизм проверки состояния и доступности соседних устройств. Механизм периодически отправляет R-U-THERE сообщения (для IKE версии 1) или пустые INFORMATIONAL сообщения (для IKE версии 2) для проверки доступности IPSec соседа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

dead-peer-detection action <MODE>

no dead-peer-detection action

Параметры

<MODE> – режим работы DPD:

restart – соединение переустанавливается;

clear – соединение останавливается;

hold – соединение поддерживается;

none – механизм выключен, никаких действий не предпринимается.

Значение по умолчанию

none

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
esr(config-ike-gw)# dead-peer-detection action clear
```

21.1.16 *dead-peer-detection interval*

Данной командой устанавливается интервал между отправкой сообщений механизмом DPD. Механизм DPD описан в разделе **21.1.15**.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

dead-peer-detection interval <SEC>

no dead-peer-detection interval

Параметры

<SEC> – интервал между отправкой сообщений механизмом DPD, принимает значения [1..180] секунд.

Значение по умолчанию

2 секунды

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
esr(config-ike-gw)# dead-peer-detection interval 15
```

21.1.17 *dead-peer-detection timeout*

Данной командой задаётся таймаут ответа на сообщения, отправленные механизмом DPD.

Механизм DPD описан в разделе **Ошибка! Источник ссылки не найден..**

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

dead-peer-detection timeout <SEC>

no dead-peer-detection timeout

Параметры

<SEC> – период времени для ответа на сообщения механизма DPD, принимает значения [1..180] секунд.

Значение по умолчанию

30 секунд.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
esr(config-ike-gw)# dead-peer-detection timeout 60
```

21.1.18 *mode*

Данной командой устанавливается режим перенаправления трафика в туннель.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

mode <MODE>

no mode

Параметры

<MODE> – режим перенаправления трафика в туннель, принимает значения:

policy-based – трафик перенаправляется на основе принадлежности к указанным в политиках подсетям¹;

route-based – трафик перенаправляется на основе маршрутов, шлюзом у которых является туннельный интерфейс.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
esr(config-ike-gw)# mode route-based
```

21.1.19 *bind-interface vti*

Данной командой указывается туннельный интерфейс, через который будет проходить трафик в режиме туннеля «route-based».

¹ В текущей версии ПО не поддерживается.

Использование отрицательной формы команды (no) удаляет привязку к туннельному интерфейсу.

Синтаксис

```
bind-interface vti <VTI>
no bind-interface vti
```

Параметры

<VTI> – идентификационный номер интерфейса VTI, принимает значения [1..500].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
esr(config-ike-gw)# bind-interface vti 1
```

21.1.20 *show security ike*

Команда используется для просмотра списка шлюзов, политик или профилей.

Синтаксис

```
show security ike { gateway | policy | proposal } [<NAME>]
```

Параметры

gateway – при указании команды «gateway» будет выведен список сконфигурированных шлюзов;

policy – при указании команды «policy» будет выведен список сконфигурированных политик;

proposal – при указании команды «proposal» будет выведен список сконфигурированных профилей;

<NAME> – имя. При указании определенного имени шлюза, политики, профиля будет выведена подробная информация.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show security ike proposal
```

```

  Proposal
  ~~~~~~
Name      Auth      Encryption      DH      Hash      Lifetime
-----
aaa       pre-sha    3des           1       sha1       3600
          red-key
```

```
esr# show security ike policy
```

Policy ~~~~~		
Name	Mode	Proposal
-----	-----	-----
ike_poll	main	ike_prop1
<pre> esr# show security ike gateway ik_gw Description: -- IKE Policy: ike_poll IKE Version: v1-only Mode: route-based Binding interface: vti1 IKE Dead Peer Detection: Action: none Interval: 2 Timeout: 30 </pre>		

21.2 Управление VPN. Настройки IPsec

21.2.1 *security ipsec proposal*

Данной командой создается профиль для набора протоколов IPsec. Профиль IPsec включает в себя параметры алгоритмов шифрования и аутентификации, протокола защиты соединения IPsec-туннеля, а также время жизни соединения.

Использование отрицательной формы команды (no) удаляет указанный профиль.

Команда устанавливает режим командной строки SECURITY IPSEC PROPOSAL.

Синтаксис

```
[no] security ipsec proposal <NAME>
```

Параметры

<NAME> — имя профиля IPsec, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IPsec профили.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```

esr(config)# security ipsec proposal ipsec_prop1
esr(config-ipsec-proposal)#

```

21.2.2 *security ipsec policy*

Данной командой создается политика набора протоколов IPsec, которая включает в себя профили набора протоколов IPsec для согласования второй фазы протокола IKE.

Использование отрицательной формы команды (no) удаляет установленное значение.

Команда устанавливает режим командной строки SECURITY IPSEC POLICY.

Синтаксис

[no] security ipsec policy <NAME>

Параметры

<NAME> – имя политики IPsec, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IPsec-политики.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# security ipsec policy ipsec_poll
esr(config-ipsec-policy)#
```

21.2.3 security ipsec vpn

Данной командой создается VPN на основе набора протоколов IPsec и устанавливается командный режим SECURITY IPSEC VPN.

Использование отрицательной формы команды (no) удаляет сконфигурированный VPN.

Синтаксис

[no] security ipsec vpn <NAME>

Параметры

<NAME> – имя VPN, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все VPN.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
esr(config)# security ipsec vpn ipsec_vpn1
esr(config-ipsec-vpn)#
```

21.2.4 description

Данной командой выполняется изменение описания.

Использование отрицательной формы команды (no) удаляет описание.

Синтаксис

description <DESCRIPTION>

no description

Параметры

<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

CONFIG-IPSEC-PROPOSAL

CONFIG-IPSEC-POLICY

Пример

```
esr(config-ipsec-vpn)# description "VPN to Moscow Office"
```

21.2.5 *ike ipsec-policy*

Данная команда устанавливает привязку IPsec-политики к VPN. Данная команда актуальна, только если в VPN выбран режим согласования ключей «ike». Настройка режима согласования ключей описана в 21.2.8.

Синтаксис

ike ipsec-policy <NAME>

no ike ipsec-policy

Параметры

<NAME> – имя IPsec-политики, задаётся строка до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
esr(config-ipsec-vpn)# ike ipsec-policy ipsec_poll
```

21.2.6 *ike gateway*

Данной командой осуществляется привязка IKE шлюза к VPN. Данная команда актуальна, только если в VPN выбран режим согласования ключей «ike». Настройка режима согласования ключей описана в 21.2.8.

Синтаксис

ike gateway <NAME>

no ike gateway

Параметры

<NAME> – имя IKE шлюза, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
esr(config-ipsec-vpn)# ike gateway ike_gw1
```

21.2.7 *ike establish-tunnel*

Командой устанавливается режим активации VPN. Данная команда актуальна, только если в VPN выбран режим согласования ключей «ike». Настройка режима согласования ключей описана в 21.2.8).

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

ike establish-tunnel <MODE>

no ike establish-tunnel

Параметры

<MODE> – режим активации VPN:

by-request – соединение активируется встречной стороной;

route – соединение активируется при появлении трафика, маршрутизируемого в туннель;

immediate – туннель активируется автоматически после применения конфигурации.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
esr(config-ipsec-vpn)# ike establish-tunnel route
```

21.2.8 *ike dscp*

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов IKE-протокола.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

ike dscp <DSCP>

no ike dscp

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

63

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
esr(config-ipsec-vpn)# ike dscp 40
```

21.2.9 *mode*

Данной командой устанавливается режим согласования данных, необходимых для активации VPN.

Синтаксис

```
mode <MODE>
no mode
```

Параметры

<MODE> – режим работы VPN:

ike – согласование алгоритмов аутентификации и шифрования, ключей аутентификации и шифрования, индекса параметра безопасности и других данных осуществляется через протокол IKE;

manual – пользователь должен сам настроить идентичные параметры на обоих узлах для работы VPN. При данном режиме не происходит установления IKE-соединения между узлами. Каждый из узлов шифрует и дешифрует пакеты, основываясь только на заданных параметрах.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
esr(config-ipsec-vpn)# mode ike
```

21.2.10 *manual mode*

С помощью данной команды осуществляется установка режима перенаправления трафика в туннель. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в 21.2.8.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
manual mode <MODE>
no manual mode
```


Параметры

<MODE> – режим прохождения трафика:

policy-based – трафик перенаправляется на основе принадлежности к указанным в политиках подсетям;¹

route-based – трафик перенаправляется на основе маршрутов, шлюзом у которых является туннельный интерфейс.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
esr(config-ipsec-vpn)# manual mode route-based
```

21.2.11 *enable*

Данной командой активируется IPSEC VPN.

Использование отрицательной формы команды (no) деактивирует IPSEC VPN .

Синтаксис

[no] enable

Параметры

Команда не содержит аргументов.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
esr(config-ipsec-vpn)# enable
```

21.2.12 *manual bind-interface vti*

Данной командой указывается туннельный интерфейс, через который будет проходить трафик в режиме туннеля route-based. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в 21.2.8.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

¹ В текущей версии ПО не поддерживается

Синтаксис

```
manual bind-interface vti <VTI>  
no manual bind-interface vti
```

Параметры

<VTI> – индекс интерфейса VTI, принимает значения [1..500].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
esr(config-ipsec-vpn)# manual bind-interface vti 0
```

21.2.13 *manual protocol*

Данной командой устанавливается инкапсулирующий протокол. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в 21.2.8. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
manual protocol <TYPE>  
no manual protocol
```

Параметры

<TYPE> – тип протокола, принимает значения:

ah – данный протокол осуществляет только аутентификацию трафика, шифрование данных не выполняется;

esp – данный протокол осуществляет аутентификацию и шифрование трафика.

Значение по умолчанию

esp

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
esr(config-ipsec-vpn)# manual protocol ah
```

21.2.14 *manual spi*

Данной командой устанавливается индекс параметров безопасности. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в 21.2.8. Использование отрицательной формы команды (no) удаляет индекс параметров безопасности.

Синтаксис

```
manual spi <HEX>
no manual spi
```

Параметры

<HEX> – индекс параметров безопасности, задаётся значение размером 32 бита (8 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
esr(config-ipsec-vpn)# manual spi FF
```

21.2.15 *manual authentication algorithm*

Данной командой устанавливается алгоритм аутентификации. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в 21.2.8.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
manual authentication algorithm <ALGORITHM>
no manual authentication algorithm
```

Параметры

<ALGORITHM> – алгоритм аутентификации, принимает значения [md5, md5-128, sha1, sha1-160, aesxcbc, sha2-256, sha2-384, sha2-512].

Значение по умолчанию

none

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
esr(config-ipsec-vpn)# manual authentication algorithm sha1
```

21.2.16 *manual authentication key*

Данной командой устанавливается ключ аутентификации. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в 21.2.8.

Синтаксис

```
manual authentication key { ascii-text {<TEXT> | encrypted <ENCRYPTED-TEXT>} | hexadecimal  
{<HEX> | encrypted <ENCRYPTED-HEX> } }
```

```
no manual authentication key
```

Параметры

<TEXT> – строка [1..64] ASCII символов;

<HEX> – число размером [1..32] байт задаётся строкой [2..128] символов в шестнадцатеричном формате(0xYYYY...) или (YYYY...);

<ENCRYPTED_TEXT> – зашифрованный пароль размером [1..32] байт, задаётся строкой [2..128] символов;

<ENCRYPTED_HEX> – зашифрованное число размером [2..64] байт, задаётся строкой [2..256] символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
esr(config-ipsec-vpn)# manual authentication key hexadecimal abcdef
```

21.2.17 *manual encryption algorithm*

Данной командой устанавливается алгоритм шифрования. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в 21.2.8.

Использование отрицательной формы команды (no) удаляет установленное значение.

Синтаксис

```
manual encryption algorithm <ALGORITHM>
```

```
no manual encryption algorithm
```

Параметры

<ALGORITHM> – алгоритм шифрования, принимает значения: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.

Значение по умолчанию

3des

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
esr(config-ipsec-vpn)# manual encryption algorithm blowfish128
```

21.2.18 *manual encryption key*

Данной командой устанавливается ключ шифрования. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в 21.2.8.

Использование отрицательной формы команды (no) удаляет установленное значение.

Синтаксис

```
manual encryption key { ascii-text {< TEXT> | encrypted <ENCRYPTED-TEXT>} | hexadecimal {<HEX> | encrypted <ENCRYPTED-HEX> } }
```

```
no manual encryption key
```

Параметры

<TEXT> – строка [1..64] ASCII символов;

<HEX> – число размером [1..32] байт задаётся строкой [2..128] символов в шестнадцатеричном формате(0хYYYY...) или (YYYY...);

<ENCRYPTED-TEXT> – зашифрованный пароль размером [1..32] байт, задаётся строкой [2..128] символов;

<ENCRYPTED-HEX> – зашифрованное число размером [2..64] байт, задаётся строкой [2..256] символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
esr(config-ipsec-vpn)# manual encryption key hexadecimal 0x123456
```

21.2.19 *protocol*

Данной командой устанавливается инкапсулирующий протокол.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

protocol <PROTOCOL>

no protocol

Параметры

<PROTOCOL> – инкапсулирующий протокол, принимает значения:

ah – данный протокол осуществляет только аутентификацию трафика, шифрование данных не выполняется;

esp – данный протокол осуществляет аутентификацию и шифрование трафика.

Значение по умолчанию

esp

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-PROPOSAL

Пример

```
esr(config-ipsec-proposal)# protocol ah
```

21.2.20 authentication algorithm

Данной командой устанавливается алгоритм аутентификации. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

authentication algorithm <ALGORITHM>

no authentication algorithm

Параметры

<ALGORITHM> – алгоритм аутентификации, принимает значения: md5, sha1, sha2-256, sha2-384, sha2-512.

Значение по умолчанию

sha1

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-PROPOSAL

Пример

```
esr(config-ipsec-proposal)# authentication algorithm md5
```

21.2.21 *encryption algorithm*

Данной командой устанавливается алгоритм шифрования. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
encryption algorithm <ALGORITHM>
no encryption algorithm
```

Параметры

<ALGORITHM> – протокол шифрования, принимает значения: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.

Значение по умолчанию

3des

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-PROPOSAL

Пример

```
esr(config-ipsec-proposal)# encryption algorithm blowfish128
```

21.2.22 *lifetime*

Данной командой устанавливается время жизни IPsec -туннеля.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
lifetime seconds <SEC>
no lifetime seconds
```

Параметры

<SEC> – период времени жизни IPsec-туннеля, по истечении происходит пересогласование. Принимает значения [1140..86400] секунд.

Значение по умолчанию

28800 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-PROPOSAL

Пример

```
esr(config-ipsec-proposal)# lifetime seconds 3600
```

21.2.23 *proposal*

Данной командой к политике привязываются профили набора протоколов IPsec.

Использование отрицательной формы команды (no) удаляет привязку к указанному профилю.

Синтаксис

[no] proposal <NAME>

Параметры

<NAME> – имя профиля набора протоколов IPsec, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-POLICY

Пример

```
esr(config-ipsec-policy)# proposal ipsec_prop1
```

21.2.24 *show security ipsec*

Данной командой выполняется просмотр конфигураций VPN, политик и профилей набора протоколов IPsec.

Синтаксис

show security ipsec { vpn { configuration | status } | policy | proposal } [<NAME>]

Параметры

vpn configuration – при указании данной команды будет выведена конфигурация всех VPN;

vpn status – при указании данной команды будет выведено оперативное состояние всех VPN;

policy – при указании данной команды будет выведен список сконфигурированных политик набора протоколов IPsec;

proposal – при указании команды будет выведен список сконфигурированных профилей набора протоколов IPsec;

<NAME> – имя. При указании определенного имени VPN, политики или профиля будет выведена подробная информация.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show security ipsec proposal

Proposal
```



```

~~~~~
Name                Prot  Enc. alg.          Auth. alg.          Lifetime
-----
ipsec_prop1         esp   aes128             sha1                 28800 sec

esr# show security ipsec policy
Name                Description          Proposal
-----
ipsec_pol1          ipsec_prop1

esr# show security ipsec vpn configuration vpn1
Description:         --
State:               Enabled
IKE:
  Establish tunnel:   immediate
  IPsec policy:       ipsec_pol1
  IKE gateway:        ike_gw1

```

21.2.25 *show security ipsec vpn status*

Данной командой выполняется просмотр статуса всех VPN, которые устанавливают соединение через IKE-протокол либо определенного VPN при указании его имени.

Синтаксис

show security ipsec vpn status [<NAME>]

Параметры

<NAME> – имя VPN, задаётся строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```

esr# show security ipsec vpn status
Name                Local host      Remote host      Initiator spi      Responder spi      State
-----
ipsec_vpn1          10.100.14.1     10.100.14.2     0x05d8e0ac3543f0cb 0xcfa1c4179d001154 Established

```

22 УПРАВЛЕНИЕ VPN. НАСТРОЙКИ УДАЛЕННОГО ДОСТУПА

22.1 Общие команды настройки удаленного доступа

22.1.1 *description*

Команда используется для изменения описания профиля OpenVPN, PPTP и L2TP over IPsec серверов. Использование отрицательной формы команды (no) удаляет описание профиля.

Синтаксис

description <DESCRIPTION>

no description

Параметры

<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPTP

CONFIG-L2TP

CONFIG-OPENVPN

Пример

Установить описание для профиля PPTP-сервера:

```
esr(config-pptp)# description "Our remote workers"
```

22.1.1 *security-zone*

Данной командой указывается зона безопасности, с которой будут соотноситься интерфейсы, создающиеся при подключении удаленных пользователей по протоколам OpenVPN, PPTP и L2TP over IPsec.

Использование отрицательной формы команды (no) удаляет привязку к зоне безопасности.

Синтаксис

security-zone <NAME>

no security-zone

Параметры

<NAME> – имя зоны безопасности, задаётся строкой до 12 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPTP

CONFIG-L2TP

CONFIG-OPENVPN

Пример

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

22.1.2 *show remote-access configuration*

Командой выполняется просмотр параметров профилей OpenVPN, PPTP и L2TP over IPsec серверов.

Синтаксис

```
show remote-access configuration { pptp | l2tp | openvpn } [ <NAME> ]
```

Параметры

<NAME> – имя профиля OpenVPN, PPTP или L2TP over IPsec сервера.

При выполнении команды без параметра будут показаны параметры всех OpenVPN, PPTP или L2TP over IPsec-серверов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
esr# show remote-access configuration pptp pptp1
State:                               Enabled
Description:                         --
Security zone:                       trusted
Authentication mode:                 local
MTU:                                 1500
Local address:                       192.168.1.1
Remote address:                      rem_pptp(10.0.10.20-10.0.10.40)
Outside address:                     115.0.0.1
DNS server:                          --
WINS server:                         --
```

Users

~~~~~

| # | Name | State   | Encrypted password |
|---|------|---------|--------------------|
| 0 | pptp | Enabled | 8CB5107EA7005AFF   |
| 1 | petr | Enabled | CCE5513EE45A1EAC   |

## 22.1.3 *show remote-access counters*

Командой выполняется просмотр счетчиков соединений OpenVPN, PPTP и L2TP over IPsec пользователей.

### Синтаксис

```
show remote-access counters [ pptp | l2tp | openvpn ] [ server <SERVER-NAME> ] [ username <USER-NAME> ] [ ip-address <ADDR> ]
```

## Параметры

<SERVER-NAME> – имя профиля PPTP или L2TP over IPsec сервера;

<USER-NAME> – имя OpenVPN, PPTP или L2TP over IPsec пользователя;

<ADDR> – IP-адрес OpenVPN, PPTP или L2TP over IPsec пользователя.

При выполнении команды без параметра будут показаны счетчики всех соединений OpenVPN, PPTP и L2TP over IPsec пользователей.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show remote-access counters
User          IP-address    UC recv      Bytes recv   Err recv     MC recv
-----
ivan          10.20.20.5    262          25365        0            0
fedor         20.20.20.160  59           5236         0            0

User          IP-address    UC sent      Bytes sent   Err sent
-----
ivan          10.20.20.5    249          29298        0
fedor         20.20.20.160  16           739          0

esr# show remote-access counters l2tp
PPTP Server: remote-workers
User: ivan(10.20.20.5)
Packets received:      231
Bytes received:        22229
Dropped on receive:    0
Receive errors:        0
Multicasts received:   0
Receive length errors: 0
Receive buffer overflow errors: 0
Receive CRC errors:    0
Receive frame errors:   0
Receive FIFO errors:   0
Receive missed errors: 0
Receive compressed:    0
Packets transmitted:   189
Bytes transmitted:     21858
Dropped on transmit:    0
Transmit errors:       0
Transmit aborted errors: 0
Transmit carrier errors: 0
Transmit FIFO errors:  0
Transmit heartbeat errors: 0
Transmit window errors: 0
Transmit compressed:   0
Collisions:            0
```

### 22.1.4 *show remote-access status*

Командой выполняется просмотр состояния соединений OpenVPN, PPTP и L2TP over IPsec пользователей.

## Синтаксис

```
show remote-access status [ pptp | l2tp | openvpn ] [ server <SERVER-NAME> ] [ username <USER-NAME> ] [ ip-address <ADDR> ]
```

## Параметры

<SERVER-NAME> – имя профиля OpenVPN, PPTP или L2TP over IPsec сервера;

<USER-NAME> – имя OpenVPN, PPTP или L2TP over IPsec пользователя;

<ADDR> – IP-адрес OpenVPN, PPTP или L2TP over IPsec пользователя.

При выполнении команды без параметра будет показано состояние всех соединений OpenVPN, PPTP и L2TP over IPsec пользователей.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show remote-access status
User          IP-address      Server
-----
ivan          10.20.20.5      pptp(remote-workers)
fedor         20.20.20.160    l2tp(remote-workers-l2tp)
Count sessions: 2
```

## 22.1.5 clear remote-access session

Данной командой осуществляется завершение соединений OpenVPN, PPTP и L2TP over IPsec пользователей.

## Синтаксис

```
clear remote-access session [ pptp | l2tp | openvpn ] [ server <SERVER-NAME> ] [ username <USER-NAME> ] [ ip-address <ADDR> ]
```

## Параметры

<SERVER-NAME> – имя профиля OpenVPN, PPTP или L2TP over IPsec сервера;

<USER-NAME> – имя OpenVPN, PPTP или L2TP over IPsec пользователя;

<ADDR> – IP-адрес OpenVPN, PPTP или L2TP over IPsec пользователя. При выполнении команды без параметра будут завершены все OpenVPN, PPTP и L2TP over IPsec соединения.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример

```
esr# clear remote-access session
```

---

## 22.1.6 *clear remote-access counters*

---

Данной командой осуществляется сброс счетчиков соединений OpenVPN, PPTP и L2TP over IPsec пользователей.

### Синтаксис

```
clear remote-access counters [ pptp | l2tp | openvpn ] [ server <SERVER-NAME> ] [ username <USER-NAME> ] [ ip-address <ADDR> ]
```

### Параметры

<SERVER-NAME> – имя профиля OpenVPN, PPTP или L2TP over IPsec сервера;

<USER-NAME> – имя OpenVPN, PPTP или L2TP over IPsec пользователя;

<ADDR> – IP-адрес OpenVPN, PPTP или L2TP over IPsec пользователя.

При выполнении команды без параметра будут сброшены все счетчики соединений OpenVPN, PPTP и L2TP over IPsec пользователей.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# clear remote-access counters
```

---

## 22.2 *Настройка L2TP over IPsec/PPTP*

---

### 22.2.1 *remote-access pptp*

---

Данной командой создается профиль PPTP-сервера.

Использование отрицательной формы команды (no) удаляет указанный профиль.

Команда устанавливает режим командной строки PPTP SERVER.

### Синтаксис

```
[no] remote-access pptp <NAME>
```

### Параметры

<NAME> – имя профиля PPTP-сервера, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# remote-access pptp remote-workers  
esr(config-pptp)#
```

---

## 22.2.2 *remote-access l2tp*

---

Данной командой создается профиль L2TP over IPsec сервера.

Использование отрицательной формы команды (no) удаляет указанный профиль.

Команда устанавливает режим командной строки L2TP SERVER.

### Синтаксис

```
[no] remote-access l2tp <NAME>
```

### Параметры

<NAME> – имя профиля L2TP over IPsec-сервера, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# remote-access l2tp remote-workers  
esr(config-l2tp)#
```

---

## 22.2.3 *authentication mode*

---

Данной командой устанавливается режим аутентификации удаленных пользователей, подключающихся по протоколам PPTP или L2TP over IPsec.

Использование отрицательной формы команды (no) удаляет установленный режим.

### Синтаксис

```
authentication mode { local | radius }
```

```
no authentication mode
```

### Параметры

local – режим аутентификации, использующий локальную базу пользователей конфигурируемого профиля.

radius – режим, при котором аутентификация пользователей проходит через RADIUS-сервер.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-PPTP

CONFIG-L2TP

### Пример

```
esr(config-pptp)# authentication mode local
```

## 22.2.4 *dns-servers*

---

Данной командой указывается список DNS-серверов, которые будут использовать удаленные пользователи, подключающиеся по протоколам PPTP и L2TP over IPsec.

Использование отрицательной формы команды (no) удаляет настроенные адреса DNS-серверов.

### Синтаксис

dns-servers object-group <NAME>

no dns-servers

### Параметры

<NAME> – имя профиля IP-адресов, который содержит адреса необходимых DNS-серверов, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-PPTP

CONFIG-L2TP

### Пример

```
esr(config-pptp)# dns-servers object-group pptp_dns
```

## 22.2.5 *dscp*

---

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов PPTP и L2TP over IPsec серверов.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

### Синтаксис

dscp <DSCP>

no dscp

### Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

### Значение по умолчанию

32

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-PPTP

CONFIG-L2TP

### Пример

```
esr(config-pptp)# dscp 40
```



## 22.2.6 *enable*

Данной командой активируется конфигурируемый профиль PPTP и L2TP over IPsec-серверов. Использование отрицательной формы команды (no) деактивирует конфигурируемый профиль.

### Синтаксис

[no] enable

### Параметры

Команда не содержит аргументов.

### Значение по умолчанию

Выключено

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-PPTP

CONFIG-L2TP

### Пример

```
esr(config-pptp)# enable
```

## 22.2.7 *ipsec authentication method*

Данной командой выбирается метод аутентификации по ключу для IKE-соединения. Аутентификация сообщений по ключу используется при установлении IKE-соединения, ключ задаётся командой «ipsec authentication pre-shared-key» (см п. 22.2.8).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

ipsec authentication method psk

no ipsec authentication method

### Параметры

psk – метод аутентификации, использующий предварительно полученные ключи шифрования.

### Необходимый уровень привилегий

10

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-L2TP

### Пример

```
esr(config-l2tp)# ipsec authentication method psk
```

## 22.2.8 *ipsec authentication pre-shared-key*

Данной командой устанавливается общий секретный ключ для аутентификации, который должен совпадать у обеих сторон, устанавливающих туннель.

Использование отрицательной формы команды (no) удаляет установленный ключ.

### Синтаксис

```
ipsec authentication pre-shared-key { ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> } |  
hexadecimal {<HEX> | encrypted <ENCRYPTED-HEX> } }  
no ipsec authentication pre-shared-key
```

### Параметры

<TEXT> – строка [1..64] ASCII символов;

<HEX> – число размером [1..32] байт задаётся строкой [2..128] символов в шестнадцатеричном формате(0xYYYY...) или (YYYY...).

<ENCRYPTED-TEXT> – зашифрованный пароль размером [1..32] байт, задаётся строкой [2..128] символов;

<ENCRYPTED-HEX> – зашифрованное число размером [2..64] байт, задаётся строкой [2..256] символов.

### Значение по умолчанию

none

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-L2TP

### Пример

```
esr(config-l2tp)# ipsec authentication pre-shared-key ascii-text password
```

## 22.2.9 *local-address*

Данной командой указывается IP-адрес, используемый PPTP или L2TP over IPsec сервером в качестве локального IP-адреса туннеля.

Использование отрицательной формы команды (no) удаляет настроенный локальный IP-адрес туннеля.

### Синтаксис

```
local-address { object-group <NAME> | ip-address <ADDR> }  
no local-address
```

### Параметры

<NAME> – имя профиля IP-адресов, который содержит локальный IP-адрес туннеля, задаётся строкой до 31 символа;

<ADDR> – локальный IP-адрес туннеля задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-PPTP

CONFIG-L2TP

## Пример

```
esr(config-pptp)# local-address object-group pptp_local
```

### 22.2.10 *mtu*

Данной командой указывается MTU для интерфейсов, которые будут создаваться при подключении удаленных пользователей по протоколам PPTP и L2TP over IPsec.

Использование отрицательной формы команды (no) устанавливает значение MTU по умолчанию.

## Синтаксис

mtu <MTU>

no mtu

## Параметры

<MTU> – значение MTU, принимает значения в диапазоне [1280..1500].

## Значение по умолчанию

1500

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-PPTP

CONFIG-L2TP

## Пример

```
esr(config-pptp)# mtu 1400
```

### 22.2.11 *outside-address*

Данной командой указывается IP-адрес, который будет прослушиваться PPTP или L2TP over IPsec сервером на наличие входящих подключений.

Использование отрицательной формы команды (no) удаляет настроенный адрес для прослушивания.

## Синтаксис

outside-address { object-group <NAME> | ip-address <ADDR> }

no outside-address

## Параметры

<NAME> – имя профиля IP-адресов, содержащий адрес, который будет прослушиваться PPTP или L2TP over IPsec сервером на наличие входящих подключений, задаётся строкой до 31 символа;

<ADDR> – IP-адрес, который будет прослушиваться PPTP или L2TP over IPsec сервером на наличие входящих подключений, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-PPTP

CONFIG-L2TP

## Пример

```
esr(config-pptp)# outside-address object-group pptp_outside
```

### 22.2.12 remote-address

Данной командой указывается список IP-адресов, из которого PPTP или L2TP over IPsec сервером выдаются динамические IP-адреса удаленным пользователям.

Использование отрицательной формы команды (no) удаляет список IP-адресов удаленных пользователей.

## Синтаксис

remote-address { object-group <NAME> | address-range <FROM-ADDR>-<TO-ADDR> }

no remote-address

## Параметры

<NAME> – имя профиля IP-адресов, который содержит список IP-адресов удаленных пользователей, задаётся строкой до 31 символа;

<FROM-ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-PPTP

CONFIG-L2TP

## Пример

```
esr(config-pptp)# remote-address object-group pptp_remote
```

## 22.2.13 *username*

Данной командой создается пользователь для подключения к PPTP или L2TP over IPsec серверам.

Использование отрицательной формы команды (no) удаляет указанного пользователя.

Команда устанавливает режим командной строки PPTP USER или L2TP USER в зависимости от текущего командного режима.

### Синтаксис

[no] username <NAME>

### Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа.

### Командный режим

CONFIG-PPTP

CONFIG-L2TP

### Пример

```
esr(config-pptp)# username fedor
esr(config-pptp-user)#
```

## 22.2.14 *password*

Команда для установки пароля в открытой или зашифрованной форме определённому пользователю для подключения к PPTP или L2TP over IPsec-серверам. Пароль пользователя хранится в конфигурации в зашифрованной форме. При конфигурировании можно задать пароль в открытой форме либо скопировать пароль в зашифрованной форме с другого устройства.

Использование отрицательной формы команды (no) удаляет пароль пользователя.

### Синтаксис

password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }

no password

### Параметры

<CLEAR-TEXT> – пароль в открытой форме, задаётся строкой [8 .. 64] символов, может включать символы [0-9a-fA-F].

<ENCRYPTED-TEXT> – пароль в зашифрованной форме, задаётся строкой [16..128] символов.

**Примечание.** Пароли хранятся в конфигурации в зашифрованной форме независимо от формата использованного при вводе команды.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-PPTP-USER

CONFIG-L2TP-USER

## Пример

```
esr(config-pptp-user) password ascii-text 01234567
```

### 22.2.1 *wins-servers*

Данной командой указывается список WINS-серверов, которые будут использовать удаленные пользователи, подключающиеся по протоколам PPTP и L2TP over IPsec.

Использование отрицательной формы команды (no) удаляет настроенные адреса WINS-серверов.

#### Синтаксис

```
wins-servers object-group <NAME>
```

```
no wins-servers
```

#### Параметры

<NAME> – имя профиля IP-адресов, который содержит адреса необходимых WINS-серверов, задаётся строкой до 31 символа.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-PPTP

CONFIG-L2TP

## Пример

```
esr(config-pptp) # wins-servers object-group l2tp_wins
```

### 22.2.1 *dns-servers*

Данной командой указывается список DNS-серверов, которые будут использовать удаленные пользователи, подключающиеся по протоколам PPTP и L2TP over IPsec.

Использование отрицательной формы команды (no) удаляет настроенные адреса DNS-серверов.

#### Синтаксис

```
dns-servers object-group <NAME>
```

```
no dns-servers
```

#### Параметры

<NAME> – имя профиля IP-адресов, который содержит адреса необходимых DNS-серверов, задаётся строкой до 31 символа.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-PPTP

CONFIG-L2TP

## Пример

```
esr(config-pptp)# dns-servers object-group pptp_dns
```

## 22.3 Настройка OpenVPN

### 22.3.1 *remote-access openvpn*

Данной командой создается профиль OpenVPN-сервера.

Использование отрицательной формы команды (no) удаляет указанный профиль.

Команда устанавливает режим командной строки CONFIG-OPENVPN.

#### Синтаксис

```
[no] remote-access openvpn <NAME>
```

#### Параметры

<NAME> – имя профиля OpenVPN-сервера, задаётся строкой до 31 символа.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

```
esr(config)# remote-access openvpn remote-workers
esr(config-openvpn)#
```

### 22.3.1 *enable*

Данной командой активируется конфигурируемый профиль OpenVPN сервера.

Использование отрицательной формы команды (no) деактивирует конфигурируемый профиль.

#### Синтаксис

```
[no] enable
```

#### Параметры

Команда не содержит аргументов.

#### Значение по умолчанию

Выключено

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-OPENVPN

#### Пример

```
esr(config-openvpn)# enable
```

---

### **22.3.1**      *address-range*

---

Данной командой указывается список IP-адресов, из которого OpenVPN сервером выдаются динамические IP-адреса удаленным пользователям в режиме L2.

Использование отрицательной формы команды (no) удаляет список IP-адресов удаленных пользователей.

#### **Синтаксис**

address-range <FROM-ADDR>-<TO-ADDR>

no address-range

#### **Параметры**

<FROM-ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-OPENVPN

#### **Пример**

```
esr(config-openvpn)# address-range 192.168.1.10-192.168.1.250
```

---

### **22.3.1**      *encryption algorithm*

---

Данной командой выбирается алгоритм шифрования, используемый при передачи данных.

Использование отрицательной формы команды (no) отключает шифрование.

#### **Синтаксис**

encryption algorithm <ALGORITHM>

no encryption algorithm

#### **Параметры**

<ALGORITHM> – идентификатор протокола шифрования, принимает значения: 3des,blowfish128, aes128.

#### **Значение по умолчанию**

Шифрование отключено.

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-OPENVPN

#### **Пример**

```
esr(config-openvpn)# encryption algorithm aes128
```



### 22.3.2 *network*

Данной командой определяется подсеть, из которой выдаются IP-адреса пользователям. Первый IP-адрес в подсети выступает шлюзом для пользовательских сессий.

Использование отрицательной формы команды (no) удаляет данную подсеть.

#### Синтаксис

```
network <ADDR/LEN>
```

```
no network
```

#### Параметры

<ADDR/LEN> – адрес подсети, имеет один из следующих форматов:

AAA.BBB.CCC.DDD/EE – IP-адрес подсети с маской в форме префикса, где AAA-DDD принимают значения [0..255] и EE принимает значения [1..32];

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-OPENVPN

#### Пример

```
esr(config-openvpn)# network 192.168.25.0/24
```

### 22.3.3 *route*

Данной командой включается анонсирование указанных подсетей, шлюзом является IP адрес OpenVPN-сервера (первый IP-адрес из подсети, заданной с помощью команды network, описанной в разделе 22.3.2).

Использование отрицательной формы команды (no) отключает анонсирование указанных подсетей.

#### Синтаксис

```
route <ADDR/LEN>
```

```
no route
```

#### Параметры

<ADDR/LEN> – адрес подсети, имеет один из следующих форматов:

AAA.BBB.CCC.DDD/EE – IP-адрес подсети с маской в форме префикса, где AAA-DDD принимают значения [0..255] и EE принимает значения [1..32];

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-OPENVPN

#### Пример

```
esr(config-openvpn)# route 192.168.25.0/24, 192.168.26.0/24
```

---

### **22.3.4**      *port*

---

Данной командой устанавливается TCP/UDP порт, который будет прослушиваться OpenVPN сервером.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### **Синтаксис**

port <PORT>

no port

#### **Параметры**

<PORT> – TCP/UDP порт, принимает значения [1..65535].

#### **Значение по умолчанию**

1194

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-OPENVPN

#### **Пример**

```
esr(config-openvpn)# port 5000
```

---

### **22.3.1**      *compression*

---

Данной командой включается механизм сжатия передаваемых данных между клиентами и сервером OpenVPN.

Использование отрицательной формы команды (no) отключает механизм сжатия передаваемых данных.

#### **Синтаксис**

[no] compression

#### **Параметры**

Команда не содержит аргументов.

#### **Значение по умолчанию**

Выключено

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-OPENVPN

#### **Пример**

```
esr(config-openvpn)# compression
```

## 22.3.2 *client-isolation*

Данной командой включается блокировка передачи данных между клиентами.

Использование отрицательной формы команды (no) снимает блокировку.

### Синтаксис

[no] client-isolation

### Параметры

Команда не содержит аргументов.

### Значение по умолчанию

Выключено

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OPENVPN

### Пример

```
esr(config-openvpn)# client-isolation
```

## 22.3.3 *username*

Данной командой определяется подсеть для указанного пользователя OpenVPN сервера.

Использование отрицательной формы команды (no) удаляет принадлежность к определенной подсети.

### Синтаксис

username <NAME> subnet <ADDR/LEN>

no username <NAME>

### Параметры

<ADDR/LEN> – адрес подсети, имеет один из следующих форматов:

AAA.BBB.CCC.DDD/EE – IP-адрес подсети с маской в форме префикса, где AAA-DDD принимают значения [0..255] и EE принимает значения [1..32];

<NAME> – имя пользователя, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OPENVPN

### Пример

```
esr(config-bgp)# username bookkeeper subnet 192.168.25.128/28
```

---

### 22.3.4 *client-max*

---

Данной командой устанавливается максимальное количество одновременных пользовательских сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

client-max <VALUE>

no client-max

#### Параметры

<VALUE> – максимальное количество пользователей, принимает значения [1..65535].

#### Значение по умолчанию

Не ограничено.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-OPENVPN

#### Пример

```
esr(config-openvpn)# clients-max 500
```

---

### 22.3.1 *certificate*

---

Данной командой указываются сертификаты и ключи. Сертификаты и ключи должны быть предварительно скопированы на маршрутизатор с помощью команды **copy**, описанной в разделе 5.14.

Использование отрицательной формы команды (no) удаляет из профиля указанный сертификат.

#### Синтаксис

certificate <CERTIFICATE-TYPE> <NAME>

no certificate <CERTIFICATE-TYPE>

#### Параметры

<CERTIFICATE-TYPE> - тип сертификата или ключа, может принимать следующие значения:

- ca – сертификат удостоверяющего сервера;
- crl – список отозванных сертификатов;
- dh – ключ Диффи-Хелмана;
- server-crt – публичный сертификат сервера;
- server-key – приватный ключ сервера;
- ta – HMAC ключ.

<NAME> – имя сертификата или ключа, задаётся строкой до 31 символа.

#### Необходимый уровень привилегий

10

## Командный режим

CONFIG-OPENVPN

### Пример

```
esr(config-openvpn)# certificate ca ca.crt
```

## 22.3.2 *protocol*

Данной командой устанавливается инкапсулирующий протокол.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

protocol <PROTOCOL>

no protocol

### Параметры

<TYPE> – тип инкапсуляции, возможные значения:

IP-инкапсуляция в IP-пакет;

UDP-инкапсуляция в UDP-дейтаграммы.

### Значение по умолчанию

Не установлено.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-OPENVPN

### Пример

```
esr(config-openvpn)# protocol udp
```

## 22.3.3 *tunnel*

Данной командой определяется тип соединения с частной сетью через OpenVPN сервер.

Использование отрицательной формы команды (no) удаляет текущее значение.

### Синтаксис

tunnel <TYPE>

no tunnel

### Параметры

<TYPE> – инкапсулирующий протокол, принимает значения:

ip – соединение точка-точка;

ethernet – подключение к L2 домену.

---

**Значение по умолчанию**

Отсутствует.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OPENVPN

**Пример**

```
esr(config-openvpn)# tunnel ip
```

---

**22.3.1      *timers keepalive***

---

Данной командой устанавливается временной интервал, по истечении которого идет проверка соединения со встречной стороной.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

timers keepalive <TIME>

no timers keepalive

**Параметры**

<TIME> – время в секундах, принимает значения [1..65535].

**Значение по умолчанию**

10 секунд

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OPENVPN

**Пример**

```
esr(config-openvpn)# timers keepalive 120
```

---

**22.3.1      *timers holdtime***

---

Данной командой устанавливается временной интервал, по истечении которого встречная сторона считается недоступной. Таймер запускается после установления отношений соседства и начинает отсчет от 0. Таймер сбрасывается при получении каждого ответа на keepalive сообщение от встречной стороны. Рекомендуется устанавливать значение таймера равное  $3 * keepalive$ .

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

timers holdtime <TIME>

no timers holdtime

## Параметры

<TIME> – время в секундах, принимает значения [1..65535].

## Значение по умолчанию

120 секунд

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OPENVPN

## Пример

```
esr(config-openvpn)# timers holdtime 360
```

## 22.3.2 *redirect-gateway*

Данная команда включает анонсирование маршрута по умолчанию для OpenVPN соединений, что приводит к замене маршрута по умолчанию на клиентской стороне. Новым шлюзом по умолчанию станет IP-адрес OpenVPN сервера.

Использование отрицательной формы команды (no) отключает анонсирование маршрута по умолчанию.

## Синтаксис

[no] redirect-gateway

## Параметры

Команда не содержит аргументов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OPENVPN

## Пример

```
esr(config-openvpn)# redirect-gateway
```

## 22.3.3 *bridge-group*

Данная команда используется для включения клиентских соединений по OpenVPN в L2 домен.

Использование отрицательной формы команды (no) исключает соединения из L2 домена.

## Синтаксис

bridge-group <BRIDGE-ID>

no bridge-group

## Параметры

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе 303.3.

---

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OPENVPN

## Пример

```
esr(config-openvpn)# bridge-group 15
```

---

### 22.3.1 wins-servers

Данной командой указывается список WINS-серверов, которые будут использовать удаленные пользователи.

Использование отрицательной формы команды (no) удаляет настроенные адреса WINS-серверов.

## Синтаксис

wins-servers <ADDR>

no wins-servers { <ADDR> | all }

## Параметры

<ADDR> – IP-адрес WINS сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

all – удалить все сконфигурированные IP-адреса DNS серверов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OPENVPN

## Пример

```
esr(config-openvpn)# wins-servers 1.1.1.1
```

---

### 22.3.2 dns-servers

Данной командой указывается список DNS-серверов, которые будут использовать удаленные пользователи.

Использование отрицательной формы команды (no) удаляет настроенные адреса DNS-серверов.

## Синтаксис

dns-servers <ADDR>

no dns-servers { <ADDR> | all }

## Параметры

< ADDR> – IP-адрес DNS сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

all – удалить все сконфигурированные диапазоны IP-адресов.



---

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OPENVPN

**Пример**

```
esr(config-openvpn)# dns-server 1.1.1.1
```

## 23 МАРШРУТИЗАЦИЯ

### 23.1 Общие настройки маршрутизации

---

#### 23.1.1 *ip protocols preference*

---

Данная команда позволяет настроить приоритетность протоколов маршрутизации для основной таблицы маршрутизации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

##### Синтаксис

ip protocols <PROTOCOL> preference <VALUE>

no ip protocols <PROTOCOL> preference

##### Параметры

<PROTOCOL> – вид протокола, принимает значения: static (только в глобальном режиме), rip (только в глобальном режиме), ospf, bgp;

<VALUE> – приоритетность протокола, принимает значения в диапазоне [1..255].

##### Значение по умолчанию

BGP (170)

OSPF (150)

RIP (100)

Static (1)

##### Необходимый уровень привилегий

10

##### Командный режим

CONFIG

CONFIG-VRF

##### Пример

```
esr(config)# ip protocols ospf preference 44
```

#### 23.1.1 *ip protocols max-routes*

---

Данная команда позволяет настроить емкость таблиц маршрутизации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

##### Синтаксис

ip protocols <PROTOCOL> max-routes <VALUE>

no ip protocols <PROTOCOL> max-routes

## Параметры

<PROTOCOL> – вид протокола, принимает значения: rip (только в глобальном режиме), ospf, bgp;

<VALUE> – количество маршрутов в маршрутной таблице, принимает значения в диапазоне:

- OSPF [1..500000];
- RIP [1..10000];
- BGP [1..2600000].

## Значение по умолчанию для глобального режима

BGP (2600000)

OSPF (500000)

RIP (10000)

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

CONFIG-VRF

## Пример

```
esr(config)# ip protocols ospf max-routes 4400
```

### 23.1.2 *ipv6 protocols max-routes*

Данная команда позволяет настроить емкость таблиц маршрутизации IPv6.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

ip protocols <PROTOCOL> max-routes <VALUE>

no ip protocols <PROTOCOL> max-routes

## Параметры

<PROTOCOL> – вид протокола, принимает значения: rip (только в глобальном режиме), ospf, bgp;

<VALUE> – количество маршрутов в маршрутной таблице, принимает значения в диапазоне:

- OSPFv3 [1..500000];
- IPv6 BGP [1..2600000].

## Значение по умолчанию для глобального режима

IPv6 BGP (2600000)

OSPFv3 (500000)

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

CONFIG-VRF

### Пример

```
esr(config)# ipv6 protocols ospf max-routes 4400
```

### 23.1.3 *router log-adjacency-changes*

Данная команда позволяет включить вывод информации о состоянии отношений с соседями для протоколов маршрутизации.

Использование отрицательной формы команды (no) отключает вывод информации.

#### Синтаксис

[no] router <TYPE> log-adjacency-changes

#### Параметры

<TYPE> – протокол:

ospf – протокол OSPFv2;

bgp – протокол BGP.

#### Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример

```
esr(config)# router bgp log-adjacency-changes
```

### 23.1.4 *ipv6 router log-adjacency-changes*

Данная команда позволяет включить вывод информации о состоянии отношений с соседями для протоколов маршрутизации.

Использование отрицательной формы команды (no) отключает вывод информации.

#### Синтаксис

[no] ipv6 router <TYPE> log-adjacency-changes

#### Параметры

<TYPE> – протокол:

ospf – протокол OSPFv3;

bgp – протокол BGP.

#### Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример

```
esr(config)# ipv6 router ospf log-adjacency-changes
```

## 23.1.5 *ip path-mtu-discovery*

Данная команда разрешает поиск PMTU для протоколов TCP, SCTP, DCCP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

ip path-mtu-discovery <ACT>

no ip path-mtu-discovery

### Параметры

<ACT> – назначаемое действие:

enable – разрешает поиск PMTU для протоколов TCP, SCTP, DCCP;

disable – запрещает поиск PMTU для протоколов TCP, SCTP, DCCP.

### Значение по умолчанию

Поиск PMTU разрешен

### Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример

```
esr(config)# ip path-mtu-discovery disable
```

## 23.1.6 *ip tcp adjust-mss*

Данной командой переопределяется значение поля MSS (Maximum segment size) во входящих TCP пакетах.

Использование отрицательной формы команды (no) отключает корректировку значения поля MSS.

### Синтаксис

ip tcp adjust-mss <MSS>

no ip tcp adjust-mss

### Параметры

<MSS> – значение MSS, принимает значения в диапазоне [500..1460].

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE  
CONFIG-L2TPV3  
CONFIG-VTI

### Пример

```
esr(config-if-gi) # ip tcp adjust-mss 1400
```

### 23.1.7 *ipv6 tcp adjust-mss*

Данной командой переопределяется значение поля MSS (Maximum segment size) во входящих TCP пакетах.

Использование отрицательной формы команды (no) отключает корректировку значения поля MSS.

#### Синтаксис

ipv6 tcp adjust-mss <MSS>  
no ipv6 tcp adjust-mss

#### Параметры

<MSS> – значение MSS, принимает значения в диапазоне [40..1940].

#### Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE

### Пример

```
esr(config-if-gi) # ipv6 tcp adjust-mss 1400
```

### 23.1.8 *show ip route*

Данная команда позволяет просмотреть таблицу маршрутизации устройства. Если задан параметр <SUBNET>, то детально отображаются маршруты к данной подсети. Если задан параметр <VRF>, то команда отображает таблицу маршрутизации указанного экземпляра VRF.

#### Синтаксис

show ipv6 route [ vrf <VRF> ] [ <SUBNET> | all | summary ]

## Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<SUBNET> – адрес назначения, опциональный параметр, может быть задан в следующих видах:

AAA.BBB.CCC.DDD – IP-адрес хоста, где каждая часть принимает значения [0..255];

AAA.BBB.CCC.DDD/NN – IP-адрес подсети с маской в виде префикса, где AAA-DDD принимают значения [0..255] и NN принимает значения [1..32];

all – выводит информацию о всех маршрутах, включая неактивные;

summary – выводит суммарную статистику протоколов маршрутизации.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show ip route
Codes: C - connected, S - static, R - RIP derived,
       O - OSPF derived, IA - OSPF inter area route,
       E1 - OSPF external type 1 route, E2 - OSPF external type 2 route
       B - BGP derived, D - DHCP derived, K - kernel route,
       * - FIB route

C      * 192.168.1.0/24      [0/0]    dev br1                [direct 01:14:16]
C      * 10.100.100.0/24    [0/0]    dev gi1/0/5           [direct 01:14:17]

esr# show ip route summary
Direct Connected: 12
Static:           46
RIP:              0
OSPF:             2000
BGP:              100000
```

### 23.1.9 *show ipv6 route*

Команда для просмотра таблицы маршрутизации устройства. Если задан параметр <SUBNET>, то детально отображаются маршруты к данной подсети. Если задан параметр <VRF>, то команда отображает таблицу маршрутизации указанного экземпляра VRF.

## Синтаксис

show ipv6 route [ vrf <VRF> ] [ <SUBNET> | all | summary ]

## Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<SUBNET> – адрес назначения, опциональный параметр, может быть задан в следующих видах:

X:X:X:X:X – IPv6-адрес хоста, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];

X:X:X:X/EE – IPv6-адрес подсети с маской в виде префикса, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128];

all – выводит информацию о всех маршрутах, включая неактивные;

summary – выводит суммарную статистику протоколов маршрутизации.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show ipv6 route
Codes: C - connected, S - static, R - RIP derived,
       O - OSPF derived, IA - OSPF inter area route,
       E1 - OSPF external type 1 route, E2 - OSPF external type 2 route
       B - BGP derived, D - DHCP derived, K - kernel route,
       * - FIB route

S      * ::/0                [1/0]   via fc00::1 on gil/0/5          [static
03:16:23]
S      * 2001::/120          [1/6]   dev gil/0/5                    [static
03:16:23]
C      * fc00::/120          [0/0]   dev gil/0/5                    [direct
03:16:23]
S      * fc00:3::1/128       [1/0]   via fc00::1 on gil/0/5          [static
03:16:23]

esr# show ipv6 route summary
Direct Connected: 1
Static:           3
RIP:              0
OSPF:             0
BGP:              0
```

## 23.2 Общие команды анонсирования и приема маршрутов

### 23.2.1 *ip prefix-list*

Данной командой создается список IP-подсетей, который в дальнейшем будет использоваться для фильтрации анонсируемых и получаемых IP-маршрутов.

Использование отрицательной формы команды (no) удаляет список префиксов.

#### Синтаксис

[no] ip prefix-list <NAME>

#### Параметры

<NAME> – имя конфигурируемого списка подсетей, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10



## Командный режим

CONFIG

### Пример

```
esr(config)# ip prefix-list ospf_in
```

### 23.2.2 *ipv6 prefix-list*

Данной командой создается список IPv6-подсетей, который в дальнейшем будет использоваться для фильтрации анонсируемых и получаемых IPv6-маршрутов.

Использование отрицательной формы команды (no) удаляет список префиксов.

#### Синтаксис

[no] ipv6 prefix-list <NAME>

#### Параметры

<NAME> – имя конфигурируемого списка подсетей, задаётся строкой до 31 символа.

#### Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример

```
esr(config)# ipv6 prefix-list ospfv3_in
```

### 23.2.3 *permit/deny*

Данной командой разрешаются (permit) или запрещаются (deny) списки префиксов.

#### Синтаксис

permit { object-group <OBJ-GROUP-NETWORK-NAME> [ { eq <LEN> | le <LEN> | ge <LEN> [ le <LEN> ] } ] | default-route }

deny { object-group <OBJ-GROUP-NETWORK-NAME> [ { eq <LEN> | le <LEN> | ge <LEN> [ le <LEN> ] } ] | default-route }

no {object-group <OBJ-GROUP-NETWORK-NAME>|default-route}

#### Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP/IPv6-адресов, задаётся строкой до 31 символа;

<LEN> – длина префикса, принимает значения [1..32] в IP-списках префиксов и [1..128] в IPv6-списках префиксов;

eq – при указании команды длина префикса должна соответствовать указанной;

le – при указании команды длина префикса должна быть меньше либо соответствовать указанной;

ge – при указании команды длина префикса должна быть больше либо соответствовать указанной;

default-route – фильтрация маршрута по умолчанию.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-PL

CONFIG-IPV6-PL

## Пример

```
esr(config-pl)# permit static ge 24 le 28
```

### 23.2.4 *redistribute static*

Данной командой включается анонсирование статических маршрутов.

Использование отрицательной формы команды (no) отключает анонсирование статических маршрутов.

## Синтаксис

```
redistribute static [ route-map <NAME> ]
```

```
no redistribute static
```

## Параметры

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых статических маршрутов, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP

CONFIG-OSPF

CONFIG-RIP

CONFIG-OSPFV3

## Пример

```
esr(config-bgp)# redistribute static
```

### 23.2.5 *redistribute connected*

Данной командой включается анонсирование напрямую подключенных подсетей.

Использование отрицательной формы команды (no) отключает анонсирование напрямую подключенных подсетей.

## Синтаксис

```
redistribute connected [ route-map <NAME> ]
```

```
no redistribute connected
```

## Параметры

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых напрямую подключенных подсетей, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP

CONFIG-OSPF

CONFIG-RIP

CONFIG-OSPFV3

## Пример

```
esr(config-rip)# redistribute connected
```

### 23.2.6 *redistribute rip*

Данной командой включается анонсирование маршрутов из базы маршрутов RIP.

Использование отрицательной формы команды (no) отключает анонсирование маршрутов из базы RIP.

## Синтаксис

redistribute rip [ route-map <NAME> ]

no redistribute rip

## Параметры

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых RIP-маршрутов, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP

CONFIG-OSPF

## Пример

```
esr(config-bgp)# redistribute rip
```

### 23.2.7 *redistribute ospf*

Данной командой включается анонсирование маршрутов из базы OSPF-процесса согласно выбранным условиям.

Использование отрицательной формы команды (no) отключает анонсирование маршрутов из базы OSPF-процесса.

## Синтаксис

```
redistribute ospf <ID> <ROUTE-TYPE> [ route-map <NAME> ]
```

```
no redistribute ospf <ID>
```

### Параметры

<ID> – номер процесса, может принимать значение [1..65535];

<ROUTE-TYPE> – тип маршрута:

intra-area – анонсирование маршрутов OSPF-процесса в пределах зоны;

inter-area – анонсирование маршрутов OSPF-процесса между зонами;

external1 – анонсирование внешних маршрутов OSPF-формата 1;

external2 – анонсирование внешних маршрутов OSPF-формата 2;

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых OSPF-маршрутов, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP

CONFIG-OSPF

CONFIG-RIP

CONFIG-OSPFV3

### Пример

```
esr(config-bgp)# redistribute ospf 10 external2
```

## 23.2.8 *redistribute bgp*

Данной командой включается анонсирование маршрутов автономной системы BGP.

Использование отрицательной формы команды (no) отключает анонсирование маршрутов автономной системы BGP.

### Синтаксис

```
redistribute bgp <AS> [ route-map <NAME> ]
```

```
no redistribute bgp <AS>
```

### Параметры

<AS> – номер автономной системы, может принимать значения [1..4294967295];

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых BGP-маршрутов, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP

CONFIG-OSPF

## CONFIG-RIP

### Пример

```
esr(config-bgp)# redistribute bgp 30
esr(config-ospf)# redistribute bgp 35
esr(config-rip)# redistribute bgp 300
```

## 23.2.9 *network*

Данной командой включается анонсирование указанной подсети.

Использование отрицательной формы команды (no) отключает анонсирование указанной подсети.

### Синтаксис

[no] network <ADDR/LEN>

### Параметры

<ADDR/LEN> – адрес подсети, имеет один из следующих форматов:

AAA.BBB.CCC.DDD/EE – IP-адрес подсети с маской в форме префикса, где AAA-DDD принимают значения [0..255] и EE принимает значения [1..32];

X:X:X:X::X/EE – IPv6-адрес и маска подсети, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP

CONFIG-OSPF-AREA

CONFIG-RIP

CONFIG-OSPFV3-AREA

### Пример

```
esr(config-bgp)# network 192.168.25.0/24
```

## 23.2.10 *prefix-list*

Данной командой добавляется фильтрация подсетей во входящих или исходящих обновлениях.

Использование отрицательной формы команды (no) отключает фильтрацию.

### Синтаксис

prefix-list <PREFIX-LIST-NAME> { in | out }

no prefix-list { in | out }

### Параметры

<PREFIX-LIST-NAME> – имя сконфигурированного списка подсетей, задаётся строкой до 31 символа:

in – фильтрация входящих маршрутов;

out – фильтрация анонсируемых маршрутов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP-NEIGHBOR

CONFIG-OSPF

CONFIG-RIP

CONFIG-OSPFV3

### Пример

```
esr(config-rip)# prefix-list rip_in in
```

## 23.3 Маршрутизация на основе политик (PBR)

### 23.3.1 route-map

Данной командой создается маршрутная карта, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых и получаемых IP-маршрутов, и осуществляется переход в режим настройки параметров маршрутной карты.

Использование отрицательной формы команды (no) удаляет маршрутную карту.

### Синтаксис

[no] route-map <NAME>

### Параметры

<NAME> – имя конфигурируемой маршрутной карты, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# route-map drop-local-net
esr(config-route-map)#
```

### 23.3.2 rule

Данной командой создается правило маршрутной карты с определённым номером и осуществляется переход в режим настройки параметров правила. Правила обрабатываются устройством в порядке возрастания номеров правил.

Использование отрицательной формы команды (no) удаляет правило по номеру либо все правила.

### Синтаксис

[no] rule <ORDER>

## Параметры

<ORDER> – номер правила, принимает значения [1 .. 10000].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP

## Пример

```
esr(config-route-map)# rule 2
esr(config-route-map-rule)#
```

### 23.3.3 *description*

Данная команда используется для изменения описания конфигурируемого правила маршрутной карты.

Использование отрицательной формы команды (no) удаляет установленное описание.

## Синтаксис

description <DESCRIPTION>

no description

## Параметры

<DESCRIPTION> – описание правила маршрутной карты, задаётся строкой до 255 символов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# description "Drop Local NETs"
```

### 23.3.4 *action*

Данная команда используется для указания действия, которое должно быть применено для маршрутной информации, удовлетворяющей заданным критериям.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

action <ACT>

no action

## Параметры

<ACT> – назначаемое действие:

9. permit – прием/анонсирование маршрутной информации разрешено;
10. deny – прием/анонсирование маршрутной информации запрещено.

## Значение по умолчанию

permit

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-acl-rule)# action deny
```

### 23.3.5 *match as-path*

Данной командой устанавливается значение атрибута BGP AS-Path в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

match as-path [ begin | end | contain ] <AS-PATH>

no match as-path

#### Параметры

<AS-PATH> – список номеров автономных систем, задаётся в виде AS,AS,AS, где каждая часть принимает значения [1..4294967295]. Можно указать до 10 номеров автономных систем.

Оptionальные параметры, с помощью которых задаётся частичное соответствие атрибута:

begin – значение атрибута начинается с указанного списка номеров автономных систем;

end – значение атрибута оканчивается на указанный список номеров автономных систем;

contain – значение атрибута содержит указанный список номеров автономных систем.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# match as-path begin 100,200,300
```

### 23.3.6 *match community*

Данной командой устанавливается значение атрибута BGP Community в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

match community <COMMUNITY-LIST>

no match community



## Параметры

<COMMUNITY-LIST> – список community, задаётся в виде AS:N,AS:,AS, где каждая часть принимает значения [1..4294967295]. Можно указать до 64 community.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# match community 100:1,200:3,300:65000
```

## 23.3.7 match extcommunity

Данной командой устанавливается значение атрибута BGP ExtCommunity в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

match extcommunity <EXTCOMMUNITY-LIST>

no match extcommunity

## Параметры

<EXTCOMMUNITY-LIST> – список extcommunity, задаётся в виде KIND:AS:N, KIND:AS:N, KIND:AS:N, где

- KIND – тип extcommunity, принимает значения RT (Route Target) и RO (Route Origin);
- AS – номер автономной системы, принимает значения [1..4294967295];
- N – номер extcommunity, определяющий политику маршрутизации трафика, принимает значения [1..65535].

Можно указать до 64 extcommunity.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# match extcommunity ro:435:6
```

## 23.3.8 match ip access-group

Данной командой устанавливается ACL группа, для которой должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

match match ip access-group <NAME>

no match match ip access-group

## Параметры

<NAME> – имя списка контроля доступа, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# match ip access-group ACCESS
```

### 23.3.9 *match ip address*

Данной командой устанавливается профиль IP-адресов, содержащий значения подсетей назначения в маршруте, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

match ip address object-group <OBJ-GROUP- NETWORK -NAME>

no match ip address

## Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который должен содержать префиксы подсетей назначения, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# match ip address object-group local_nets
```

### 23.3.10 *match ip next-hop*

Данной командой устанавливается профиль IP-адресов, содержащий значения атрибута BGP Next-Hop в маршруте, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

match ip next-hop object-group <OBJ-GROUP-NETWORK-NAME>

no match ip next-hop

## Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который должен содержать диапазоны IP-адресов шлюзов, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# match ip next-hop object-group block_nexthop
```

### 23.3.11 *match ip route-source*

Командой устанавливается профиль IP-адресов. Профиль содержит IP-адреса маршрутизатора, анонсировавшего маршрут. Используется для фильтрации по IP-адресу источника при анонсировании маршрутной информации.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

match ip route-source object-group <OBJ-GROUP-NETWORK-NAME>

no match ip route-source

## Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который должен содержать диапазоны IP-адресов источника маршрутной информации, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# match ip route-source object-group source_routers
```

### 23.3.12 *match ipv6 address*

Данной командой устанавливается профиль IPv6-адресов, содержащий значения подсетей назначения в маршруте, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

match ipv6 address object-group <OBJ-GROUP-NETWORK-NAME>

no match ipv6 address

## Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IPv6-адресов, который должен содержать префиксы подсетей назначения, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# match ipv6 address object-group local_nets
```

### 23.3.13 *match ipv6 next-hop*

Данной командой устанавливается профиль IPv6-адресов, содержащий значения атрибута BGP Next-Hop в маршруте, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

```
match ipv6 next-hop object-group <OBJ-GROUP-NETWORK-NAME>
```

```
no match ipv6 next-hop
```

#### Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IPv6-адресов, который должен содержать диапазоны IPv6-адресов шлюзов, задаётся строкой до 31 символа.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# match ipv6 next-hop object-group block_nexthop
```

### 23.3.14 *match ipv6 route-source*

Командой устанавливается профиль IPv6-адресов. Профиль содержит IPv6-адреса маршрутизатора, анонсировавшего маршрут. Используется для фильтрации по IPv6-адресу источника при анонсировании маршрутной информации.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

```
match ipv6 route-source object-group <OBJ-GROUP-NETWORK-NAME>
```

```
no match ipv6 route-source
```

#### Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IPv6-адресов, который должен содержать диапазоны IPv6-адресов источника маршрутной информации, задаётся строкой до 31 символа.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# match ipv6 route-source object-group source_routers
```

### 23.3.15 *match metric bgp*

Данной командой устанавливается значение атрибута BGP MED в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

match metric <METRIC>

no match metric

#### Параметры

<METRIC> – значение атрибута BGP MED, принимает значения [0..4294967295].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-ROUTE-MAP-RULE

#### Пример

```
esr(config-route-map-rule)# match metric bgp 10
```

### 23.3.16 *match metric ospf*

Данной командой устанавливается значение атрибута OSPF Metric в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

match metric ospf <TYPE> <METRIC>

no match metric ospf

#### Параметры

<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2;

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-ROUTE-MAP-RULE

#### Пример

```
esr(config-route-map-rule)# match metric ospf type-1 10
```

### 23.3.17 *match metric rip*

Данной командой устанавливается значение атрибута RIP Metric в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match metric rip <METRIC>
no match metric rip
```

## Параметры

<METRIC> – значение атрибута RIP Metric, принимает значения [0..16].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# match metric rip 5
```

### 23.3.18 *match tag ospf*

Данной командой устанавливается значение атрибута OSPF Tag в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
match tag ospf <TAG>
no match tag ospf
```

## Параметры

<TAG> – значение атрибута OSPF Tag, принимает значения [0..4294967295].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# match tag ospf 20
```

### 23.3.19 *match tag rip*

Данной командой устанавливается значение атрибута RIP Tag в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
action set tag rip <RIP>
no action set tag rip
```

## Параметры

<RIP> – значение атрибута RIP Tag, принимает значения [0..65535].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-ROUTE-MAP-RULE

#### Пример

```
esr(config-route-map-rule)# match tag rip 20
```

### 23.3.20 *action set as-path prepend*

Данной командой устанавливается значение атрибута BGP AS-Path, которое будет добавляться в начало списка автономных систем в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

action set as-path prepend <AS-PATH>

no match as-path

#### Параметры

<AS-PATH> – список номеров автономных систем, который будет добавлен к текущему значению в маршруте. Задаётся в виде AS,AS,AS, где каждая часть принимает значения [1..4294967295]. Можно указать до 10 номеров автономных систем.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-ROUTE-MAP-RULE

#### Пример

```
esr(config-route-map-rule)# action set as-path prepend 100,200,300
```

### 23.3.21 *action set community*

Данной командой задается значение атрибута BGP Community, которое будет установлено в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

action set community { <COMMUNITY-LIST> | no-advertise | no-export }

no action set community

#### Параметры

<COMMUNITY-LIST> – список community, задаётся в виде AS:N,AS:N,AS:N, где каждая часть принимает значения [1..65535]. Можно указать до 64 community;

no-advertise – при указании команды маршруты, которые передаются с данным значением атрибута community, не должны анонсироваться другим BGP-соседям;

no-export – при указании команды маршруты, которые передаются с таким значением атрибута community, не должны анонсироваться за пределы конфедерации (автономная система, которая не является частью конфедерации считается конфедерацией). То есть, маршруты не анонсируются eBGP-соседям, но анонсируются внешним соседям в конфедерации.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# action set community no-advertise
```

## 23.3.22 *action set extcommunity*

Данной командой задается значение атрибута BGP ExtCommunity, которое будет установлено в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

action set extcommunity <EXTCOMMUNITY-LIST>

no action set extcommunity

### Параметры

<EXTCOMMUNITY-LIST> – список community, задаётся в виде KIND:AS:N,KIND:AS:N,KIND:AS:N, где

- KIND – тип extcommunity, принимает значения RT (Route Target) и RO (Route Origin);
- AS – номер автономной системы, принимает значения [1..4294967295];
- N – номер extcommunity, определяющий политику маршрутизации трафика, принимает значения [1..65535].

Можно указать до 64 ExtCommunity.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-ROUTE-MAP-RULE

### Пример

```
esr(config-route-map-rule)# action set extcommunity ro:435:6
```

## 23.3.23 *action set ip bgp-next-hop*

Данной командой задается значение атрибута BGP Next-Hop, которое будет установлено в маршруте при анонсировании по BGP.

Использование отрицательной формы команды (no) отменяет назначение.



## Синтаксис

```
action set ip bgp-next-hop <ADDR>  
no action set ip next-hop
```

## Параметры

<ADDR> – IP-адрес шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# action set ip bgp-next-hop 10.100.100.1
```

### 23.3.24 *action set ip next-hop*

Данной командой задается значение Next-Хор, которое будет установлено в маршруте, полученном по BGP.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

```
action set ip next-hop { <NEXTHOP> | blackhole | unreachable | prohibit }  
no action set ip next-hop [address]
```

## Параметры

<NEXTHOP> – IP-адрес шлюза задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

– blackhole – при указании команды пакеты до данной подсети будут удаляться устройством без отправки уведомлений отправителю;

– unreachable – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Host unreachable, code 1);

– prohibit – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Communication administratively prohibited, code 13);

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# action set ip next-hop prohibit
```

---

### **23.3.25**      ***action set ipv6 bgp-next-hop***

---

Данной командой задается значение атрибута BGP Next-Hop для IPv6, которое будет установлено в маршруте при анонсировании по BGP.

Использование отрицательной формы команды (no) отменяет назначение.

#### **Синтаксис**

action set ipv6 bgp-next-hop <ADDR>

no action set ipv6 next-hop

#### **Параметры**

<IPv6-ADDR> – IPv6-адрес шлюза, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-ROUTE-MAP-RULE

#### **Пример**

```
esr(config-route-map-rule)# action set ipv6 bgp-next-hop 2002::765:1
```

---

### **23.3.26**      ***action set ipv6 next-hop***

---

Данной командой задается значение Next-Hop для IPv6, которое будет установлено в маршруте, полученном по BGP.

Использование отрицательной формы команды (no) отменяет назначение.

#### **Синтаксис**

action set ipv6 next-hop <NEXTHOP>

no action set ipv6 next-hop

#### **Параметры**

<NEXTHOP> – IPv6-адрес шлюза, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-ROUTE-MAP-RULE

#### **Пример**

```
esr(config-route-map-rule)# action set ipv6 next-hop 55::205:2
```

### 23.3.27 *action set ip next-hop verify-availability*

Данной командой задается Next-Хор для пакетов, которые попадают под критерия в указанном списке доступа (ACL).

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

```
action set ip next-hop verify-availability <NEXTHOP> <METRIC>
no action set ip next-hop verify-availability {<NEXTHOP>| all}
```

#### Параметры

<NEXTHOP> – IP-адрес шлюза задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

[METRIC] – метрика маршрута, принимает значения [0..255].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-ROUTE-MAP-RULE

#### Пример

```
esr(config-route-map-rule)# action set ip next-hop verify-availability 1.1.1.1 25
```

### 23.3.28 *action set local-preference*

Данной командой задается значение атрибута BGP Local Preference, которое будет установлено в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

```
action set local-preference <PREFERENCE>
no action set local-preference
```

#### Параметры

<PREFERENCE> – значение атрибута BGP Local Preference, принимает значения [0..255].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-ROUTE-MAP-RULE

#### Пример

```
esr(config-route-map-rule)# action set local-preference 120
```

### 23.3.29 *action set origin*

---

Данной командой задается значение атрибута BGP Origin, которое будет установлено в маршруте. Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

```
action set origin <ORIGIN>
no action set origin
```

#### Параметры

<ORIGIN> – значение атрибута BGP Origin, принимает следующие значения:

egp – маршрут выучен по протоколу Exterior Gateway Protocol (EGP);

igr – маршрут получен внутри исходной автономной системы;

incomplete – маршрут выучен другим образом.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-ROUTE-MAP-RULE

#### Пример

```
esr(config-route-map-rule)# action set origin igp
```

### 23.3.30 *action set metric bgp*

---

Данной командой задается значение атрибута BGP MED, которое будет установлено в маршруте. Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

```
action set metric <METRIC>
no action set metric
```

#### Параметры

<METRIC> – значение атрибута BGP MED, принимает значения [0..4294967295].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-ROUTE-MAP-RULE

#### Пример

```
esr(config-route-map-rule)# action set metric bgp 10
```

### 23.3.31 *action set metric ospf*

---

Данной командой задается значение атрибута OSPF Metric, которое будет установлено в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

action set metric ospf <TYPE> <METRIC>

no action set metric ospf

#### Параметры

<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2;

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-ROUTE-MAP-RULE

#### Пример

```
esr(config-route-map-rule)# action set metric ospf type-1 10
```

### 23.3.32 *action set metric rip*

Данной командой задается значение атрибута RIP Metric, которое будет установлено в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

action set metric rip <METRIC>

no action set metric rip

#### Параметры

<METRIC> – значение атрибута RIP Metric, принимает значения [0..16].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-ROUTE-MAP-RULE

#### Пример

```
esr(config-route-map-rule)# action set metric rip 5
```

### 23.3.33 *action set tag ospf*

Данной командой задается значение атрибута OSPF Tag, которое будет установлено в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

#### Синтаксис

action set tag ospf <TAG>

no action set tag ospf

## Параметры

<TAG> – значение атрибута OSPF Tag, принимает значения [0..4294967295].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# action set tag ospf 20
```

### 23.3.34 *action set tag rip*

Данной командой задается значение атрибута RIP Tag, которое будет установлено в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

## Синтаксис

action set tag rip <RIP>

no action set tag rip

## Параметры

<RIP> – значение атрибута RIP Tag, принимает значения [0..65535].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-ROUTE-MAP-RULE

## Пример

```
esr(config-route-map-rule)# action set tag rip 20
```

### 23.3.35 *route-map*

Командой добавляется фильтрация и модификация маршрутов во входящих или исходящих направлениях.

Использование отрицательной формы команды (no) отключает фильтрацию и модификацию маршрутов в соответствующем направлении.

## Синтаксис

route-map <NAME> <DIRECTION>

no route-map <DIRECTION>

## Параметры

<NAME> – имя сконфигурированной маршрутной карты, задается строкой до 31 символа;

<DIRECTION> – направление:

in – фильтрация и модификация получаемых маршрутов;

out – фильтрация и модификация анонсируемых маршрутов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP-NEIGHBOR

CONFIG-OSPF

CONFIG-OSPFV3

CONFIG-RIP

### Пример

```
esr(config-bgp-neighbor)# route-map drop-local-net in
```

## 23.3.36 *ip policy route-map*

Данной командой на интерфейс назначается политика маршрутизации на основе списков доступа (ACL).

Использование отрицательной формы команды (no) удаляет политику маршрутизации.

### Синтаксис

ip policy route-map <NAME>

no ip policy route-map

### Параметры

<NAME> – имя сконфигурированной политики маршрутизации, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

### Пример

```
esr(config-subif)# route-map drop-local-net in
```

## 23.3.37 *show ip route-map*

Данная команда используется для просмотра маршрутных карт.

## Синтаксис

show ip route-map <NAME> [ <ORDER> ]

## Параметры

<NAME> – имя маршрутной карты, задаётся строкой до 31 символа;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будет показана информация только по данному правилу.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show ip route-map drop-local-net
Order:                                     2
Description:                             Drop route to private nets
Matching pattern:
  Access group                           --
  AS path                                --
  Community                              --
  Extcommunity                           --
  BGP metric (MED):                      --
  Address (object-group):                 local_net
  Next hop (object-group):                --
  Route source (object-group):            --
  RIP metric                             --
  RIP tag                                --
  OSPF metric type                        --
  OSPF metric                            --
  OSPF tag                                --
Actions:
  Decision:                              Deny
  Route next hop address:                 --
  Route IPv6 next hop address:            --
  Route next hop:                         --
  AS path (prepend):                      --
  Community:                              --
  Extcommunity:                           --
  Local preference:                       --
  BGP next hop address:                   --
  BGP IPv6 next hop address:              --
  BGP metric (MED):                      --
  Origin:                                 --
  RIP metric                             --
  RIP tag                                --
  OSPF metric type                        --
  OSPF metric                            --
  OSPF tag                                --
```



---

## 23.4 Настройка связей ключей

---

### 23.4.1 *key-chain*

---

Командой добавляется связка ключей в систему и осуществляется переход в режим настройки параметров связки ключей.

Использование отрицательной формы команды (no) удаляет указанный список.

#### Синтаксис

[no] key-chain <KEYCHAIN>

#### Параметры

<KEYCHAIN> – идентификатор списка ключей, строка до 16 ASCII символов.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

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

### 23.4.2 *key*

---

Данной командой добавляется ключ в связку ключей и осуществляется переход в режим настройки параметров ключа.

Использование отрицательной формы команды (no) удаляет указанный ключ.

#### Синтаксис

[no] key <ID>

#### Параметры

<ID> – идентификатор ключа, задается в диапазоне [0..255].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-KEYCHAIN

#### Пример

```
esr(config-keychain)# key 25
```

### 23.4.3 *key-string*

---

Данной командой устанавливается пароль для аутентификации.

Использование отрицательной формы команды (no) удаляет пароль.

## Синтаксис

key-string ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }

no key-string

## Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-KEYCHAIN-KEY

## Пример

```
esr(config-keychain-key)# authentication key ascii-text 123456789
esr(config-keychain-key)# authentication key ascii-text encrypted
CDE65039E5591FA3F1
```

### 23.4.4 *send-lifetime*

Данная команда определяет период времени, в течение которого данный ключ может использоваться для аутентификации при отправке пакетов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

send-lifetime <TIME\_B> <DAY\_B> <MONTH\_B> <YEAR\_B> <TIME\_E> <DAY\_E> <MONTH\_E>  
<YEAR\_E>

no send-lifetime

## Параметры

<TIME\_B> – устанавливаемое время начала использования ключа, задаётся в виде HH:MM:SS, где:

HH – часы, принимает значение [0..23];

MM – минуты, принимает значение [0 .. 59];

SS – секунды, принимает значение [0 .. 59].

<DAY\_B> – день месяца начала использования ключа, принимает значения [1..31];

<MONTH\_B> – месяц начала использования ключа, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];

<YEAR\_B> – год начала использования ключа, принимает значения [2001..2037];

<TIME\_E> – устанавливаемое время окончания использования ключа, задаётся в виде HH:MM:SS, где:

HH – часы, принимает значение [0..23];

MM – минуты, принимает значение [0 .. 59];

SS – секунды, принимает значение [0 .. 59].

<DAY\_E> – день месяца окончания использования ключа, принимает значения [1..31];

<MONTH\_E> – месяц окончания использования ключа, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];

<YEAR\_E> – год окончания использования ключа, принимает значения [2001..2037].

### Значение по умолчанию

Ключ действителен постоянно.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-KEYCHAIN-KEY

### Пример

```
esr(config-keychain-key)# send-lifetime 16:35:00 15 May 2014 16:35:00 21 June 2018
```

## 23.4.5 *accept-time*

Данная команда определяет период времени, в течение которого данный ключ может использоваться для аутентификации принятых пакетов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

accept-time <TIME\_B> <DAY\_B> <MONTH\_B> <YEAR\_B> <TIME\_E> <DAY\_E> <MONTH\_E>  
<YEAR\_E>

no accept-time

### Параметры

<TIME\_B> – устанавливаемое время начала использования ключа, задаётся в виде HH:MM:SS, где:

HH – часы, принимает значение [0..23];

MM – минуты, принимает значение [0 .. 59];

SS – секунды, принимает значение [0 .. 59].

<DAY\_B> – день месяца начала использования ключа, принимает значения [1..31];

<MONTH\_B> – месяц начала использования ключа, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];

<YEAR\_B> – год начала использования ключа, принимает значения [2001..2037].

<TIME\_E> – устанавливаемое время окончания использования ключа, задаётся в виде HH:MM:SS, где:

HH – часы, принимает значение [0..23];

MM – минуты, принимает значение [0 .. 59];

SS – секунды, принимает значение [0 .. 59].

<DAY\_E> – день месяца окончания использования ключа, принимает значения [1..31];

<MONTH\_E> – месяц окончания использования ключа, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];

<YEAR\_E> – год окончания использования ключа, принимает значения [2001..2037].

### Значение по умолчанию

Ключ действителен постоянно.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-KEYCHAIN-KEY

### Пример

```
esr(config-keychain-key)# accept-time 16:35:00 10 May 2015 16:35:00 10 June 2021
```

## 23.5 Настройка статических маршрутов IPv4/IPv6

### 23.5.1 *ip route*

Команда позволяет создать статический IP-маршрут к указанной подсети.

Использование отрицательной формы команды (no) удаляет указанный маршрут.

#### Синтаксис

ip route [ vrf <VRF> ] <SUBNET> { <NEXTHOP> [ resolve ] | interface <IF> | tunnel <TUN> | wan load-balance rule <RULE> | blackhole | unreachable | prohibit } [ <METRIC> ]

no ip route [ vrf <VRF> ] <SUBNET> [ <METRIC> ]

#### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<SUBNET> – адрес назначения, может быть задан в следующих видах:

AAA.BBB.CCC.DDD – IP-адрес хоста, где каждая часть принимает значения [0..255];

AAA.BBB.CCC.DDD/NN – IP-адрес подсети с маской в виде префикса, где AAA-DDD принимают значения [0..255] и NN принимает значения [1..32].

<NEXTHOP> – IP-адрес шлюза задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

resolve – при указании данного параметра IP-адрес шлюза будет рекурсивно вычислен через таблицу маршрутизации. Если при рекурсивном вычислении не удастся найти шлюз из напрямую подключенной подсети, то данный маршрут не будет установлен в систему;

<IF> – имя IP-интерфейса, задаётся в виде, описанном в разделе 3.3;

<TUN> – имя туннеля, задаётся в виде, описанном в разделе 3.4;

<RULE> – номер правила wan, задаётся в диапазоне [1..50];

blackhole – при указании команды пакеты до данной подсети будут удаляться устройством без отправки уведомлений отправителю;

unreachable – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Host unreachable, code 1);

prohibit – при указании команды, пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Communication administratively prohibited, code 13);



**Если в качестве подсети указать 0.0.0.0/0, то будет задан маршрут по умолчанию.**

[METRIC] – метрика маршрута, принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример 1

Задать маршрут до подсети 192.165.3.0/24 с метрикой 6 через шлюз 192.165.56.65:

```
esr(config)# ip route 192.165.3.0/24 192.165.56.65 6
```

### Пример 2

Задать маршрут до подсети 192.165.3.0/24 с метрикой 6 через интерфейс GigabitEthernet 1/0/5:

```
esr(config)# ip route 192.165.3.0/24 interface gigabitethernet 1/0/5 6
```

### Пример 3

Задать маршрут до подсети 192.165.3.0/24 через туннельный интерфейс vti 1:

```
esr(config)# ip route 192.165.3.0/24 interface vti 1
```

## 23.5.2 *ipv6 route*

Данная команда позволяет создать статический IPv6-маршрут к указанной подсети.

Использование отрицательной формы команды (no) удаляет указанный маршрут.

### Синтаксис

```
ipv6 route [ vrf <VRF> ] <SUBNET> { <NEXTHOP> [ resolve ] | interface <IF> | wan load-balance rule <RULE> | blackhole | unreachable | prohibit } [ <METRIC> ]
```

```
no ipv6 route [ vrf <VRF> ] <SUBNET> [ <METRIC> ]
```

### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<SUBNET> – адрес назначения, может быть задан в следующих видах:

X:X:X:X – IPv6-адрес хоста, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

X:X:X:X/EE – IPv6-адрес подсети с маской в виде префикса, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].

<NEXTHop> – IPv6-адрес шлюза, задается в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

resolve – при указании данного параметра IPv6-адрес шлюза будет рекурсивно вычислен через таблицу маршрутизации. Если при рекурсивном вычислении не удастся найти шлюз из напрямую подключенной подсети, то данный маршрут не будет установлен в систему;

<IF> – имя IP-интерфейса, задается в виде, описанном в разделе 3.3;

blackhole – при указании команды пакеты до данной подсети будут удаляться устройством без отправки уведомлений отправителю;

unreachable – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Host unreachable, code 1);

prohibit – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Communication administratively prohibited, code 13);



**Если в качестве подсети указать ::/0, то будет задан маршрут по умолчанию.**

[METRIC] – метрика маршрута, принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример 1

Задать маршрут до подсети 2001::/120 с метрикой 6 через шлюз fc00::1:

```
esr(config)# ipv6 route 2001::/120 fc00::1 6
```

### Пример 2

Задать маршрут до подсети 2001::/120 с метрикой 6 через интерфейс GigabitEthernet 1/0/5:

```
esr(config)# ipv6 route 2001::/120 interface gigabitethernet 1/0/5 6
```

### Пример 3

Задать маршрут до подсети 2001::/120 через туннельный интерфейс vti 1:

```
esr(config)# ipv6 route 2001::/120 interface vti 1
```

---

## 23.6 Настройка протокола BGP

---

### 23.6.1 *router bgp*

---

Данной командой добавляется BGP-процесс в систему и осуществляется переход в режим настройки параметров BGP-процесса.

Использование отрицательной формы команды (no) удаляет BGP-процесс из системы.

#### Синтаксис

[no] router bgp <AS>

#### Параметры

<AS> – номер автономной системы процесса, принимает значения [1..4294967295].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

```
esr(config)# router bgp 1000
esr(config-bgp)#
```

Добавлен BGP-процесс с автономной системой 1000.

### 23.6.1 *router bgp maximum-paths*

---

Данной командой включается ECMP и определяется максимальное количество равноценных маршрутов до цели.

Использование отрицательной формы команды (no) отключает ECMP.

#### Синтаксис

router bgp maximum-paths <VALUE>

no router bgp maximum-paths

#### Параметры

<VALUE> – количество допустимых равноценных маршрутов до цели, принимает значения [1..16].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

```
esr(config)# router bgp maximum-paths 14
```

## 23.6.2 *address-family*

---

Данной командой определяется тип конфигурируемой маршрутной информации и переход в данный режим настройки.

Использование отрицательной формы команды (no) удаляет идентификатор.

### Синтаксис команды

```
[no] address-family { ipv4 | ipv6 } [ vrf <VRF> ]
```

### Параметры

- ipv4 – семейство IPv4;
- ipv6 – семейство IPv6;
- <VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP

### Пример

```
esr(config-bgp)# address-family ipv4 vrf OFFICE
```

## 23.6.3 *router-id*

---

Данной командой устанавливается идентификатор маршрутизатора.

Использование отрицательной формы команды (no) удаляет идентификатор.

### Синтаксис команды

```
router-id <ID>
```

```
no router-id
```

### Параметры

<ID> – идентификатор маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-BGP-FAMILY

### Пример

```
esr(config-bgp-af)# router-id 1.1.1.1
```



## 23.6.4 *timers keepalive*

Данной командой устанавливается временной интервал, по истечении которого идет проверка соединения со встречной стороной.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
timers keepalive <TIME>
no timers keepalive
```

### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-BGP-FAMILY
CONFIG-BGP -NEIGHBOR
```

### Значение по умолчанию

60 секунд

### Пример

```
esr(config-bgp-af)# timers keepalive 120
```

## 23.6.5 *timers holdtime*

Данной командой устанавливается временной интервал, по истечении которого встречная сторона считается недоступной. Таймер запускается после установления отношений соседства и начинает отсчет от 0. Таймер сбрасывается при получении каждого ответа на keepalive сообщение от встречной стороны. Рекомендуется устанавливать значение таймера равное  $3 * keepalive$ .

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
timers holdtime <TIME>
no timers holdtime
```

### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-BGP-FAMILY
CONFIG-BGP-NEIGHBOR
```

## Значение по умолчанию

180 секунд

## Пример

```
esr(config-bgp-af)# timers holdtime 360
```

### 23.6.6 neighbor

Данной командой добавляется BGP-сосед и осуществляется переход в режим настройки параметров BGP-соседа. Использование отрицательной формы команды (no) удаляет параметры соседнего маршрутизатора из конфигурации.

## Синтаксис

```
[no] neighbor {<ADDR> | <IPV6-ADDR>}
```

## Параметры

<ADDR> – IP адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<IPV6-ADDR> IPv6-адрес клиента, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP-FAMILY

## Пример

```
esr(config-bgp-af)# neighbor 192.168.0.2
esr(config-bgp-neighbor)#
```

### 23.6.7 remote-as

Данной командой устанавливается номер автономной системы BGP-соседа.

Использование отрицательной формы команды (no) удаляет номер автономной системы.

## Синтаксис

```
remote-as <AS>
no remote-as
```

## Параметры

<AS> – номер автономной системы, принимает значения [1..4294967295].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP-NEIGHBOR

## Пример

```
esr(config-bgp-neighbor)# remote-as 2000
```

### 23.6.8 *ebgp-multihop*

Данной командой разрешается подключение к соседям, которые находятся не в напрямую подключенных подсетях.

Использование отрицательной формы команды (no) отключает данную функцию.

#### Синтаксис

```
[no] ebgp-multihop
```

#### Параметры

Команда не содержит аргументов.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-BGP-NEIGHBOR

#### Пример

```
esr(config-bgp-neighbor)# ebgp-multihop
```

### 23.6.9 *next-hop-self*

Данной командой задается режим, в котором все обновления отправляются BGP-соседу с указанием в качестве next-hop IP-адреса исходящего интерфейса локального маршрутизатора.

Использование отрицательной формы команды (no) отключает данную функцию.

#### Синтаксис

```
[no] next-hop-self
```

#### Параметры

Команда не содержит аргументов.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-BGP-NEIGHBOR

#### Пример

```
esr(config-bgp-neighbor)# next-hop-self
```

### 23.6.10 *remove-private-as*

Данной командой задается режим, в котором перед отправлением обновления из BGP атрибута AS Path маршрутов удаляются приватные номера автономных систем (в соответствии с RFC 6996).

Использование отрицательной формы команды (no) отключает данную функцию.

#### Синтаксис

```
[no] remove-private-as
```

## Параметры

Команда не содержит аргументов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP-NEIGHBOR

## Пример

```
esr(config-bgp-neighbor) # remove-private-as
```

### 23.6.11 *allow-local-as*

Данной командой задается режим, в котором разрешен приём маршрутов в BGP атрибуте AS Path которых содержатся номера автономной системы процесса.

Использование отрицательной формы команды (no) отключает данную функцию.

## Синтаксис

allow-local-as <NUMBER>

no allow-local-as

## Параметры

<NUMBER> – пороговое число вхождений номера автономной системы процесса в атрибуте AS Path, при которых маршрут будет принят, диапазон допустимых значений [1..10].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP-NEIGHBOR

## Пример

```
esr(config-bgp-neighbor) # allow-local-as
```

### 23.6.12 *default-originate*

Данной командой задается режим, в котором BGP-соседу в обновлении наряду с другими маршрутами отправляется маршрут по умолчанию.

Использование отрицательной формы команды (no) отключает данную функцию.

## Синтаксис

[no] default-originate

## Параметры

Команда не содержит аргументов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP-NEIGHBOR

### Пример

```
esr(config-bgp-neighbor)# default-originate
```

## 23.6.13 *cluster-id*

Командой устанавливается идентификатор Route-Reflector кластера, которому принадлежит BGP-процесс маршрутизатора.

Использование отрицательной формы команды (no) удаляет идентификатор.

### Синтаксис команды

cluster-id <ID>

no cluster-id

### Параметры

<ID> – идентификатор Route-Reflector кластера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP-FAMILY

### Пример

```
esr(config-bgp-af)# cluster-id 1.1.1.1
```

## 23.6.14 *route-reflector-client*

Данной командой указывается, что BGP-сосед является Route-Reflector клиентом.

Использование отрицательной формы команды (no) отключает данную функцию.

### Синтаксис

[no] route-reflector-client

### Параметры

Команда не содержит аргументов.

### Необходимый уровень привилегий

10

## Командный режим

CONFIG-BGP-NEIGHBOR

### Пример

```
esr(config-bgp-neighbor)# route-reflector-client
```

---

### **23.6.15**      *preference*

---

Данная команда определяет приоритетность маршрутов, получаемых от соседа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### **Синтаксис**

preference <VALUE>

no preference

#### **Параметры**

<VALUE> – приоритетность маршрутов соседа, принимает значения в диапазоне [1..255].

#### **Значение по умолчанию**

170

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-BGP-NEIGHBOR

#### **Пример**

```
esr(config-bgp-neighbor)# preference 30
```

---

### **23.6.16**      *authentication algorithm*

---

Данная команда определяет алгоритм аутентификации. Настройки аутентификации для определенного соседа имеют преимущество перед глобальными настройками конфигурации для BGP-процесса.

Использование отрицательной формы команды (no) отключает аутентификацию.

#### **Синтаксис**

authentication algorithm <ALGORITHM>

no authentication algorithm

#### **Параметры**

<ALGORITHM> – алгоритм шифрования:

- md5 – пароль шифруется по алгоритму md5.

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-BGP-FAMILY

CONFIG-BGP-NEIGHBOR

#### **Пример**

```
esr(config-bgp-af)# authentication algorithm md5
```

### 23.6.17 authentication key

Данная команда устанавливает пароль для аутентификации с соседом.

Использование отрицательной формы команды (no) удаляет пароль.

#### Синтаксис

```
authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
```

```
no authentication key
```

#### Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов.

<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-BGP-FAMILY

CONFIG-BGP-NEIGHBOR

#### Пример

```
esr(config-bgp-af) # authentication key ascii-text 123456789
esr(config-bgp-af) # authentication key ascii-text encrypted CDE65039E5591FA3F1
```

### 23.6.18 enable

Данной командой включается BGP-процесс.

Использование отрицательной формы команды (no) отключает BGP-процесс.

#### Синтаксис

```
[no] enable
```

#### Параметры

Команда не содержит аргументов.

#### Значение по умолчанию

Процесс выключен.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-BGP-FAMILY

#### Пример

```
esr(config-bgp-af) # enable
```

---

### **23.6.19**      ***router bgp log-neighbor-changes***

---

Данной командой включается логирование изменений состояния BGP-соседей.

Использование отрицательной формы команды (no) отключает логирование изменений состояния BGP-соседей.

#### **Синтаксис**

[no] router bgp log-neighbor-changes

#### **Параметры**

Команда не содержит аргументов.

#### **Значение по умолчанию**

Логирование выключено.

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG

#### **Пример**

```
esr(config)# router bgp log-neighbor-changes
```

---

### **23.6.20**      ***show ip bgp***

---

Данная команда отображает таблицу маршрутизации BGP или детальную информацию об определенном маршруте при использовании фильтров.

#### **Синтаксис**

show ip bgp [<AS> [vrf <VRF> [<ADDR> |<ADDR/LEN>]] |<ADDR> |<ADDR/LEN>]

#### **Параметры**

<AS> – номер автономной системы процесса, принимает значения [1..4294967295];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации;

<ADDR> – IP-адрес назначения, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ADDR/LEN> – подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

#### **Необходимый уровень привилегий**

1

#### **Командный режим**

ROOT

#### **Пример 1**

```
esr# show ip bgp
Terra# show ip bgp
Status codes: u - unicast, b - broadcast, m - multicast a - anycast
               * - valid, > - best
```



Origin codes: i - IGP, e - EGP, ? - incomplete

| Network           | Next Hop   | Metric | LocPrf | Path |
|-------------------|------------|--------|--------|------|
| *> u 10.0.10.0/24 | 10.115.0.1 |        | 100    | i    |
| *> u 0.0.0.0/0    | 10.115.0.1 |        | 100    | i    |
| * u 14.0.10.0/24  | 10.115.0.1 |        | 100    | i    |

## Пример 2

```
Sword# show ip bgp 75.0.0.0
75.0.0.0/24 via 115.0.0.40 on gil/0/14 [bgp20 2000-01-15] (AS90?)
Administrative Distance: 68
Type: unicast
Origin: Incomplete
AS PATH: 1 30 70 90
Next Hop: 115.0.0.40
MED: 0
Local Preference: 100
Community: (1:555)
Valid, Best
```

## 23.6.21 show ipv6 bgp

Данная команда отображает таблицу маршрутизации BGP или детальную информацию об определенном маршруте при использовании фильтров.

### Синтаксис

```
show ipv6 bgp [<AS> [vrf <VRF> [<IPv6-ADDR> [<IPv6-ADDR/LEN>] [<IPv6-ADDR> [<IPv6-ADDR/LEN>]]]]
```

### Параметры

<AS> – номер автономной системы процесса, принимает значения [1..4294967295];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации;

<IPv6-ADDR> – IPv6-адрес назначения, задается в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<IPv6-ADDR/LEN> – подсеть, задается в виде X:X:X:X/EE, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF], EE принимает значения [1..128].

### Необходимый уровень привилегий

1

### Командный режим

ROOT

## Пример 1

```
esr# show ipv6 bgp 20
Status codes: u - unicast, b - broadcast, m - multicast a - anycast
              * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete

Network      Next Hop      Metric  LocPrf  Path
*> u 222::/120 44:44:44::44  0       100     1 ?
*> u 40::75:0/120 44:44:44::44  0       100     1 ?
*> u 40::77:0/120 44:44:44::44  0       100     1 ?
* u 2002::7300:0/120 44:44:44::44  0       100     1 ?
```

## Пример 2

```
Sword# show ipv6 bgp 20 202::7300:0/120
202::7300:0/120    via 2002::7300:a on gi1/0/14    [bgp20 14:43:48] (AS1000e)
  Administrative Distance: 170
  Type:                unicast
  Origin:               EGP
  AS PATH:              1000
  Next Hop:             2002::7300:a (fe80::6666:b3ff:fe06:cb18)
  MED:                  0
  Local Preference:     188
  Valid, Best
```

### 23.6.22 *show ip bgp neighbors*

Данная команда отображает информацию о всех или о выбранном BGP-соседе.

#### Синтаксис

```
show ip bgp <AS> neighbors [ vrf <VRF> ] [ <ADDR> [ routes | advertise-routes ] ]
```

#### Параметры

<AS> – номер автономной системы процесса, принимает значения [1..4294967295];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации;

<ADDR> – IP-адрес соседа, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

routes – при указании команды отображается маршрутная информация, полученная от соседа;

advertise-routes – при указании команды отображается маршрутная информация, объявленная соседу.

#### Необходимый уровень привилегий

1

#### Командный режим

ROOT

#### Пример 1

```
esr# show ip bgp 20 neighbors
BGP neighbor is 10.115.0.1
  BGP state:           Established
  Neighbor address:    10.115.0.1
  Neighbor AS:         20
  Neighbor ID:         115.0.0.1
  Neighbor caps:       refresh restart-aware AS4
  Session:             internal multihop AS4
  Source address:      10.115.0.2
  Hold timer:          137/180
  Keepalive timer:     10/60
```

#### Пример 2

```
esr# show ip bgp 20 neighbors 10.115.0.1 routes
Status codes: u - unicast, b - broadcast, m - multicast a - anycast
               * - valid, > - best
```

Origin codes: i - IGP, e - EGP, ? - incomplete

| Network           | Next Hop   | Metric | LocPrf | Path |
|-------------------|------------|--------|--------|------|
| *> u 10.0.10.0/24 | 10.115.0.1 |        | 100    | i    |
| *> u 0.0.0.0/0    | 10.115.0.1 |        | 100    | i    |
| * u 14.0.10.0/24  | 10.115.0.1 |        | 100    | i    |

### Пример 3

```
esr# show ip bgp 20 neighbors 115.0.0.40 advertise-routes
Status codes: u - unicast, b - broadcast, m - multicast a - anycast
               * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete

Network          Next Hop          Metric  LocPrf  Path
*> u 1.1.1.0/24   115.0.0.1         215     100    20 i
*> u 1.1.0.0/24   115.0.0.1         215     100    20 i
*> u 2.2.2.0/24   115.0.0.1         215     100    20 i
```

## 23.6.23 *show ipv6 bgp neighbors*

Данная команда отображает информацию о всех или о выбранном BGP-соседе.

### Синтаксис

`show ipv6 bgp <AS> [ vrf<VRF> ] neighbors [ <IPV6-ADDR> [ routes | advertise-routes ] ]`

### Параметры

<AS> – номер автономной системы процесса, принимает значения [1..4294967295];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации;

<IPV6-ADDR> IPv6-адрес соседа, задается в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

routes – при указании команды отображается маршрутная информация, полученная от соседа;

advertise-routes – при указании команды отображается маршрутная информация, объявленная соседу.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример 1

```
esr# show ipv6 bgp 20 neighbors
BGP neighbor is 2002::7300:c
  BGP state:      Established
  Neighbor address: 2002::7300:c
  Neighbor AS:    1
  Neighbor ID:    77.0.0.1
  Neighbor caps:   refresh
  Session:        external
  Source address:  2002::7300:1
  Hold timer:     127/154
  Keepalive timer: 1/34
```

```
BGP neighbor is 2002::7300:a
  BGP state:      Established
  Neighbor address: 2002::7300:a
  Neighbor AS:    1000
  Neighbor ID:    10.10.10.10
  Neighbor caps:  refresh AS4
  Session:       external AS4
  Source address: 2002::7300:1
  Hold timer:    157/180
  Keepalive timer: 32/60
```

## Пример 2

```
esr# show ipv6 bgp 20 neighbors 2002::7300:a routes
Status codes: u - unicast, b - broadcast, m - multicast a - anycast
               * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network                Next Hop                Metric  LocPrf  Path
*  u 2002::7300:0/120      2002::7300:a                0       188    1000 e
*> u 220::7300:0/120       2002::7300:a                0       188    1000 e
*> u 22::7300:0/120        2002::7300:a                0       188    1000 e
*> u 2002::40:0/120        2002::7300:a                0       188    1000 e
```

## Пример 3

```
esr# show ipv6 bgp 20 neighbors 2002::7300:a advertise-routes
Status codes: u - unicast, b - broadcast, m - multicast a - anycast
               * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network                Next Hop                Metric  LocPrf  Path
*> u 77::77:0/120          2002::7300:1                100      20     20 i
*> u 33::33:0/120          2002::7300:1                100      20     20 i
*> u 44::44:44/128         2002::7300:1                100      20     20 i
*> u 222::/120             44:44:44::44                0        100    20 1 ?
*> u 40::75:0/120          44:44:44::44                0        100    20 1 ?
```

### 23.6.24 *clear ip bgp*

Данная команда сбрасывает все или определенный BGP-процесс.

#### Синтаксис

```
clear ip bgp [vrf <VRF>] [ <AS> | neighbor <ADDR>]
```

#### Параметры

<AS> – номер автономной системы, принимает значения [1..4294967295];

<ADDR> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации.

#### Необходимый уровень привилегий

10

#### Командный режим

ROOT

## Пример

```
esr# clear ip bgp
esr# clear ip bgp 1000
```

### 23.6.25 *clear ipv6 bgp*

Данная команда сбрасывает все или определенный BGP-процесс.

#### Синтаксис

```
clear ipv6 bgp [vrf <VRF>] [ <AS> | neighbor <IPV6-ADDR>]
```

#### Параметры

<AS> – номер автономной системы, принимает значения [1..4294967295];

<IPV6-ADDR> IPv6-адрес клиента, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации.

#### Необходимый уровень привилегий

10

#### Командный режим

ROOT

## Пример

```
esr# clear ipv6 bgp
esr# clear ipv6 bgp 1000
```

## 23.7 Настройка протокола RIP

### 23.7.1 *router rip*

Данной командой осуществляется переход в режим настройки параметров RIP-процесса.

Использование отрицательной формы команды (no) устанавливает параметры RIP-процесса на значения по умолчанию.

#### Синтаксис

```
[no] router rip
```

#### Параметры

Команда не содержит аргументов.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

## Пример

```
esr(config)# router rip
esr(config-rip)#
```

### **23.7.2**      *enable*

---

Данной командой включается RIP-протокол.

Использование отрицательной формы команды (no) отключает RIP-протокол.

#### **Синтаксис**

[no] enable

#### **Параметры**

Команда не содержит аргументов.

#### **Значение по умолчанию**

Протокол выключен.

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-RIP

## Пример

```
esr(config-rip)# enable
```

### **23.7.3**      *preference*

---

Данная команда определяет приоритетность маршрутов протокола RIP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### **Синтаксис**

preference <VALUE>  
no preference

#### **Параметры**

<VALUE> – приоритетность маршрутов протокола RIP, принимает значения в диапазоне [1..255].

#### **Значение по умолчанию**

100

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-RIP

## Пример

```
esr(config-rip)# preference 30
```

### 23.7.4 authentication algorithm

Данная команда определяет алгоритм аутентификации.

Использование отрицательной формы команды (no) отключает аутентификацию.

#### Синтаксис

```
authentication algorithm <ALGORITHM>  
no authentication algorithm
```

#### Параметры

<ALGORITHM> – алгоритм аутентификации:

- cleartext – пароль, передается открытым текстом;
- md5 – пароль хешируется по алгоритму md5.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-RIP

## Пример

```
esr(config-rip)# authentication algorithm cleartext
```

### 23.7.5 authentication key

Данная команда устанавливает пароль для аутентификации открытым текстом с соседом.

Использование отрицательной формы команды (no) удаляет пароль.

#### Синтаксис

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

#### Параметры

<CLEAR-TEXT> – пароль, задается строкой от 8 до 16 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

#### Командный режим

CONFIG-RIP

## Пример

```
esr(config-rip)# authentication key ascii-text 123456789  
esr(config-rip)# authentication key ascii-text encrypted CDE65039E5591FA3F1
```

---

### **23.7.6**      *authentication key-chain*

---

Данная команда определяет список паролей для аутентификации через алгоритм хеширования md5.

Использование отрицательной формы команды (no) удаляет привязку к списку паролей.

#### **Синтаксис**

authentication key-chain <KEYCHAIN>

no authentication key-chain

#### **Параметры**

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

RIP

#### **Пример**

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

---

### **23.7.7**      *passive-interface*

---

Данная команда выключает анонсирования маршрутов по интерфейсу.

Использование отрицательной формы команды (no) восстанавливает анонсирования маршрутов

#### **Синтаксис**

[no] passive-interface {<IF> | <TUN> }

#### **Параметры**

<IF> – интерфейс, задаётся в виде, описанном в разделе 3.3;

<TUN> – имя туннеля, задаётся в виде, описанном в разделе 3.4.

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-RIP

#### **Пример**

```
esr(config-rip)# passive-interface gigabitethernet 1/0/15
```

---

### **23.7.8**      *timers update*

---

Данной командой устанавливается временной интервал, по истечении которого производится анонсирование.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.



### Синтаксис

timers update <TIME>  
no timers update

### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

### Значение по умолчанию

180 секунд

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-RIP

### Пример

```
esr(config-rip)# timers update 25
```

## 23.7.9 *timers invalid*

Данной командой определяется временной интервал корректности маршрутной записи без обновления. По истечению срока, без получения обновления, маршрут помечается как не доступный.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

timers invalid <TIME>  
no timers invalid

### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

### Значение по умолчанию

180 секунд

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-RIP

### Пример

```
esr(config-rip)# timers invalid 240
```

---

### **23.7.10**      *timers flush*

---

Данной командой устанавливается временной интервал, по истечении которого производится удаление данного маршрута. При установке значения нужно учитывать следующее правило: «timers invalid + 60».

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### **Синтаксис**

timers flush <TIME>

no timers flush

#### **Параметры**

<TIME> – время в секундах, принимает значения [1..65535].

#### **Значение по умолчанию**

240 секунд

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-RIP

#### **Пример**

```
esr(config-rip)# timers flush 300
```

---

### **23.7.11**      *ip rip metric*

---

Данная команда устанавливает величину метрики на интерфейсе.

Использование отрицательной формы команды (no) устанавливает значение метрики по умолчанию.

#### **Синтаксис**

ip rip metric <VALUE>

no ip rip metric

#### **Параметры**

<VALUE> – величина метрики, задаётся в размере [0..32767].

#### **Значение по умолчанию**

5

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-GRE  
CONFIG-IP4IP4

#### Пример

```
esr(config-if-gi)# ip rip metric 11
```

### 23.7.12 *ip rip mode*

Данная команда устанавливает режим анонсирования маршрутов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

ip rip mode <MODE>  
no ip rip mode

#### Параметры

<MODE> – режим анонсирования маршрутов:

multicast – маршруты анонсируются в многоадресном режиме;

broadcast – маршруты анонсируются в широковещательном режиме;

unicast – маршруты анонсируются в unicast-режиме соседям, настроенным с помощью команды **ip rip neighbor <ADDR>**.

#### Значение по умолчанию

multicast

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-GRE  
CONFIG-IP4IP4

#### Пример

```
esr(config-if-gi)# ip rip mode broadcast
```

---

### **23.7.13**      *ip rip neighbor*

---

Данной командой статически задается IP-адрес соседа для установления отношения в unicast-режиме анонсирования маршрутов.

Использование отрицательной формы команды (no) удаляет статически заданный адрес соседа.

#### **Синтаксис**

[no] ip rip neighbor <ADDR>

#### **Параметры**

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-GRE  
CONFIG-IP4IP4

#### **Пример**

```
esr(config-if-gi)# ip rip neighbor 10.100.100.5
```

---

### **23.7.14**      *ip rip summary-address*

---

Данной командой включается суммаризация подсетей.

Использование отрицательной формы команды (no) отключает суммаризацию подсетей.

#### **Синтаксис**

[no] ip rip summary-address <ADDR/LEN>

#### **Параметры**

<ADDR/LEN> – IP-адрес и маска подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

### Пример

```
esr(config-if-gi)# ip rip summary-address 10.200.200.0/24
```

## 23.7.15 *show ip rip*

Данная команда отображает таблицу маршрутизации RIP.

### Синтаксис

show ip rip

### Параметры

Команда не содержит аргументов.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show ip rip Sword# sho ip rip
10.10.0.1/32    via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
10.1.90.0/24   via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
192.168.16.0/24 via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
```

## 23.7.16 *clear ip rip*

Данная команда удаляет содержимое базы маршрутов RIP.

### Синтаксис

clear ip rip

### Параметры

Команда не содержит аргументов.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# clear ip rip
```

---

## **23.8 Настройка протокола OSPF**

---

### **23.8.1      *router ospf***

---

Данной командой добавляется OSPF-процесс в систему и осуществляется переход в режим настройки параметров OSPF-процесса.

Использование отрицательной формы команды (no) удаляет OSPF-процесс из системы.

#### **Синтаксис**

[no] router ospf <ID> [vrf <VRF>]

#### **Параметры**

<ID> – номер автономной системы процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации.

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG

#### **Пример**

```
esr(config)# router ospf 300
esr(config-ospf)#
```

### **23.8.2      *router-id***

---

Данной командой устанавливается идентификатор маршрутизатора.

Использование отрицательной формы команды (no) удаляет идентификатор.

#### **Синтаксис команды**

router-id <ID>

no router-id

#### **Параметры**

<ID> – идентификатор маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-OSPF

#### **Пример**

```
esr(config-ospf)# router-id 1.1.1.1
```

### 23.8.3 *preference*

Данная команда определяет приоритетность маршрутов процесса OSPF.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

preference <VALUE>

no preference

#### Параметры

<VALUE> – приоритетность маршрутов процесса OSPF, принимает значения в диапазоне [1..255].

#### Значение по умолчанию

10

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-OSPF

#### Пример

```
esr(config-ospf)# preference 30
```

### 23.8.4 *compatible rfc1583*

Включается совместимость с RFC 1583.

Использование отрицательной формы команды (no) отключает совместимость с RFC 1583.

#### Синтаксис команды

[no] compatible rfc1583

#### Параметры

Команда не содержит аргументов.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-OSPF

#### Пример

```
esr(config-ospf)# compatible rfc1583
```

### 23.8.5 *area*

Данной командой устанавливается идентификатор области.

Использование отрицательной формы команды (no) удаляет созданную область.

## Синтаксис команды

[no] area <AREA\_ID>

### Параметры

<AREA\_ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPF

### Пример

```
esr(config-ospf)# area 11.11.11.51
```

## 23.8.6 *area-type*

Команда определяет тип области.

Использование отрицательной формы команды (no) устанавливает тип области как стандартный.

### Синтаксис команды

[no] area-type <TYPE> [ no-summary ]

### Параметры

<TYPE> – тип области:

- stub – устанавливает значение stub (тупиковая область);

no-summary – команда в связке с параметром «stub» образует область «totally stubby» (для передачи информации за пределы области используется только маршрут по умолчанию);

- nssa – устанавливает значение nssa (область NSSA);

no-summary – в связке с параметром nssa образует область totally nssa (автоматически генерирует маршрут по умолчанию как межобластной).

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPF-AREA

### Пример

```
esr(config-ospf-area)# area-type stub
```

## 23.8.7 *default-information-originate*

Включается генерация маршрута по умолчанию для NSSA-области и анонсирование его в качестве NSSA-LSA.



Использование отрицательной формы команды (no) отключает генерацию маршрута по умолчанию.

### Синтаксис команды

[no] default-information-originate

### Параметры

Команда не содержит аргументов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPF-AREA

### Пример

```
esr(config-ospf-area)# default-information-originate
```

---

## 23.8.8 *summary-address*

Командой включается суммаризация или скрывание подсетей.

Использование отрицательной формы команды (no) отключает суммаризацию или скрывание подсетей.

### Синтаксис команды

[no] summary-address <ADDR/LEN> { advertise | not-advertise }

### Параметры

<ADDR/LEN> – IP-адрес и маска подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

advertise – при указании команды вместо указанных подсетей будет анонсироваться суммарная подсеть;

not-advertise – при указании команды подсети, входящие в указанную подсеть, анонсироваться не будут.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPF-AREA

### Пример

```
esr(config-ospf-area)# summary-address 192.168.16.0/24
```

---

## 23.8.9 *virtual-link*

Устанавливается виртуальное соединение между основной и удалёнными областями, имеющие между ними несколько областей.

Использование отрицательной формы команды (no) удаляет созданное виртуальное соединение.

### Синтаксис команды

[no] virtual-link <ID>

### Параметры

<ID> – идентификатор маршрутизатора, с которым устанавливается виртуальное соединение, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPF-AREA

### Пример

```
esr(config-ospf-area)# virtual-link 160.0.0.2
```

## 23.8.10 *retransmit-interval*

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор повторно отправит пакет, который не получил подтверждения о получении (например, Database Description пакет или Link State Request пакеты).

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

retransmit-interval <TIME>

no retransmit-interval

### Параметры

<TIME> – время в секундах, принимает значения [2..65535].

### Значение по умолчанию

5 секунд

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPF-VLINK

### Пример

```
esr(config-ospf-vlink)# retransmit-interval 4
```

## 23.8.11 *hello-interval*

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор отправляет следующий hello-пакет.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

#### Синтаксис

```
hello-interval <TIME>
no hello-interval
```

#### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

#### Значение по умолчанию

10 секунд

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-OSPF-VLINK

#### Пример

```
esr(config-ospf-vlink)# hello-interval 8
```

### 23.8.12 *dead-interval*

Данной командой устанавливается интервал времени в секундах, по истечении которого сосед будет считаться "мертвым". Этот интервал должен быть кратным значению «hello-interval». Как правило, «dead-interval» равен 4 интервалам отправки hello-пакетов, то есть 40 секундам.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

#### Синтаксис

```
dead-interval <TIME>
no dead-interval
```

#### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

#### Значение по умолчанию

40 секунд

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-OSPF-VLINK

#### Пример

```
esr(config-ospf-vlink)# dead-interval 60
```

---

### **23.8.13      *enable***

---

Данной командой включается OSPF-процесс, область, виртуальное соединение.

Использование отрицательной формы команды (no) выключает OSPF-процесс, зону, виртуальное соединение.

#### **Синтаксис**

[no] enable

#### **Параметры**

Команда не содержит аргументов.

#### **Значение по умолчанию**

Выключено.

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-OSPF

CONFIG-OSPF-AREA

CONFIG-OSPF-VLINK

#### **Пример 1**

Включение процесса OSPF 300.

```
esr(config-ospf)# enable
```

#### **Пример 2**

Активация области.

```
esr(config-ospf-area)# enable
```

#### **Пример 3**

Активация виртуального соединения.

```
esr(config-ospf-vlink)# enable
```

---

### **23.8.14      *ip ospf instance***

---

Данная команда определяет принадлежность интерфейса к определенному OSPF-процессу.

Использование отрицательной формы команды (no) удаляет принадлежность интерфейса к OSPF-процессу.

#### **Синтаксис**

ip ospf instance <ID>

no ip ospf instance

#### **Параметры**

<ID> – номер процесса, принимает значения [1..65535].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE  
CONFIG-LOOPBACK

## Пример

```
esr(config-ip4ip4)# ip ospf instance 300
```

### **23.8.15**      *ip ospf area*

Данная команда определяет принадлежность интерфейса к определенной области OSPF-процесса.

Использование отрицательной формы команды (no) удаляет принадлежность интерфейса к определенной области OSPF-процесса.

## Синтаксис

ip ospf area <AREA\_ID>  
no ip ospf area

## Параметры

<AREA\_ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE  
CONFIG-LOOPBACK

## Пример

```
esr(config-ip4ip4)# ip ospf area 1.1.1.1
```

---

## **23.8.16**      ***ip ospf***

---

Данной командой включают маршрутизацию по протоколу OSPF на интерфейсе.

Использование отрицательной формы команды (no) отключает маршрутизацию по протоколу OSPF на интерфейсе.

### **Синтаксис**

[no] ip ospf

### **Параметры**

Команда не содержит аргументов.

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

### **Пример**

```
esr(config-if-gi)# ip ospf
```

---

## **23.8.17**      ***ip ospf mtu-ignore***

---

Данной командой включается режим, в котором OSPF-процесс будет игнорировать значение MTU интерфейса во входящих Database Description-пакетах.

Использование отрицательной формы команды (no) отключает режим игнорирования MTU интерфейса.

### **Синтаксис**

[no] ip ospf mtu-ignore

### **Параметры**

Команда не содержит аргументов.

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE  
CONFIG-LOOPBACK

#### Пример

```
esr(config-if-gi)# ip ospf mtu-ignore
```

### 23.8.18 *ip ospf authentication algorithm*

Данная команда определяет алгоритм аутентификации.

Использование отрицательной формы команды (no) отключает аутентификацию.

#### Синтаксис

```
ip ospf authentication algorithm <ALGORITHM>
no ip ospf authentication algorithm
```

#### Параметры

<ALGORITHM> – алгоритм аутентификации:

- cleartext – пароль, передается открытым текстом;
- md5 – пароль хешируется по алгоритму md5.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE  
CONFIG-LOOPBACK

#### Пример

```
esr(config-if-gi)# ip ospf authentication algorithm cleartext
```

### 23.8.19 *ip ospf authentication key*

Данная команда устанавливает пароль для аутентификации с соседом при передаче пароля открытым текстом.

Использование отрицательной формы команды (no) удаляет пароль.

#### Синтаксис

```
ip ospf authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
```

```
no ip ospf authentication key
```

### Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)# ip ospf authentication key ascii-text 123456789
esr(config-if-gi)# ip ospf authentication key ascii-text encrypted
CDE65039E5591FA3F1
```

## 23.8.20 *ip ospf authentication key-chain*

Данная команда определяет список паролей для аутентификации через алгоритм хеширования md5 с соседом.

Использование отрицательной формы команды (no) удаляет привязку к списку паролей.

### Синтаксис

```
ip ospf authentication key-chain <KEYCHAIN>
```

```
no ip ospf authentication key-chain
```

### Параметры

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4



CONFIG-GRE

CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)# ip ospf authentication key-chain lock
```

## 23.8.21 *ip ospf wait-interval*

Данной командой определяется интервал времени в секундах, по истечении которого маршрутизатор выберет DR в сети.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

```
ip ospf wait-interval <TIME>
```

```
no ip ospf wait-interval
```

### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

### Значение по умолчанию

40 секунд.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

LOOPBACK

### Пример

```
esr(config-if-gi)# ip ospf wait-interval 60
```

## 23.8.22 *ip ospf retransmit-interval*

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор повторно отправит пакет, на который не получил подтверждения о получении (например, Database Description пакет или Link State Request пакеты).

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

ip ospf retransmit-interval <TIME>

no ip ospf retransmit-interval

### Параметры

<TIME> – время в секундах, принимает значения [2..65535].

### Значение по умолчанию

5 секунд

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)#ip ospf retransmit-interval 4
```

## 23.8.23 *ip ospf hello-interval*

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор отправляет следующий hello-пакет.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

ip ospf hello-interval <TIME>

no ip ospf hello-interval

### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

### Значение по умолчанию

10 секунд

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE  
CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)# ip ospf hello-interval 8
```

## 23.8.24 *ip ospf dead-interval*

Данной командой устанавливается интервал времени в секундах, по истечении которого сосед будет считаться "мертвым". Этот интервал должен быть кратным значению hello-interval. Как правило, dead-interval равен 4 интервалам отправки hello-пакетов, то есть 40 секундам.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

```
ip dead-interval <TIME>
no ip dead-interval
```

### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

### Значение по умолчанию

40 секунд

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE  
CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)# ip ospf dead-interval 60
```

## 23.8.25 *ip ospf poll-interval*

Данная команда устанавливает интервал времени, в течение которого NBMA-интерфейс ждет, прежде чем отправить HELLO-пакет соседу, даже в случае, если сосед неактивен.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

```
ip poll-interval <TIME>
no ip poll-interval
```

### Параметры

<TIME> – время в секундах, принимает значения [1 .. 65535].

### Значение по умолчанию

120 секунд

### Необходимый уровень привилегий

10

### Командный режим

```
CONFIG-GI
CONFIG-SUBIF
CONFIG-PORT-CHANNEL
CONFIG-BRIDGE
CONFIG-IP4IP4
CONFIG-GRE
CONFIG-LOOPBACK
```

### Пример

```
esr(config-if-gi)# ip ospf poll-interval 60
```

## 23.8.26 *ip ospf neighbor*

Данной командой статически задается IP-адрес соседа для установления отношения в NMBA и P2MP (Point-to-MultiPoint) сетях. Использование отрицательной формы команды (no) удаляет статически заданный адрес соседа.

### Синтаксис

```
[no] ip ospf neighbor <IP> [ eligible ]
```

### Параметры

<IP> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

eligible – опциональный параметр, позволяет устройству участвовать в процессе выбора DR в NBMA-сетях. Приоритет интерфейса должен быть больше нуля, команда изменения приоритета описана в разделе 23.8.28.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE  
CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)# ip ospf neighbor 10.0.0.2
```

## 23.8.27 *ip ospf network*

Данная команда определяет тип сети.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
ip ospf network <TYPE>  
no ip ospf network
```

### Параметры

<TYPE> – тип сети:

broadcast – тип соединения широковещательный;

non-broadcast – тип соединения NBMA;

point-to-multipoint – тип соединения точка-многоточие;

point-to-multipoint non-broadcast – тип соединения NBMA точка-многоточие;

point-to-point – тип соединения точка-точка.

### Значение по умолчанию

broadcast

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE

**Пример**

```
esr(config-if-gi)# ip ospf network point-to-point
```

---

**23.8.28      *ip ospf priority***

---

Данной командой устанавливается приоритет маршрутизатора, который используется для выбора DR и BDR.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

ip ospf priority <VALUE>

no ip ospf priority

**Параметры**

<VALUE> – приоритет интерфейса, принимает значения [1..65535].

**Значение по умолчанию**

120

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

**Пример**

```
esr(config-if-gi)# ip ospf priority 300
```

---

**23.8.29      *ip ospf cost***

---

Данная команда устанавливает величину метрики на интерфейсе или туннеле.

Использование отрицательной формы команды (no) устанавливает значение метрики по умолчанию.

**Синтаксис**

ip ospf cost <VALUE>

no ip ospf cost

## Параметры

<VALUE> – величина метрики, задаётся в размере [0..32767].

## Значение по умолчанию

150

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

## Пример

```
esr(config-if-gi)# ip ospf cost 11
```

## 23.8.30 *router ospf log-neighbor-changes*

Данной командой включается логирование изменений состояния OSPF-соседей.

Использование отрицательной формы команды (no) отключает логирование изменений состояния OSPF-соседей.

## Синтаксис

[no] router ospf log-neighbor-changes

## Параметры

Команда не содержит аргументов.

## Значение по умолчанию

Логирование выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# router ospf log-neighbor-changes
```

## 23.8.31 *clear ip ospf*

Данная команда сбрасывает все или определенный OSPF-процесс.

## Синтаксис

```
clear ip ospf [ <ID> ] [ vrf <VRF> ]
```

## Параметры

<ID> – номер процесса, принимает значения [1..65535].

<VRF> – имя экземпляра VRF, для которого будут сброшены все или определенный OSPF-процесс, задается строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

ROOT

## Пример

```
esr# clear ip ospf
esr# clear ip ospf 1000
```

## 23.8.32 *show ip ospf*

Данная команда отображает таблицу маршрутизации OSPF, если не указан аргумент. При указании процесса выводит информацию о конфигурации интерфейсов по данному процессу.

## Синтаксис

```
show ip ospf [ <ID> ] [ vrf <VRF> ]
```

## Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, для которого будет отображена таблица маршрутизации OSPF, задается строкой до 31 символа.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

Отображение таблицы маршрутизации.

```
esr# show ip ospf
0          2.2.2.0/24          [150/10] dev gil/0/1          [ospf2 19:40:31] (2.2.2.2)
```

## 23.8.33 *show ip ospf interface*

Данная команда отображает информацию об OSPF-интерфейсе.

## Синтаксис

```
show ip ospf [ <ID> [ vrf <VRF> ] ] interface [ <IF> | <TUN> ]
```



## Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе 3.3;

<TUN> – имя туннеля, задаётся в виде, описанном в разделе 3.4;

<VRF> – имя экземпляра VRF для OSPF-процесса, в котором будет отображена информация об OSPF-интерфейсе, задается строкой до 31 символа.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show ip ospf interface gil/0/1
Interface:                gigabitethernet 1/0/1
Internet Address:         25.25.0.1/24
Router ID:                6.0.0.1
Network Type:             broadcast
Area:                    0.0.0.0 (0)
Interface has:            no authentication
Transmit:                 1
State:                   dr
Priority:                 128
Cost:                    10
ECMP weight:              1
Hello timer:              10
Wait timer:              40
Dead timer:              40
Retransmit timer:         5
Designed router (ID):      6.0.0.1
Designed router (IP):      25.25.0.1
Backup designed router (ID): 6.0.0.3
Backup designed router (IP): 25.25.0.3
Neighbor Count:           0
Adjacent neighbor count:   0
```

### 23.8.34 *show ip ospf database*

Данная команда отображает таблицу данных OSPF.

## Синтаксис

show ip ospf <ID> [ vrf <VRF> ] database

## Параметры

<ID> – номер OSPF-процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPF-процесса, в котором будет отображена таблица данных, задается строкой до 31 символа.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show ip ospf 111 database
Global
Type   LS ID           Router           Age   Sequence   Checksum
0005   10.166.11.12     10.1.0.1         1020  80000013   01b7
0005   0.0.0.0           10.166.11.1      245   80000010   aa48
0005   10.62.19.128      10.166.11.1      725   8000000e   6d2b
0005   10.62.20.0        10.166.11.1      731   8000000d   69af
0005   10.62.20.128      10.166.11.1      244   80000010   5e37
0005   10.62.21.128      10.166.11.1      244   80000010   5341
0005   10.166.11.0       10.166.11.1      245   80000010   cc6d
0005   10.166.11.12      10.166.11.1      245   80000010   54d9

Area 0.0.11.1
Type   LS ID           Router           Age   Sequence   Checksum
0001   10.1.0.1         10.1.0.1         1015  80000067   989e
0001   10.166.11.1      10.166.11.1      1021  80000018   8d96
0002   10.166.11.14     10.166.11.1      1021  80000001   68a5
```

### 23.8.35 *show ip ospf neighbors*

Данная команда отображает информацию о всех соседях или соседях определенного OSPF-процесса.

#### Синтаксис

```
show ip ospf [ <ID> [ vrf <VRF> ] ] neighbors
```

#### Параметры

<ID> – номер OSPF-процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPF процесса, в котором будет отображена информация по соседям, задается строкой до 31 символа.

#### Необходимый уровень привилегий

1

#### Командный режим

ROOT

#### Пример

```
esr# show ip ospf neighbors
Router ID      Pri      State      DTime  Interface  Router IP
160.0.0.2      0        full/ptp    00:53  vlink0     160.0.0.2
95.0.0.1       1        full/dr     00:31  gil_15     115.0.0.10
10.100.100.2   128      full/ptp    00:37  gre_25     25.25.0.2
153.0.0.1      1        full/bdr    00:30  pol        1.1.0.2
10.100.100.2   128      2way/other  00:34  gil_14.25  14.25.0.2
24.24.24.24    15       full/bdr    00:32  gil_1      24.0.0.2
```

### 23.8.36 *show ip ospf virtual-links*

Данная команда отображает информацию о виртуальных соединениях.

#### Синтаксис

```
show ip ospf <ID> [ vrf <VRF> ] virtual-links
```

#### Параметры

<ID> – номер OSPF-процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPF процесса, в котором будет отображена информация по виртуальным соединениям, задается строкой до 31 символа.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show ip ospf 10 virtual-links
Virtual Link to router 160.0.0.2 is ptp
Peer IP: 160.0.0.2
Transit area: 1.1.1.1
Interface has no authentication
Timer intervals configured: Hello 10, Dead 60, Retransmit 5, Wait 60
Adjacency State full
```

## 23.9 Настройка протокола OSPFv3

### 23.9.1 *ipv6 router ospf*

Командой добавляется OSPFv3-процесс в систему и осуществляется переход в режим настройки параметров OSPFv3-процесса.

Использование отрицательной формы команды (no) удаляет OSPFv3-процесс из системы.

### Синтаксис

[no] ipv6 router ospf <ID> [vrf <VRF>]

### Параметры

<ID> – номер автономной системы процесса, принимает значения [1..65535].

<VRF> – имя экземпляра VRF, в котором будет работать OSPFv3-процесс, задается строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ipv6 router ospf 300
esr(config-ipv6-ospf)#
```

### 23.9.2 *router-id*

Командой устанавливается идентификатор маршрутизатора.

Использование отрицательной формы команды (no) удаляет идентификатор.

### Синтаксис команды

router-id <ID>

no router-id

### Параметры

<ID> – идентификатор маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPFV3

### Пример

```
esr(config-ipv6-ospf)# router-id 1.1.1.1
```

## 23.9.3 *preference*

Команда определяет приоритетность маршрутов процесса OSPFv3.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

preference <VALUE>

no preference

### Параметры

<VALUE> – приоритетность маршрутов процесса OSPFv3, принимает значения в диапазоне [1..255].

### Значение по умолчанию

10

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPFV3

### Пример

```
esr(config-ipv6-ospf)# preference 30
```

## 23.9.4 *compatible rfc1583*

Командой включается совместимость с RFC 1583.

Использование отрицательной формы команды (no) отключает совместимость с RFC 1583.

### Синтаксис команды

[no] compatible rfc1583

### Параметры

Команда не содержит аргументов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OSPFV3

## Пример

```
esr(config-ipv6-ospf) # compatible rfc1583
```

## 23.9.5 *area*

Командой устанавливается идентификатор области.

Использование отрицательной формы команды (no) удаляет созданную область.

## Синтаксис команды

[no] area <AREA\_ID>

## Параметры

<AREA\_ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OSPFV3

## Пример

```
esr(config-ipv6-ospf) # area 11.11.11.51
```

## 23.9.6 *area-type*

Данная команда определяет тип области.

Использование отрицательной формы команды (no) устанавливает тип области как стандартный.

## Синтаксис команды

[no] area-type <TYPE> [ no-summary ]

## Параметры

<TYPE> – тип области:

- stub – устанавливает значение stub (тупиковая область);

no-summary – команда в связке с параметром «stub» образует область «totally stubby» (для передачи информации за пределы области используется только маршрут по умолчанию);

- nssa – устанавливает значение nssa (область NSSA).

no-summary – команда в связке с параметром «nssa» образует область «totally nssa» (автоматически генерирует маршрут по умолчанию как межобластной).

## Необходимый уровень привилегий

**Командный режим**

CONFIG-OSPFV3-AREA

**Пример**

```
esr(config-ipv6-ospf-area)# area-type stub
```

## 23.9.7 *default-information-originate*

Данной командой включается генерация маршрута по умолчанию для NSSA-области и анонсирование его в качестве NSSA-LSA.

Использование отрицательной формы команды (no) отключает генерацию маршрута по умолчанию.

**Синтаксис команды**

```
[no] default-information-originate
```

**Параметры**

Команда не содержит аргументов.

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-OSPFV3-AREA

**Пример**

```
esr(config-ipv6-ospf-area)# default-information-originate
```

## 23.9.8 *summary-address*

Данной командой включается суммаризация или скрывание подсетей.

Использование отрицательной формы команды (no) отключает суммаризацию или скрывание подсетей.

**Синтаксис команды**

```
[no] summary-address <IPv6-ADDR/LEN> { advertise | not-advertise }
```

**Параметры**

<IPv6-ADDR/LEN> – IPv6-адрес и маска подсети, задаётся в виде X:X:X:X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128];

advertise – при указании команды вместо подсетей, входящих в указанную подсеть, будет анонсироваться суммарная подсеть;

not-advertise – подсети входящие в указанную подсеть анонсироваться не будут.

**Необходимый уровень привилегий**

## Командный режим

CONFIG-OSPFV3-AREA

## Пример

```
esr(config-ipv6-ospf-area)# summary-address 2002:30::0/90
```

### 23.9.9 *virtual-link*

Данной командой устанавливается виртуальное соединение между основной и удаленными областями, имеющие между ними несколько областей.

Использование отрицательной формы команды (no) удаляет созданное виртуальное соединение.

## Синтаксис команды

[no] virtual-link <ID>

## Параметры

<ID> – идентификатор маршрутизатора, с которым устанавливается виртуальное соединение, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OSPFV3-AREA

## Пример

```
esr(config-ipv6-ospf-area)# virtual-link 160.0.0.2
```

### 23.9.10 *retransmit-interval*

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор повторно отправит пакет, который не получил подтверждения о получении (например, Database Description пакет или Link State Request пакеты).

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

## Синтаксис

retransmit-interval <TIME>

no retransmit-interval

## Параметры

<TIME> – время в секундах, принимает значения [2..65535].

## Значение по умолчанию

5 секунд

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OSPFV3-VLINK

## Пример

```
esr(config-ipv6-ospf-vlink)# retransmit-interval 4
```

### 23.9.11 *hello-interval*

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор отправляет следующий hello-пакет.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

## Синтаксис

hello-interval <TIME>

no hello-interval

## Параметры

<TIME> – время в секундах, принимает значения [1..65535].

## Значение по умолчанию

10 секунд

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-OSPFV3-VLINK

## Пример

```
esr(config-ipv6-ospf-vlink)# hello-interval 8
```

### 23.9.12 *dead-interval*

Данной командой устанавливается интервал времени в секундах, по истечении которого сосед будет считаться неактивным. Этот интервал должен быть кратным значению «hello-interval». Как правило, «dead-interval» равен 4 интервалам отправки hello-пакетов, то есть 40 секундам.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

## Синтаксис

dead-interval <TIME>

no dead-interval

## Параметры

<TIME> – время в секундах, принимает значения [1..65535].



### Значение по умолчанию

40 секунд

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPFV3-VLINK

### Пример

```
esr(config-ipv6-ospf-vlink)# dead-interval 60
```

## 23.9.13 *enable*

Данной командой включается OSPFv3-процесс, область, виртуальное соединение.

Использование отрицательной формы команды (no) выключает OSPFv3-процесс, зону, виртуальное соединение.

### Синтаксис

[no] enable

### Параметры

Команда не содержит аргументов.

### Значение по умолчанию

Выключено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-OSPFV3

CONFIG-OSPFV3-AREA

CONFIG-OSPFV3-VLINK

### Пример 1

Включение процесса OSPFv3 300.

```
esr(config-ipv6-ospf)# enable
```

### Пример 2

Активация области.

```
esr(config-ipv6-ospf-area)# enable
```

### Пример 3

Активация виртуального соединения.

```
esr(config-ipv6-ospf-vlink)# enable
```

---

### **23.9.14**      *ipv6 ospf instance*

---

Данная команда определяет принадлежность интерфейса к определенному OSPFv3-процессу.

Использование отрицательной формы команды (no) удаляет принадлежность интерфейса к OSPFv3-процессу.

#### **Синтаксис**

ipv6 ospf instance <ID>

no ipv6 ospf instance

#### **Параметры**

<ID> – номер процесса, принимает значения [1..65535].

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

#### **Пример**

```
esr(config-ip4ip4)# ipv6 ospf instance 300
```

---

### **23.9.15**      *ipv6 ospf area*

---

Данная команда определяет принадлежность интерфейса к определенной области OSPFv3-процесса.

Использование отрицательной формы команды (no) удаляет принадлежность интерфейса к определенной области OSPFv3-процесса.

#### **Синтаксис**

ipv6 ospf area <AREA\_ID>

no ipv6 ospf area

#### **Параметры**

<AREA\_ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

### Пример

```
esr(config-ip4ip4)# ipv6 ospf area 1.1.1.1
```

## 23.9.16 *ipv6 ospf*

Данной командой включают маршрутизацию по протоколу OSPFv3 на интерфейсе.

Использование отрицательной формы команды (no) отключает маршрутизацию по протоколу OSPFv3 на интерфейсе.

### Синтаксис

[no] ipv6 ospf

### Параметры

Команда не содержит аргументов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)# ipv6 ospf
```

## 23.9.17 *ipv6 ospf mtu-ignore*

Данной командой включается режим, в котором OSPFv3-процесс будет игнорировать значение MTU интерфейса во входящих Database Description пакетах.

Использование отрицательной формы команды (no) отключает режим игнорирования MTU интерфейса.

### Синтаксис

[no] ipv6 ospf mtu-ignore

### Параметры

Команда не содержит аргументов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)# ipv6 ospf mtu-ignore
```

## **23.9.18**      *ipv6 ospf wait-interval*

Данной командой определяется интервал времени в секундах, по истечении которого маршрутизатор выберет DR в сети.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

```
ipv6 ospf wait-interval <TIME>  
no ipv6 ospf wait-interval
```

### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

### Значение по умолчанию

40 секунд.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)# ipv6 ospf wait-interval 60
```

## **23.9.19**      *ipv6 ospf retransmit-interval*

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор повторно отправит пакет, на который не получил подтверждения о получении (например, Database Description пакет или Link State Request пакеты).

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

ipv6 ospf retransmit-interval <TIME>

no ipv6 ospf retransmit-interval

### Параметры

<TIME> – время в секундах, принимает значения [2..65535].

### Значение по умолчанию

5 секунд

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)#ipv6 ospf retransmit-interval 4
```

## 23.9.20 *ipv6 ospf hello-interval*

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор отправляет следующий hello-пакет.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

### Синтаксис

ipv6 ospf hello-interval <TIME>

no ipv6 ospf hello-interval

### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

### Значение по умолчанию

10 секунд

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

**Пример**

```
esr(config-if-gi)# ipv6 ospf hello-interval 8
```

### **23.9.21**      *ipv6 ospf dead-interval*

Данной командой устанавливается интервал времени в секундах, по истечении которого сосед будет считаться "мертвым". Этот интервал должен быть кратным значению hello-interval. Как правило, dead-interval равен 4 интервалам отправки hello-пакетов, то есть 40 секундам.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

**Синтаксис**

```
ipv6 dead-interval <TIME>
```

```
no ipv6 dead-interval
```

**Параметры**

<TIME> – время в секундах, принимает значения [1..65535].

**Значение по умолчанию**

40 секунд

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

**Пример**

```
esr(config-if-gi)# ipv6 ospf dead-interval 60
```

### **23.9.22**      *ipv6 ospf poll-interval*

Данная команда устанавливает интервал времени, в течение которого NBMA-интерфейс ждет, прежде чем отправить hello-пакет соседу, даже в случае, если сосед неактивен.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

**Синтаксис**

```
ipv6 poll-interval <TIME>
```

no ipv6 poll-interval

### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

### Значение по умолчанию

120 секунд

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)# ipv6 ospf poll-interval 60
```

## 23.9.23 *ipv6 ospf neighbor*

Данной командой статически задается IPv6-адрес соседа для установления отношения в NMBA и P2MP (Point-to-MultiPoint) сетях. Использование отрицательной формы команды (no) удаляет статически заданный адрес соседа.

### Синтаксис

[no] ipv6 ospf neighbor <IPV6-ADDR> [ eligible ]

### Параметры

<IPV6-ADDR> – IPv6-адрес соседа, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

eligible – опциональный параметр, позволяет устройству участвовать в процессе выбора DR в NBMA-сетях. Приоритет интерфейса должен быть больше нуля, команда изменения приоритета описана в разделе 23.8.28.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

## Пример

```
esr(config-if-gi)# ipv6 ospf neighbor fc00::2
```

### 23.9.24 *ipv6 ospf network*

Данная команда определяет тип сети.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
ipv6 ospf network <TYPE>
```

```
no ipv6 ospf network
```

#### Параметры

<TYPE> – тип сети:

broadcast – тип соединения широковещательный;

non-broadcast – тип соединения NBMA;

point-to-multipoint – тип соединения точка-многоточие;

point-to-multipoint non-broadcast – тип соединения NBMA точка-многоточие;

point-to-point – тип соединения точка-точка.

#### Значение по умолчанию

broadcast

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

## Пример

```
esr(config-if-gi)# ipv6 ospf network point-to-point
```

### 23.9.25 *ipv6 ospf priority*

Данной командой устанавливается приоритет маршрутизатора, который используется для выбора DR и BDR.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
ipv6 ospf priority <VALUE>
```

```
no ipv6 ospf priority
```



### Параметры

<VALUE> – приоритет интерфейса, принимает значения [1..65535].

### Значение по умолчанию

120

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)# ipv6 ospf priority 300
```

## 23.9.26 *ipv6 ospf cost*

Данная команда устанавливает величину метрики на интерфейсе или туннеле.

Использование отрицательной формы команды (no) устанавливает значение метрики по умолчанию.

### Синтаксис

ipv6 ospf cost <VALUE>

no ipv6 ospf cost

### Параметры

<VALUE> – величина метрики, задаётся в размере [0..32767].

### Значение по умолчанию

150

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

### Пример

```
esr(config-if-gi)# ipv6 ospf cost 11
```

### 23.9.27 *clear ipv6 ospf*

Данная команда сбрасывает все или определенный OSPFv3-процесс.

#### Синтаксис

```
clear ipv6 ospf [ <ID> ] [ vrf <VRF> ]
```

#### Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, для которого будут сброшены все или определенный OSPFv3-процесс, задается строкой до 31 символа.

#### Необходимый уровень привилегий

10

#### Командный режим

ROOT

#### Пример

```
esr# clear ipv6 ospf
esr# clear ipv6 ospf 1000
```

### 23.9.28 *show ipv6 ospf*

Данная команда отображает таблицу маршрутизации OSPFv3, если не указан аргумент. При указании процесса выводит информацию о конфигурации интерфейсов по данному процессу.

#### Синтаксис

```
show ipv6 ospf [ <ID> ] [ vrf <VRF> ]
```

#### Параметры

<ID> – номер процесса, принимает значения [1..65535].

<VRF> – имя экземпляра VRF, для которого будет отображена таблица маршрутизации OSPFv3, задается строкой до 31 символа.

#### Необходимый уровень привилегий

1

#### Командный режим

ROOT

#### Пример

Отображение таблицы маршрутизации.

```
esr# show ipv6 ospf
O          fc00::/120          [150/10] dev gi1/0/5          [ospf2 19:39:18] (2.2.2.2)
```

### 23.9.29 *show ipv6 ospf interface*

Данная команда отображает информацию об OSPFv3-интерфейсе.

## Синтаксис

```
show ipv6 ospf [ <ID> [ vrf <VRF> ] ] interface [ <IF> ]
```

## Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, для OSPFv3 процесса в котором будет отображена информация об OSPFv3-интерфейсе, задается строкой до 31 символа;

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе 3.3;

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# Interface:                               gigabitethernet 1/0/4
Internet Address:                             IID 0
Router ID:                                    88.88.88.88
Network Type:                                 broadcast
Area:                                         0.0.0.0 (0)
Transmit:                                     1
State:                                       backup
Priority:                                    128
Cost:                                        10
ECMP weight:                                1
Hello timer:                                10
Wait timer:                                 40
Dead timer:                                 40
Retransmit timer:                           5
Designed router (ID):                        77.0.0.1
Designed router (IP):                        fe80::c602:46ff:feed:0
Backup designed router (ID):                 88.88.88.88
Backup designed router (IP):                 fe80::1:2ff:fe03:463
Neighbor Count:                              1
Adjacent neighbor count:                     1
Adjacent with neighbor:                      77.0.0.1 (dr)
```

## 23.9.30 *show ipv6 ospf database*

Данная команда отображает таблицу данных OSPFv3.

## Синтаксис

```
show ipv6 ospf <ID> [vrf <VRF>] database
```

## Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPFv3 процесса, в котором будет отображена таблица данных, задается строкой до 31 символа

## Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```
esr# show ipv6 ospf 111 database
Global
Type      LS ID          Router           Age  Sequence  Checksum
0005  10.166.11.12    10.1.0.1        1020 80000013   01b7
0005  0.0.0.0         10.166.11.1     245  80000010   aa48
0005  10.62.19.128     10.166.11.1     725  8000000e   6d2b
0005  10.62.20.0       10.166.11.1     731  8000000d   69af
0005  10.62.20.128     10.166.11.1     244  80000010   5e37
0005  10.62.21.128     10.166.11.1     244  80000010   5341
0005  10.166.11.0      10.166.11.1     245  80000010   cc6d
0005  10.166.11.12     10.166.11.1     245  80000010   54d9

Area 0.0.11.1
Type      LS ID          Router           Age  Sequence  Checksum
0001  10.1.0.1        10.1.0.1        1015 80000067   989e
0001  10.166.11.1     10.166.11.1     1021 80000018   8d96
0002  10.166.11.14    10.166.11.1     1021 80000001   68a5
```

### 23.9.31 *show ipv6 ospf neighbors*

Данная команда отображает информацию о всех соседях или соседях определенного OSPFv3-процесса.

#### Синтаксис

```
show ipv6 ospf [<ID> [vrf <VRF>]] neighbors
```

#### Параметры

<ID> – номер процесса, принимает значения [1..65535], опциональный параметр.

<VRF> – имя экземпляра VRF для OSPFv3 процесса, в котором будет отображена информация по соседям, задается строкой до 31 символа.

#### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```
esr# show ipv6 ospf neighbor
Router ID      Pri    State      DTime    Interface  Router IP
77.0.0.1       1      full/dr    00:32    gil_4      fe80::c602:46ff:feed:0
33.33.33.33    128    full/bdr   00:35    gil_1      fe80::20:3ff:fea0:498
```

### 23.9.32 *show ipv6 ospf virtual-links*

Данная команда отображает информацию о виртуальных соединениях.

#### Синтаксис

```
show ipv6 ospf <ID> [vrf <VRF>] virtual-links
```

## Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPFv3 процесса, в котором будет отображена информация по виртуальным соединениям, задается строкой до 31 символа.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show ipv6 ospf 10 virtual-links
Virtual Link to router 160.0.0.2 is ptp
Peer IP: fe80::20:3ff:fea0:498
Transit area: 1.1.1.1
Interface has no authentication
Timer intervals configured: Hello 10, Dead 60, Retransmit 5, Wait 60
Adjacency State full
```

## 24 РЕЗЕРВИРОВАНИЕ

### 24.1 Управление VRRP

---

#### 24.1.1 *vrrp*

---

Данная команда включает VRRP-процесс на IP-интерфейсе.

Использование отрицательной формы команды (no) выключает VRRP-процесс.

##### Синтаксис

[no] vrrp

##### Параметры

Команда не содержит аргументов.

##### Необходимый уровень привилегий

10

##### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

##### Пример

```
esr(config-if-gi)# vrrp
```

#### 24.1.2 *vrrp ip*

---

Данной командой устанавливается виртуальный IP-адрес VRRP-маршрутизатора.

Использование отрицательной формы команды (no) удаляет виртуальный IP-адрес маршрутизатора.

##### Синтаксис

vrrp ip <ADDR>

no vrrp ip

##### Параметры

<ADDR> – виртуальный IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно указать несколько IP-адресов перечислением через запятую. Может быть назначено до 8 IP-адресов на интерфейс.

##### Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE

## Пример

```
esr(config-bridge)# vrrp ip 115.0.0.1
```

### 24.1.3 *vrrp id*

Данной командой устанавливается идентификатор VRRP-маршрутизатора.

Использование отрицательной формы команды (no) удаляет идентификатора виртуального маршрутизатора.

## Синтаксис

vrrp id <VRID>  
no vrrp id

## Параметры

<VRID> – идентификатора VRRP-маршрутизатора, принимает значения [1..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE

## Пример

```
esr(config-if-gi)# vrrp id 125
```

### **24.1.4**      ***vrrp priority***

---

Данной командой устанавливается приоритет VRRP-маршрутизатора.

Использование отрицательной формы команды (no) устанавливает значение приоритета по умолчанию.

#### **Синтаксис**

vrrp priority <PR>

no vrrp priority

#### **Параметры**

<PR> – приоритет VRRP-маршрутизатора, принимает значения [1..254].

#### **Значение по умолчанию**

100

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

#### **Пример**

```
esr(config-if-gi)# vrrp priority 189
```

### **24.1.5**      ***vrrp group***

---

Данной командой устанавливается принадлежность VRRP-маршрутизатора к группе. Группа предоставляет возможность синхронизировать несколько VRRP-процессов, так если в одном из процессов произойдет смена мастера, то в другом процессе также произведется смена ролей.

Использование отрицательной формы команды (no) удаляет VRRP-маршрутизатор из группы.

#### **Синтаксис**

vrrp group <GRID>

no vrrp group

#### **Параметры**

<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].

#### **Необходимый уровень привилегий**



10

## Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE

## Пример

```
esr(config-if-gi)# vrrp group 10
```

### 24.1.6 *vrrp source-ip*

Данной командой устанавливается IP-адрес, который будет использоваться в качестве IP-адреса отправителя для VRRP-сообщений.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес.

## Синтаксис

vrrp source-ip <IP>  
no vrrp source-ip

## Параметры

<IP> – IP-адрес отправителя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE

## Пример

```
esr(config-bridge)# vrrp source-ip 115.0.0.10
```

### 24.1.7 *vrrp timers advertise*

Данная команда определяет интервал между отправкой VRRP-сообщений.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

## Синтаксис

vrrp timers advertise <TIME>

no vrrp timers advertise

### Параметры

<TIME> – время в секундах, принимает значения [1..40].

### Значение по умолчанию

1 секунда.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

### Пример

```
esr(config-gre)# vrrp timers advertise 4
```

## 24.1.8 *vrrp timers garp delay*

Данная команда определяет интервал, по истечении которого происходит отправка Gratuitous ARP сообщения(ий) при переходе маршрутизатора в состояние Master.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

### Синтаксис

vrrp timers garp delay <TIME>

no vrrp timers garp delay

### Параметры

<TIME> – время в секундах, принимает значения [1..60].

### Значение по умолчанию

5 секунд.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE

### Пример

```
esr(config-gre)# vrrp timers garp delay 4
```

## 24.1.9 *vrrp timers garp repeat*

Данная команда определяет количество Gratuitous ARP сообщений, которые будут отправлены при переходе маршрутизатора в состояние Master.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

### Синтаксис

```
vrrp timers garp repeat <COUNT>  
no vrrp timers garp repeat
```

### Параметры

<COUNT> – количество сообщений, принимает значения [1..60].

### Значение по умолчанию

5

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE

### Пример

```
esr(config-gre)# vrrp timers garp repeat 10
```

## 24.1.10 *vrrp timers garp refresh*

Данная команда определяет интервал, по истечении которого будет происходить периодическая отправка Gratuitous ARP сообщения(ий) пока маршрутизатор находится в состоянии Master.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

### Синтаксис

```
vrrp timers garp refresh <TIME>
```

no vrrp timers garp refresh

### Параметры

<TIME> – время в секундах, принимает значения [1..65535].

### Значение по умолчанию

Периодическая отправка отключена.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

### Пример

```
esr(config-gre)# vrrp timers garp refresh 4
```

## 24.1.11 *vrrp timers garp refresh-repeat*

Данная команда определяет количество Gratuituous ARP сообщений, которые будут отправляться с периодом **garp refresh** пока маршрутизатор находится в состоянии Master.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

### Синтаксис

vrrp timers garp refresh-repeat <COUNT>

no vrrp timers garp refresh-repeat

### Параметры

<COUNT> – количество сообщений, принимает значения [1..60].

### Значение по умолчанию

1

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

## Пример

```
esr(config-gre)# vrrp timers garp refresh-repeat 10
```

### 24.1.12 *vrrp preempt disable*

---

Данной командой определяется, будет ли Backup-маршрутизатор с более высоким приоритетом пытаться перехватить на себя роль Master у текущего Master-маршрутизатора с более низким приоритетом.

Исключением является маршрутизатор, у которого виртуальный IP-адрес совпадает с IP-адресом на интерфейсе, он всегда будет перехватывать на себя роль Master вне зависимости от данной настройки.

Использование отрицательной формы команды (no) восстанавливает настройки по умолчанию.

## Синтаксис

[no] vrrp preempt disable

## Параметры

Команда не содержит аргументов.

## Значение по умолчанию

Переключение включено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI  
 CONFIG-SUBIF  
 CONFIG-PORT-CHANNEL  
 CONFIG-BRIDGE  
 CONFIG-IP4IP4  
 CONFIG-GRE

## Пример

```
esr(config-if-gi)# vrrp preempt disable
```

### 24.1.13 *vrrp preempt delay*

---

Данной командой задается временной интервал, по истечении которого Backup-маршрутизатор с более высоким приоритетом будет пытаться перехватить на себя роль Master у текущего Master-маршрутизатора с более низким приоритетом.

Использование отрицательной формы команды (no) восстанавливает настройки по умолчанию.

## Синтаксис

[no] vrrp preemption delay <TIME>

no vrrp preemption delay

### Параметры

<TIME> – время ожидания, определяется в секундах [1..1000].

### Значение по умолчанию

0

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

### Пример

```
esr(config-if-gi)# vrrp preemption delay 100
```

## **24.1.14**      ***vrrp authentication key***

Данная команда устанавливает пароль для аутентификации с соседом.

Использование отрицательной формы команды (no) удаляет пароль.

### Синтаксис

vrrp authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }

no vrrp authentication key

### Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

### Пример

```
esr(config-if-gi)# vrrp authentication key ascii-text 123456789
esr(config-if-gi)# vrrp authentication key ascii-text encrypted CDE65039E5591FA3F1
```

## 24.1.15 *vrrp authentication algorithm*

---

Данная команда определяет алгоритм аутентификации.

Использование отрицательной формы команды (no) отключает аутентификацию.

### Синтаксис

vrrp authentication algorithm <ALGORITHM>

no vrrp authentication algorithm

### Параметры

<ALGORITHM> – алгоритм аутентификации:

- cleartext – пароль, передается открытым текстом;
- md5 – пароль хешируется по алгоритму md5.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

### Пример

```
esr(config-gre)# vrrp authentication algorithm cleartext
```

## 24.1.16 *vrrp version*

---

Данной командой задаётся версия VRRP-протокола.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

vrrp version <VERSION>

no vrrp version

### Параметры

<VERSION> – версия VRRP-протокола: 2, 3.

### Значение по умолчанию

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

## Пример

```
esr(config-gre)# vrrp version 3
```

### 24.1.17 *show vrrp*

Данная команда выводит информации о протоколе VRRP.

## Синтаксис

```
show vrrp [ <ID> ]
```

## Параметры

<ID> – номер процесса, принимает значения [1..255].

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример 1

| Virtual router | Virtual IP | Priority | Preemption | State  |
|----------------|------------|----------|------------|--------|
| 4              | 4.4.4.1    | 100      | Enabled    | Master |

## Пример 2

```
esr# show vrrp 4
Interface                bridge 50
State:                   Master
Virtual IP address:      4.4.4.1
Source IP address:       4.4.4.4
Virtual MAC address:     00:00:5e:00:01:04
Advertisement interval:  1
Preemption:              Enabled
Priority:                 100
Synchronization group ID: --
```



---

## 24.2 Управление IPv6 VRRP

---

### 24.2.1 *ipv6 vrrp*

---

Данная команда включает VRRP-процесс на IPv6-интерфейсе.

Использование отрицательной формы команды (no) выключает VRRP-процесс.

#### Синтаксис

[no] ipv6 vrrp

#### Параметры

Команда не содержит аргументов.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

#### Пример

```
esr(config-if-gi)# ipv6 vrrp
```

### 24.2.2 *ipv6 vrrp ip*

---

Данной командой устанавливается виртуальный IPv6-адрес VRRP-маршрутизатора.

Использование отрицательной формы команды (no) удаляет виртуальный IPv6-адрес маршрутизатора.

#### Синтаксис

ipv6 vrrp ip <IPV6-ADDR>

no ipv6 vrrp ip

#### Параметры

<IPV6-ADDR> – виртуальный IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Можно указать несколько IPv6-адресов перечислением через запятую. Может быть назначено до 8 IPv6-адресов на интерфейс.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-bridge)# ipv6 vrrp ip fc00::1
```

## **24.2.3**      *ipv6 vrrp id*

Данной командой устанавливается идентификатор VRRP-маршрутизатора.

Использование отрицательной формы команды (no) удаляет идентификатора виртуального маршрутизатора.

### Синтаксис

ipv6 vrrp id <VRID>

no ipv6 vrrp id

### Параметры

<VRID> – идентификатора VRRP-маршрутизатора, принимает значения [1..255].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# ipv6 vrrp id 125
```

## **24.2.4**      *ipv6 vrrp priority*

Данной командой устанавливается приоритет VRRP-маршрутизатора.

Использование отрицательной формы команды (no) устанавливает значение приоритета по умолчанию.

### Синтаксис

ipv6 vrrp priority <PR>

no ipv6 vrrp priority

### Параметры

<PR> – приоритет VRRP-маршрутизатора, принимает значения [1..254].

### Значение по умолчанию

100

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ipv6 vrrp priority 189
```

## 24.2.5 *ipv6 vrrp group*

Данной командой устанавливается принадлежность VRRP-маршрутизатора к группе. Группа предоставляет возможность синхронизировать несколько VRRP-процессов, так если в одном из процессов произойдет смена мастера, то в другом процессе также произведется смена ролей.

Использование отрицательной формы команды (no) удаляет VRRP-маршрутизатор из группы.

## Синтаксис

vrrp group <GRID>

no vrrp group

## Параметры

<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ipv6 vrrp group 10
```

## 24.2.6 *ipv6 vrrp source-ip*

Данной командой устанавливается IPv6-адрес, который будет использоваться в качестве IPv6-адреса отправителя для VRRP-сообщений.

Использование отрицательной формы команды (no) удаляет указанный IPv6-адрес.

## Синтаксис

ipv6 vrrp source-ip <IPV6-ADDR>

no ipv6 vrrp source-ip

### Параметры

<IPv6-ADDR> – IPv6-адрес отправителя, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-bridge)# ipv6 vrrp source-ip fc00::2
```

## 24.2.7 *ipv6 vrrp timers advertise*

Данная команда определяет интервал между отправкой VRRP-сообщений.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

### Синтаксис

vrrp timers advertise <TIME>

no vrrp timers advertise

### Параметры

<TIME> – время в секундах, принимает значения [1..40].

### Значение по умолчанию

1 секунда.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# vrrp timers advertise 4
```

## 24.2.8 *ipv6 vrrp preempt disable*

Данной командой определяется, будет ли Backup-маршрутизатор с более высоким приоритетом пытаться перехватить на себя роль Master у текущего Master-маршрутизатора с более низким приоритетом.

Исключением является маршрутизатор, у которого виртуальный IPv6-адрес совпадает с IPv6-адресом на интерфейсе, он всегда будет перехватывать на себя роль Master вне зависимости от данной настройки.

Использование отрицательной формы команды (no) восстанавливает настройки по умолчанию.

### Синтаксис

```
[no] ipv6 vrrp preempt disable
```

### Параметры

Команда не содержит аргументов.

### Значение по умолчанию

Переключение включено.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# ipv6 vrrp preempt disable
```

## 24.2.9 *ipv6 vrrp preempt delay*

Данной командой задается временной интервал, по истечении которого Backup-маршрутизатор с более высоким приоритетом будет пытаться перехватить на себя роль Master у текущего Master-маршрутизатора с более низким приоритетом.

Использование отрицательной формы команды (no) восстанавливает настройки по умолчанию.

### Синтаксис

```
ipv6 vrrp preemption delay <TIME>
```

```
no ipv6 vrrp preemption delay
```

### Параметры

<TIME> – время ожидания, определяется в секундах [1..1000].

### Значение по умолчанию

0

### Необходимый уровень привилегий

**Командный режим**

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

**Пример**

```
esr(config-if-gi)# ipv6 vrrp preemption delay 100
```

### **24.2.10**      *ipv6 vrrp authentication key*

Данная команда устанавливает пароль для аутентификации с соседом.

Использование отрицательной формы команды (no) удаляет пароль.

**Синтаксис**

ipv6 vrrp authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }

no ipv6 vrrp authentication key

**Параметры**

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

**Пример**

```
esr(config-if-gi)# ipv6 vrrp authentication key ascii-text 123456789
esr(config-if-gi)# ipv6 vrrp authentication key ascii-text encrypted
CDE65039E5591FA3F1
```

### **24.2.11**      *ipv6 vrrp authentication algorithm*

Данная команда определяет алгоритм аутентификации.

Использование отрицательной формы команды (no) отключает аутентификацию.

**Синтаксис**

ipv6 vrrp authentication algorithm <ALGORITHM>

no ipv6 vrrp authentication algorithm

## Параметры

<ALGORITHM> – алгоритм аутентификации:

- cleartext – пароль, передается открытым текстом;
- md5 – пароль хешируется по алгоритму md5.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ipv6 vrrp authentication algorithm cleartext
```

## 24.2.12 *ipv6 vrrp version*

Данной командой задаётся версия VRRP-протокола.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

ipv6 vrrp version <VERSION>

no ipv6 vrrp version

## Параметры

<VERSION> – версия VRRP-протокола: 2, 3.

## Значение по умолчанию

3

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ipv6 vrrp version 2
```

## 24.2.13 *show ipv6 vrrp*

Данная команда выводит информации о протоколе VRRP.

### Синтаксис

```
show ipv6 vrrp [ <ID> ]
```

### Параметры

<ID> – номер процесса, принимает значения [1..255].

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример 1

```
esr# show ipv6 vrrp
Virtual router  Virtual IP                Priority  Preemption  State
-----
2              fc00::3                100      Disabled    Master
```

### Пример 2

```
esr# show ipv6 vrrp 2
Interface          gil/0/1
State:             Master
Source IP address: fe80::aaf9:4bff:feaa:3a1
Virtual IP address: fc00::3
Virtual MAC address: 00:00:5e:00:01:02
Advertisement interval: 1
Preemption:        Disabled
Priority:           100
Synchronization group ID: --
```

## 24.3 Настройка резервирования

### 24.3.1 *Настройка резервирования DHCP*

#### 24.3.1.1 *ip dhcp-server failover*

Данной командой включается резервирование DHCP-сервера. Использование отрицательной формы команды (no) выключает резервирование DHCP сервера.

### Синтаксис

```
[no] ip dhcp-server failover
```

### Параметры

Команда не содержит аргументов.

### Значение по умолчанию

Выключено.



## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ip dhcp-server failover
```

### 24.3.1.2 *ip dhcp-server failover local-address*

Данной командой устанавливается IP-адрес, порт (TCP 647), на котором слушает DHCP-сервер в ожидании Failover-сообщений при работе в режиме резервирования. Использование отрицательной формы команды (no) удаляет установленный IP-адрес.

## Синтаксис

ip dhcp-server failover local-address <ADDR>

no ip dhcp-server failover local-address

## Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ip dhcp-server failover local-address 192.168.1.1
```

### 24.3.1.3 *ip dhcp-server failover remote-address*

Данной командой устанавливается IP-адрес резервного DHCP-сервера, на который отправляются Failover-сообщения при работе в режиме резервирования.

Использование отрицательной формы команды (no) удаляет IP-адрес резервного DHCP-сервера.

## Синтаксис

ip dhcp-server failover remote-address <ADDR>

no ip dhcp-server failover remote-address

## Параметры

<ADDR> – IP-адрес резервного DHCP-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ip dhcp-server failover remote-address 192.168.1.2
```

### 24.3.1.4 ip dhcp-server failover role

Данной командой определяется роль DHCP-сервера при работе в режиме резервирования.

Использование отрицательной формы команды (no) удаляет установленную роль DHCP-сервера при работе в режиме резервирования.

#### Синтаксис

```
ip dhcp-server failover role <ROLE>
```

```
no ip dhcp-server failover role
```

#### Параметры

<ROLE> – роль DHCP-сервера при работе в режиме резервирования:

primary – режим активного DHCP-сервера;

secondary – режим резервного DHCP-сервера.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

## Пример

```
esr(config)# ip dhcp-server failover role primary
```

### 24.3.1.5 show ip dhcp server failover

Данная команда позволяет посмотреть состояние резервирования DHCP-сервера.

#### Синтаксис

```
show ip dhcp server failover
```

#### Параметры

Команда не содержит аргументов

#### Необходимый уровень привилегий

1

#### Командный режим

ROOT

## Пример

```
esr# show ip dhcp server failover
Status:                               Disabled
```

---

## 24.3.2 *Настройка резервирования Firewall*

---

### 24.3.2.1 *ip firewall failover*

Данной командой включается резервирование сессий Firewall.

Использование отрицательной формы команды (no) выключает резервирование сессий Firewall.

#### **Синтаксис**

[no] ip firewall failover

#### **Параметры**

Команда не содержит аргументов.

#### **Значение по умолчанию**

Выключено.

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG

#### **Пример**

```
esr(config)# ip firewall failover
```

### 24.3.2.2 *ip firewall failover source-address*

Данной командой устанавливается IP-адрес сетевого интерфейса, с которого будут отправляться сообщения при работе Firewall в режиме резервирования сессий.

Использование отрицательной формы команды (no) удаляет IP-адрес исходящего интерфейса.

#### **Синтаксис**

ip firewall failover source-address <ADDR>

no ip firewall failover source-address

#### **Параметры**

<ADDR> – IP-адрес сетевого интерфейса, с которого будут отправляться сообщения, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG

#### **Пример**

```
esr(config)# ip firewall failover source-address 192.168.1.1
```

### 24.3.2.3 *ip firewall failover destination-address*

Данной командой устанавливается IP-адрес соседа при работе резервирования сессий Firewall в unicast-режиме.

Использование отрицательной формы команды (no) удаляет IP-адрес соседа.

#### **Синтаксис**

ip firewall failover destination-address <ADDR>

no ip firewall failover destination-address

#### **Параметры**

<ADDR> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG

#### **Пример**

```
esr(config)# ip firewall failover destination-address 192.168.1.2
```

### 24.3.2.4 *ip firewall failover port*

Данной командой устанавливается номер UDP-порта службы резервирования сессий Firewall, через который происходит обмен информацией при работе в unicast-режиме.

Использование отрицательной формы команды (no) удаляет номер порта службы резервирования сессий Firewall.

#### **Синтаксис**

ip firewall failover port <PORT>

no ip firewall failover port

#### **Параметры**

<PORT> – номер порта службы резервирования сессий Firewall, указывается в диапазоне [1..65535].

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG

#### **Пример**

```
esr(config)# ip firewall failover port 3333
```

### 24.3.2.5 *ip firewall failover sync-type*

Данной командой определяется режим обмена информацией между основным и резервным маршрутизаторами.

Использование отрицательной формы команды (no) удаляет режим работы резервирования Firewall.

#### Синтаксис

ip firewall failover sync-type <MODE>

no ip firewall failover sync-type

#### Параметры

<MODE> – режим обмена информацией:

- unicast – режим unicast;
- muticast – режим multicast.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

```
esr(config)# ip firewall failover sync-type multicast
```

#### 24.3.2.6 ip firewall failover multicast-address

Данной командой устанавливается многоадресный IP-адрес, который будет использоваться для обмена информации при работе резервирования сессий Firewall в multicast-режиме.

Использование отрицательной формы команды (no) удаляет многоадресный IP-адрес.

#### Синтаксис

ip firewall failover multicast-address <ADDR>

no ip firewall failover multicast-address

#### Параметры

<ADDR> – многоадресный IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

```
esr(config)# ip firewall failover multicast-address 238.0.0.10
```

#### 24.3.2.7 ip firewall failover multicast-group

Данной командой устанавливается идентификатор multicast-группы для обмена информацией при работе резервирования сессий Firewall в multicast-режиме.

Использование отрицательной формы команды (no) удаляет идентификатор группы.

### Синтаксис

```
ip firewall failover multicast-group <GROUP>  
no ip firewall failover multicast-group
```

### Параметры

<GROUP> – multicast-группа, указывается в диапазоне [1000..9999].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ip firewall multicast-group 1028
```

#### 24.3.2.8 ip firewall failover vrrp-group

Данной командой определяется VRRP-группа, на основе которой определяется состояние (основной/резервный) маршрутизатора при резервировании сессий Firewall.

Использование отрицательной формы команды (no) удаляет идентификатор VRRP группы.

### Синтаксис

```
ip firewall failover vrrp-group <GRID>  
no ip firewall failover vrrp-group
```

### Параметры

<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ip firewall failover vrrp-group 10
```

#### 24.3.2.9 show ip firewall failover

Данная команда позволяет посмотреть состояние резервирования сессий Firewall.

### Синтаксис

```
show ip firewall server failover
```

### Параметры

Команда не содержит аргументов

### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```
esr# show ip firewall failover
Communication interface:      br6
Status:                      Running
Bytes sent:                   6407688
Bytes received:               6355040
Packets sent:                 430149
Packets received:             429844
Send errors:                  0
Receive errors:               0
```

### 24.3.3 *show high-availability state*

Данная команда позволяет посмотреть общее состояние систем резервирования и роль устройства.

#### Синтаксис

show high-availability state

#### Параметры

Команда не содержит аргументов

#### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```
esr# show high-availability state
VRRP role:                  Master
AP Tunnels:
  State:                    successful synchronization
  Last synchronization:    17:22:11 08.06.2015
DHCP server:
  State:                    successful synchronization
  Last state change:       17:49:42 03.06.2015
Firewall sessions:
  State:                    successful synchronization
  Last synchronization:    17:22:18 08.06.2015
```

## 24.4 Управление Dual-Homing<sup>1</sup>

---

### 24.4.1 *backup-interface preemption*

---

Данной командой указывается, что необходимо осуществить переключение на основной интерфейс при восстановлении связи. Если настроено восстановление основного интерфейса при активном резервном, то тогда при поднятии линка на основном интерфейсе трафик будет переключен на него.

Использование отрицательной формы команды (no) восстанавливает настройку по умолчанию.

#### Синтаксис

[no] backup-interface preemption

#### Параметры

Команда не содержит аргументов.

#### Значение по умолчанию

Переключение отключено.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

```
esr(config)# backup-interface preemption
```

### 24.4.2 *backup-interface mac-duplicate*

---

Данной командой указывается количество копий пакетов с одним и тем же MAC-адресом, которые будут отправлены в активный интерфейс при переключении.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

#### Синтаксис

backup-interface mac-duplicate <COUNT>

no backup-interface mac-duplicate

#### Параметры

<COUNT> – количество копий пакетов, принимает значение [1..4].

#### Значение по умолчанию

1 пакет

#### Необходимый уровень привилегий

10

#### Командный режим

---

<sup>1</sup> В текущей версии ПО данный функционал поддерживается только на маршрутизаторе ESR-1000



## CONFIG

### Пример

```
esr(config)# backup-interface mac-duplicate 4
```

### 24.4.3 *backup-interface mac-per-second*

Данной командой указывается количество пакетов в секунду, которое будет отправлено в активный интерфейс при переключении.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию (400 пакетов).

#### Синтаксис

```
backup-interface mac-per-second <COUNT>
```

```
no backup-interface mac-per-second
```

#### Параметры

<COUNT> – количество MAC-адресов в секунду, принимает значение \*50..400].

#### Значение по умолчанию

400 пакетов.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

### Пример

```
esr(config)# backup-interface mac-per-second 200
```

### 24.4.4 *backup interface*

Данной командой указывается резервный интерфейс, на который будет происходить переключение при потере связи на основном. Включение резервирования возможно только на тех интерфейсах, на которых отключен протокол Spanning Tree и включен VLAN Ingress Filtering.

Использование отрицательной формы команды (no) удаляет настройку с интерфейса.

#### Синтаксис

```
backup interface <IF> vlan <VID>
```

```
no backup interface
```

#### Параметры

<IF> – интерфейс, задаётся в виде, описанном в разделе 3.3;

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094]. Можно также задать диапазоном через «-» или перечислением через «,».

#### Необходимый уровень привилегий

10

#### Командный режим

## Пример

```
esr(config-if-gi)# backup interface gigabitethernet 1/0/15 vlan 10-200
```

### 24.4.5 *show interfaces backup*

Данная команды выводит информацию о состоянии основного и резервного интерфейса.

## Синтаксис

```
show interfaces backup
```

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show interfaces backup
Backup Interface Options:
Preemption is disabled.
MAC recovery packets rate 400 pps.
Recovery packets repeats count 1.
```

| VID | Master Interface      | Backup Interface      | State                 |
|-----|-----------------------|-----------------------|-----------------------|
| 10  | gigabitethernet 1/0/3 | gigabitethernet 1/0/4 | Master Up/Backup Down |
| 11  | gigabitethernet 1/0/3 | gigabitethernet 1/0/4 | Master Up/Backup Down |
| 12  | gigabitethernet 1/0/3 | gigabitethernet 1/0/4 | Master Up/Backup Down |

## 24.5 Настройка MultiWAN

### 24.5.1 *wan load-balance rule*

Данной командой создается правило WAN и осуществляется переход в режим настройки параметров правила.

Использование отрицательной формы команды (no) удаляет созданное WAN-правило.

## Синтаксис

```
wan load-balance rule <ID>
```

```
no wan load-balance rule { <ID> | all }
```

## Параметры

<ID> – идентификатор создаваемого правила, принимает значения [1..50]. Значение «all» используется при удалении всех правил WAN.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# wan load-balance rule 1
```

### 24.5.2 *ipv6 wan load-balance rule*

Данной командой создается правило WAN и осуществляется переход в режим настройки параметров правила для протокола IPv6.

Использование отрицательной формы команды (no) удаляет созданное WAN-правило.

## Синтаксис

```
ipv6 wan load-balance rule <ID>
no ipv6 wan load-balance rule { <ID> | all }
```

## Параметры

<ID> – идентификатор создаваемого правила, принимает значения [1..50]. Значение «all» используется при удалении всех правил WAN для протокола ipv6.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ipv6 wan load-balance rule 1
```

### 24.5.3 *wan load-balance target-list*

Данной командой создается список IP-адресов для проверки целостности соединения и осуществляется переход в режим настройки параметров списка, а также производится привязка списка на сетевом интерфейсе.

Использование отрицательной формы команды (no) удаляет созданный список.

## Синтаксис

```
wan load-balance target-list <NAME>
no wan load-balance target-list { <NAME> | all }
no wan load-balance target-list (удаление привязки списка в режиме конфигурирования сетевых интерфейсов)
```

## Параметры

<NAME> – название списка, задается строкой до 31 символа. Значение «all» используется при удалении всех списков IP-адресов для проверки целостности соединения.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE

### Пример

```
esr(config)# wan load-balance target-list ten1
```

## 24.5.4 *ipv6 wan load-balance target-list*

Данной командой создается список IPv6-адресов для проверки целостности соединения и осуществляется переход в режим настройки параметров списка, а также производится привязка списка на сетевом интерфейсе.

Использование отрицательной формы команды (no) удаляет созданный список.

### Синтаксис

ipv6 wan load-balance target-list <NAME>  
no ipv6 wan load-balance target-list { <NAME> | all }  
no ipv6 wan load-balance target-list (удаление привязки списка в режиме конфигурирования сетевых интерфейсов)

### Параметры

<NAME> – название списка, задается строкой до 31 символа. Значение «all» используется при удалении всех списков IPv6-адресов для проверки целостности соединения.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG  
CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE

### Пример

```
esr(config)# ipv6 wan load-balance target-list ten1
```

## 24.5.5 *enable*

Данной командой включается правило wan, проверка цели.

Использование отрицательной формы команды (no) выключает правило WAN, проверку цели.

## Синтаксис

[no] enable

## Параметры

Команда не содержит аргументов.

## Значение по умолчанию

Выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WAN-RULE

CONFIG-WAN-TARGET

CONFIG-IPV6-WAN-RULE

CONFIG-IPV6-WAN-TARGET

## Пример

Проверка цели:

```
esr(config-ipv6-wan-rule)# enable
```

## 24.5.6 *outbound*

Данной командой определяются интерфейсы или туннели, которые являются шлюзами в маршруте, создаваемом службой MultiWAN. Количество шлюзов в маршруте зависит от режима работы MultiWAN:

при балансировке в список nexthop-маршрута добавляются IP-адреса шлюзов (раздел 24.5.18) всех активных интерфейсов;

при резервировании в качестве nexthop-маршрута выбирается IP-адрес шлюза (раздел 24.5.18) активного интерфейса с наибольшим весом.

Использование отрицательной формы команды (no) исключает указанный интерфейс или туннель из правила MultiWAN.

## Синтаксис

[no] outbound { interface <IF> | tunnel <TUN> } [<WEIGHT> ]

## Параметры

<IF> – интерфейс, задаётся в виде, описанном в разделе 3.3;

<TUN> – имя туннеля, задаётся в виде, описанном в разделе 3.4. (только для IPv4);

<WEIGHT> – вес туннеля или интерфейса, определяется в диапазоне [1..255]. Если установить значение 2, то по данному интерфейсу будет передаваться в 2 раза больше трафика, чем по интерфейсу с дефолтным значением. В режиме резервирования активным будет маршрут с наибольшим весом. Значение по умолчанию 1.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-WAN-RULE  
CONFIG-IPV6-WAN-RULE

### Пример 1

```
esr(config-wan-rule)# outbound interface gigabitethernet 1/0/15
```

### Пример 2

```
esr(config-ipv6-wan-rule)# outbound interface bridge 2
```

## 24.5.7 *description*

Данной командой определяется описание правила.

Использование отрицательной формы команды (no) удаляет описание.

### Синтаксис

description <DESCRIPTION>  
no description

### Параметры

<DESCRIPTION> – описание правила wan, задаётся строкой до 255 символов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WAN-RULE  
CONFIG-IPV6-WAN-RULE

### Пример

```
esr(config-wan-rule)# description "tunnel to branch"
```

## 24.5.8 *failover*

Данной командой осуществляется переключение из режима балансировки в режим резервирования.

Использование отрицательной формы команды (no) возвращает режим балансировки.

### Синтаксис

[no] failover

### Параметры

Команда не содержит аргументов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WAN-RULE

### Пример

```
esr(config-wan-rule)# failover
```

## 24.5.9 target

---

Данной командой создается цель проверки и осуществляется переход в режим настройки параметров цели.

Использование отрицательной формы команды (no) удаляет созданную цель.

### Синтаксис

target <ID>

no target { <ID> | all }

### Параметры

<ID> – идентификатор цели, задаётся в пределах [1..50]. Если при удалении используется значение параметра «all», то будут удалены все цели для конфигурируемого списка целей.

- all - удалить все цели.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WAN-TARGET-LIST

CONFIG-IPV6-WAN-TARGET-LIST

### Пример

```
esr(config-target-list)# target 1
```

## 24.5.10 resp-time

---

Данной командой определяется время ожидания ответа на запрос по протоколу ICMP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

resp-time <TIME>

no resp-time

### Параметры

<TIME> – время ожидания, определяется в секундах [1..30].

### Значение по умолчанию

5

### Необходимый уровень привилегий

10

### Командный режим

### Пример

```
esr(config-wan-target)# resp-time 3
```

## 24.5.11 *ip address*

Данной командой указывается IP-адрес проверки.

Использование отрицательной формы команды (no) удаляет указанный адрес.

### Синтаксис

ip address <ADDR>

no ip address

### Параметры

<ADDR> – IP-адрес назначения, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-WAN-TARGET

### Пример

```
esr(config-wan-target)# ip address 18.168.1.25
```

## 24.5.12 *ipv6 address*

Данной командой указывается IPv6-адрес проверки.

Использование отрицательной формы команды (no) удаляет указанный адрес.

### Синтаксис

ipv6 address <IPV6-ADDR>

no ipv6 address

### Параметры

<IPV6-ADDR> – IPv6-адрес назначения, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IPV6-WAN-TARGET

### Пример



```
esr(config-ipv6-wan-target)# ipv6 address fc00::2
```

### 24.5.13 *wan load-balance enable*

Данной командой включается WAN режим на интерфейсе для IPv4 стека.

Использование отрицательной формы команды (no) выключает WAN режим для IPv4 стека.

#### Синтаксис

[no] wan load-balance enable

#### Параметры

Команда не содержит аргументов.

#### Значение по умолчанию

Выключено.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE

#### Пример

```
esr(config-if-gi)# wan load-balance enable
```

### 24.5.14 *ipv6 wan load-balance enable*

Данной командой включается WAN режим на интерфейсе для IPv6 стека.

Использование отрицательной формы команды (no) выключает WAN режим для IPv6 стека.

#### Синтаксис

[no] ipv6 wan load-balance enable

#### Параметры

Команда не содержит аргументов.

#### Значение по умолчанию

Выключено.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-GI  
CONFIG-SUBIF

**Пример**

```
esr(config-if-gi)# ipv6 wan load-balance enable
```

### **24.5.15 wan load-balance failure-count**

Данной командой определяется количество неудачных попыток проверки соединения, после которых, при отсутствии ответа от встречной стороны, соединение считается неактивным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

wan load-balance failure-count <VALUE>

no wan load-balance failure-count

**Параметры**

<VALUE> – количество попыток, определяется в диапазоне [1...10].

**Значение по умолчанию**

1

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

**Пример**

```
esr(config-if-gi)# wan load-balance failure-count 3
```

### **24.5.16 ipv6 wan load-balance failure-count**

Данной командой определяется количество неудачных попыток проверки соединения через IPv6 стек, после которых, при отсутствии ответа от встречной стороны, соединение считается неактивным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

**Синтаксис**

ipv6 wan load-balance failure-count <VALUE>

no ipv6 wan load-balance failure-count

**Параметры**

<VALUE> – количество попыток, определяется в диапазоне [1...10].

#### Значение по умолчанию

1

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

#### Пример

```
esr(config-if-gi)# ipv6 wan load-balance failure-count 3
```

### 24.5.17 *wan load-balance success-count*

Данной командой определяется количество успешных попыток проверки соединения, после которых, в случае успеха, соединение считается вновь активным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

wan load-balance success-count <VALUE>

no wan load-balance success-count

#### Параметры

<VALUE> – количество попыток, определяется в диапазоне [1...10].

#### Значение по умолчанию

1

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

#### Пример

```
esr(config-if-gi)# wan load-balance success-count 3
```

---

### **24.5.18      *ipv6 wan load-balance success-count***

---

Данной командой определяется количество успешных попыток проверки соединения по протоколу ipv6, после которых, в случае успеха, соединение считается вновь активным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### **Синтаксис**

ipv6 wan load-balance success-count <VALUE>

no ipv6 wan load-balance success-count

#### **Параметры**

<VALUE> – количество попыток, определяется в диапазоне [1...10].

#### **Значение по умолчанию**

1

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

#### **Пример**

```
esr(config-if-gi)# ipv6 wan load-balance success-count 3
```

---

### **24.5.19      *wan load-balance nexthop***

---

Данной командой определяется IP-адрес соседа, который будет указан в качестве одного из шлюзов в статическом маршруте, создаваемом службой MultiWAN.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес соседа.

#### **Синтаксис**

wan load-balance nexthop { <ADDR> | dhcp enable }

no wan load-balance nexthop

#### **Параметры**

<ADDR> – IP-адрес назначения (шлюз), задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

dhcp enable – если на интерфейсе IP-адрес получен через DHCP-клиента, используется шлюз с DHCP-сервера.

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG-GI

CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE  
CONFIG-IP4IP4  
CONFIG-GRE

### Пример

```
esr(config-gre)# wan load-balance nexthop 16.168.1.25
```

## 24.5.20 *ipv6 wan load-balance nexthop*

Данной командой определяется IPv6-адрес соседа, который будет указан в качестве одного из шлюзов в статическом маршруте, создаваемом службой MultiWAN.

Использование отрицательной формы команды (no) удаляет указанный IPv6-адрес соседа.

### Синтаксис

```
ipv6 wan load-balance nexthop { <IPV6-ADDR> }  
no ipv6 wan load-balance nexthop
```

### Параметры

<IPV6-ADDR> – IPv6-адрес назначения (шлюз), задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)#ipv6 wan load-balance nexthop 220::71
```

## 24.5.21 *wan load-balance source-address*

Данной командой определяется IP-адрес маршрутизатора, который будет использоваться в качестве IP-адреса источника в отправляемых ICMP пакетах для проверки целей.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес источника.

### Синтаксис

```
wan load-balance source-address <ADDR>  
no wan load-balance source-address
```

### Параметры

<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

### Пример

```
esr(config-gre)# wan load-balance source-address 16.168.1.25
```

## 24.5.22 *ipv6 wan load-balance source-address*

Данной командой определяется IPv6-адрес маршрутизатора, который будет использоваться в качестве IPv6-адреса источника в отправляемых ICMPv6 пакетах для проверки целей.

Использование отрицательной формы команды (no) удаляет указанный IPv6-адрес источника.

### Синтаксис

ipv6 wan load-balance source-address <IPv6-ADDR>

no ipv6 wan load-balance source-address

### Параметры

<IPv6-ADDR> – IPv6-адрес источника, задаётся в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)#ipv6 wan load-balance source-address 220::71
```

## 24.5.23 *wan load-balance target-list check-all*

Данной командой будут проверяться все IP-адреса из списка проверки целостности. В случае недоступности одного из проверяемых узлов, шлюз будет считаться недоступным.

Использование отрицательной формы команды (no) отменяет проверку всех IP-адресов из списка проверки целостности.

#### Синтаксис

[no] wan load-balance target-list check-all

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-GRE

#### Пример

```
esr(config-gre)# wan load-balance target-list check-all
```

### **24.5.24      *ipv6 wan load-balance target-list check-all***

Данной командой будут проверяться все IPv6-адреса из списка проверки целостности.

Использование отрицательной формы команды (no) отменяет проверку всех IPv6-адресов из списка проверки целостности. В случае недоступности одного из проверяемых узлов, шлюз будет считаться недоступным.

#### Синтаксис

[no] ipv6 wan load-balance target-list check-all

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

#### Пример

```
esr(config-if-gi)# ipv6 wan load-balance target-list check-all
```

## 25 УПРАВЛЕНИЕ QoS

### 25.1 qos enable

Данной командой включается сервис QoS на интерфейсе. Если к интерфейсу не привязана политика QoS (привязка политик описана в разделе 25.12), то интерфейс работает в режиме Basic QoS, иначе Policy-based QoS.

Basic QoS – классификация трафика выполняется на основе кодов DSCP и/или 802.1p в зависимости от выбранного режима доверия (команда описана в разделе 25.2). Трафик направляется в очереди в соответствии с таблицами DSCP-Queue и/или CoS-Queue.

Policy-based QoS – классификация и направление трафика в очереди выполняется на основе QoS политик. В каждой политике определяется набор классов, на которые необходимо разделить трафик. Отношение трафика к определенному классу политики определяется на входе в маршрутизатор правилами ACL (привязка ACL описана в разделе 25.21), для этого назначается QoS политика на входящее направление. Для ограничения полосы ранее классифицированного трафика и других функций QoS политика назначается на исходящее направление.

Использование отрицательной формы команды (no) выключает сервис QoS на интерфейсе.

#### Синтаксис

[no] qos enable

#### Параметры

Команда не содержит аргументов.

#### Значение по умолчанию

Выключено

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-L2TPV3

CONFIG-SUBTUNNEL

#### Пример

```
esr(config-if-gi)# qos enable
```



## 25.2 qos trust

Данной командой устанавливается режим доверия к значениям кодов 802.1p и DSCP во входящих пакетах для Basic QoS-режима работы интерфейса.

Использование отрицательной формы команды (no) устанавливает соответствие по умолчанию.

### Синтаксис

qos trust <MODE>

no qos trust

### Параметры

<MODE> – режим доверия к значениям кодов 802.1p и DSCP, принимает одно из следующих значений:

dscp – режим доверия значениям кодов DSCP в IP-заголовке. Не IP-пакеты будут направлены в очередь по умолчанию (команда описана в разделе 25.7);

cos – режим доверия значениям кодов 802.1p в теге 802.1q. Нетегированные пакеты будут направлены в очередь по умолчанию (команда описана в разделе 25.7);

cos-dscp – режим доверия значениям кодов DSCP для IP-пакетов и значениям кодов 802.1p для остальных пакетов.

### Значения по умолчанию:

Режим доверия значениям кодов DSCP (dscp).

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# qos trust cos-dscp
```

## 25.3 qos map dscp-queue

Данная команда устанавливает соответствие между значениями кодов DSCP входящих пакетов и исходящими очередями.

Использование отрицательной формы команды (no) устанавливает соответствие по умолчанию.

### Синтаксис

qos map dscp-queue <DSCP> to <QUEUE>

no qos map dscp-queue <DSCP>

### Параметры

<DSCP> – классификатор обслуживания в IP-заголовке пакета, принимает значения [0..63];

<QUEUE> – идентификатор очереди, принимает значения [1..8].

**Значения по умолчанию:**

DSCP: (0-7), очередь 1  
DSCP: (8-15), очередь 2  
DSCP: (16-23), очередь 3  
DSCP: (24-31), очередь 4  
DSCP: (32-39), очередь 5  
DSCP: (40-47), очередь 6  
DSCP: (48-55), очередь 7  
DSCP: (56-63), очередь 8

**Необходимый уровень привилегий**

10

**Командный режим**

CONFIG

**Пример**

```
esr(config)# qos map dscp-queue 42 to 5
```

## ***25.4 qos map cos-queue***

Данная команда устанавливает соответствие между значениями кодов 802.1p входящих пакетов и исходящими очередями.

Использование отрицательной формы команды (no) устанавливает соответствие по умолчанию.

**Синтаксис**

qos map cos-queue <COS> to <QUEUE>  
no qos map dscp-queue <COS>

**Параметры**

<COS> – классификатор обслуживания в теге 802.1q пакета, принимает значения [0..7];  
<QUEUE> – идентификатор очереди, принимает значения [1..8].

**Значения по умолчанию:**

CoS: (0), очередь 1  
CoS: (1), очередь 2  
CoS: (2), очередь 3  
CoS: (3), очередь 4  
CoS: (4), очередь 5  
CoS: (5), очередь 6  
CoS: (6), очередь 7  
CoS: (7), очередь 8

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# qos map cos-queue 7 to 5
```

## 25.5 qos map dscp-mutation

Данная команда устанавливает соответствие между значениями кодов DSCP входящих пакетов и кодов DSCP на выходе из устройства.

Использование отрицательной формы команды (no) устанавливает соответствие по умолчанию.

## Синтаксис

```
qos map dscp-queue <DSCP> to <DSCP>
```

```
no qos map dscp-queue <DSCP>
```

## Параметры

<DSCP> – классификатор обслуживания в IP-заголовке пакета, принимает значения [0..63].

## Значения по умолчанию:

Значения кодов DSCP входящих пакетов и кодов DSCP на выходе из устройства совпадают.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# qos map dscp-mutation 10 to 40
```

## 25.6 qos dscp-mutation

Данной командой включается применение изменений кодов DSCP в соответствии с таблицей DSCP-Mutation. Коды DSCP изменяются только для входящего трафика доверенных портов в режиме QoS Basic.

Использование отрицательной формы команды (no) отключает изменение кодов DSCP.

## Синтаксис

```
[no] qos dscp mutation
```

## Параметры

Команда не содержит аргументов.

## Значение по умолчанию

Выключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# qos wrr-queue 3 bandwidth 130
```

## 25.7 qos queue default

Данная команда устанавливает номер очереди по умолчанию, в которую попадает весь трафик кроме IP в режиме доверия DSCP-приоритетам в случае с Basic QoS, а также неклассифицированный трафик в случае с Policy-based QoS.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

qos queue default <QUEUE>

no qos queue default

### Параметры

<QUEUE> – идентификатор очереди, принимает значения [1..8].

### Значение по умолчанию

1

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# qos queue default 3
```

## 25.8 priority-queue out num-of-queues

Данная команда задает количество приоритетных очередей. Оставшиеся очереди являются взвешенными.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

priority-queue out num-of-queues <VALUE>

no priority-queue out num-of-queues

### Параметры

<VALUE> – количество очередей, принимает значение [1..8], где:

0 – все очереди участвуют в WRR (WRR – механизм обработки очередей на основе веса);

8 – все очереди обслуживаются как «strict priority» (strict priority – приоритетная очередь обслуживается сразу, как только появляются пакеты).

#### Значение по умолчанию

8

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

```
esr(config)# priority-queue out num-of-queues 5
```

## 25.9 qos wrr-queue

Данной командой определяются веса для соответствующих взвешенных очередей.

Использование отрицательной формы команды (no) устанавливает значение веса для указанной очереди по умолчанию.

#### Синтаксис

```
qos wrr-queue <QUEUE> bandwidth <WEIGHT>
```

```
no qos wrr-queue <QUEUE>
```

#### Параметры

<QUEUE> – идентификатор очереди, принимает значение [1..8];

<WEIGHT> – значение веса, принимает значение [1..255].

#### Значение по умолчанию

1

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

```
esr(config)# qos wrr-queue 3 bandwidth 130
```

## 25.10 traffic-shape

Данная команда устанавливает ограничение скорости исходящего трафика для определенной очереди или интерфейса в целом. Команда актуальна только для Basic QoS режима интерфейса.

Использование отрицательной формы команды (no) снимает ограничение.

#### Синтаксис

```
traffic-shape { <BANDWIDTH> [BURST] | queue <QUEUE> <BANDWIDTH> [BURST] }
```

```
no traffic-shape [ queue <QUEUE> ]
```

## Параметры

<QUEUE> – идентификатор очереди, принимает значение [1..8];

<BANDWIDTH> – средняя скорость трафика в Кбит/с, [64..1000000] для интерфейсов и туннелей;

<BURST> – размер сдерживающего порога в КБайт, принимает значение [4..16000]. По умолчанию 128 КБайт.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-L2TPV3

## Пример

```
esr(config)# qos traffic-shape queue 3 100000 2000
```

## 25.11 rate-limit

Данная команда устанавливает ограничение скорости входящего трафика. Команда актуальна только для Basic QoS режима интерфейса.

Использование отрицательной формы команды (no) снимает ограничение.

## Синтаксис

rate-limit <BANDWIDTH> [BURST]

no rate-limit

## Параметры

<BANDWIDTH> – средняя скорость трафика в Кбит/с, принимает значение [64..1000000] для интерфейсов и туннелей;

<BURST> – размер сдерживающего порога в КБайт, принимает значение [4..16000]. По умолчанию 128 КБайт.

## Значение по умолчанию

Отключено.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI  
 CONFIG-SUBIF  
 CONFIG-PORT-CHANNEL  
 CONFIG-BRIDGE  
 CONFIG-LOOPBACK  
 CONFIG-IP4IP4  
 CONFIG-GRE  
 CONFIG-VTI  
 CONFIG-L2TPV3

### Пример

```
esr(config-if-gi)# rate-limit 1670000
```

## 25.12 service-policy

Данная команда используется для привязки указанной QoS-политики к конфигурируемому интерфейсу для классификации входящего (**input**) или приоритезации исходящего (**output**) трафика. Использование отрицательной формы команды (**no**) удаляет привязку политики к данному интерфейсу.

### Синтаксис

```

service-policy { input | output } <NAME>
no service-policy { input | output }

```

### Параметры

<NAME> – имя QoS-политики, задаётся строкой до 31 символа. (Для виртуального интерфейса туннеля SoftGRE(subtunnel) возможно использование только input).

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI  
 CONFIG-SUBIF  
 CONFIG-PORT-CHANNEL  
 CONFIG-BRIDGE  
 CONFIG-LOOPBACK  
 CONFIG-IP4IP4  
 CONFIG-GRE  
 CONFIG-VTI  
 CONFIG-L2TPV3  
 CONFIG-SUBTUNNEL

### Пример

```
esr(config-if-gi)# service-policy input input_policy
```

---

## **25.13    *policy-map***

---

Данной командой создается политика QoS и осуществляется переход в режим настройки параметров политики.

Использование отрицательной формы команды (no) удаляет созданную политику.

### **Синтаксис**

```
[no] policy-map <NAME>
```

### **Параметры**

<NAME> – имя создаваемой политики, задается строкой до 31 символа.

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG

### **Пример**

```
esr(config)# policy-map input_policy
```

---

## **25.14    *ip firewall session classification enable***

---

Командой выполняется включение классифицирования сессий на основе политики QoS.

Использование отрицательной формы команды (no) отключает классифицирование сессий.

### **Синтаксис**

```
[no] ip firewall classification enable
```

### **Параметры**

Команда не содержит аргументов.

### **Значение по умолчанию**

Классифицирование сессий отключено.

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG

### **Пример**

```
esr(config)# ip firewall sessions classification enable
```

---

## **25.15    *class-map***

---

Данной командой создается класс QoS и осуществляется переход в режим настройки параметров класса.

Использование отрицательной формы команды (no) удаляет созданный класс.



### Синтаксис

[no] class-map <NAME>

### Параметры

<NAME> – имя создаваемого класса, задается строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# class-map telnet_traffic
```

---

## 25.16 class

---

Данная команда используется для привязки указанного QoS-класса к политике и осуществляется переход в режим настройки параметров класса.

Использование отрицательной формы команды (no) удаляет привязку к классу.

### Синтаксис

[no] class <NAME>

### Параметры

<NAME> – имя привязываемого класса, задается строкой до 31 символа. При указании значения «class-default» в данный класс попадает трафик неклассифицированный на входе.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-POLICY-MAP

### Пример

```
esr(config-policy-map)# class telnet_traffic
esr(config-class-policy-map)#
```

---

## 25.17 set dscp

---

Данной командой задается значение кода DSCP, которое будет установлено в IP-пакетах, соответствующих конфигурируемому классу.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

set dscp <DSCP>

no set dscp

### Параметры

<DSCP> – значение кода DSCP, принимает значения [0..63].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-CLASS-MAP

### Пример

```
esr(config-class-map)# set dscp 16
```

## 25.18 *set cos*

Данной командой задается значение 802.1p приоритета, которое будет установлено в пакетах, соответствующих конфигурируемому классу.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

set cos <COS>

no set cos

### Параметры

<COS> – значение 802.1p приоритета, принимает значения [0..7].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-CLASS-MAP

### Пример

```
esr(config-class-map)# set cos 2
```

## 25.19 *set ip-precedence*

Данной командой задается значение кода IP Precedence, которое будет установлено в IP-пакетах, соответствующих конфигурируемому классу.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

set ip-precedence <IPP>

no set ip-precedence

### Параметры

<IPP> – значение кода IP Precedence, принимает значения [0..7].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-CLASS-MAP

## Пример

```
esr(config-class-map)# set ip-precedence 5
```

## 25.20 *set queue*

Данной командой задается номер выходной аппаратной очереди QoS, в которую будут направлены пакеты, соответствующие конфигурируемому классу.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
set queue <QUEUE>
```

```
no set queue
```

### Параметры

<QUEUE> – номер выходной аппаратной очереди QoS, принимает значения [1..8].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-CLASS-MAP

## Пример

```
esr(config-class-map)# set queue 5
```

## 25.21 *match access-group*

Команда используется для привязки списка контроля доступа (ACL), по которому будет определяться отношение входящего трафика к конфигурируемому классу.

Использование отрицательной формы команды (no) удаляет привязку списка контроля доступа к данному классу.

### Синтаксис

```
match access-group <NAME>
```

```
no match access-group
```

### Параметры

<NAME> – имя списка контроля доступа, задаётся строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-CLASS-MAP

## Пример

```
esr(config-if-gi)# match access-group acl-ssh-traffic
```

---

## 25.22 *service-policy*

---

Данной командой привязывается политика QoS к классу для создания иерархического QoS. Использование отрицательной формы команды (no) удаляет привязку политики к классу.

### Синтаксис

[no] service-policy <NAME>

### Параметры

<NAME> – имя политики, задается строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-POLICY-MAP-CLASS

### Пример

```
esr(config-class-policy-map)# service-policy input_policy
```

---

## 25.23 *shape auto-distribution*

---

Данной командой включается автоматическое распределение полосы пропускания между классами, в которых нет настройки полосы пропускания, включая класс по умолчанию.

Использование отрицательной формы команды (no) отключает автоматическое распределение полосы.

### Синтаксис

[no] shape auto-distribution

### Параметры

Команда не имеет параметров.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-POLICY-MAP

### Пример

```
esr(config-policy-map)# shape auto-distribution
```

---

## 25.24 *shape average*

---

Данной командой устанавливается гарантированная полоса исходящего трафика для определенного класса или политики в целом.

Использование отрицательной формы команды (no) снимает ограничение.

### Синтаксис

shape average <BANDWIDTH> [BURST]

no shape average

## Параметры

<BANDWIDTH> – гарантированная полоса трафика в Кбит/с, принимает значение [64..10000000];

<BURST> – размер сдерживающего порога в КБайт, принимает значение [4..16000]. По умолчанию 128 КБайт.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-POLICY-MAP

CONFIG-POLICY-MAP-CLASS

## Пример

```
esr(config-policy-map)# shape average 100000 2000
```

## 25.25 *shape peak*

Устанавливается разделяемая полоса исходящего трафика для определенного класса. Данную полосу класс может занять, если менее приоритетный класс не занял свою гарантированную полосу.

Использование отрицательной формы команды (no) снимает ограничение.

## Синтаксис

shape peak <BANDWIDTH> [BURST]

no shape peak

## Параметры

<BANDWIDTH> – разделяемая полоса трафика в Кбит/с, принимает значение [64..10000000];

<BURST> – размер сдерживающего порога в КБайт, принимает значение [4..16000]. По умолчанию 128 КБайт.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-POLICY-MAP-CLASS

## Пример

```
esr(config-policy-map)# shape average 100000 2000
```

## 25.26 *mode*

Данной командой определяется режим работы класса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

mode <MODE>

no mode

## Параметры

<MODE> – режим класса:

fifo – режим FIFO (First In, First Out);

gred – режим GRED (Generalized RED);

red – режим RED (Random Early Detection);

sfq – режим SFQ (очередь SFQ распределяет передачу пакетов на базе потоков).

## Значение по умолчанию

FIFO

## Необходимый уровень привилегий

10

## Командный режим

CONFIG- POLICY-MAP-CLASS

## Пример

```
esr(config-class-policy-map)# mode red
```

## 25.27 *priority class*

Данной командой задается приоритет класса в WRR-процессе. Классы с наименьшим приоритетом обрабатываются в первую очередь.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

priority class <PRIORITY>

no priority

## Параметры

<PRIORITY> – приоритет класса в WRR-процессе, принимает значения [1..8].

## Значение по умолчанию

8

## Необходимый уровень привилегий

10

## Командный режим

CONFIG- POLICY-MAP-CLASS

## Пример

```
esr(config-class-policy-map)# priority class 5
```

## 25.28 *priority level*

Данной командой класс переводится в режим Strict Priority и задается приоритет класса. Классы с наименьшим приоритетом обрабатываются в первую очередь.

Использование отрицательной формы команды (no) переводит класс в режим WRR.

### Синтаксис

priority level <PRIORITY>

no priority

### Параметры

<PRIORITY> – приоритет класса в Strict Priority-процессе, принимает значения [1..8].

### Значение по умолчанию

Класс работает в режиме WRR, приоритет не задан.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG- POLICY-MAP-CLASS

### Пример

```
esr(config-class-policy-map)# priority level 5
```

## 25.29 *fair-queue*

Данной командой определяется предельное количество виртуальных очередей.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

fair-queue <QUEUE-LIMIT>

no fair-queue

### Параметры

<QUEUE-LIMIT> – предельное количество виртуальных очередей, принимает значения в диапазоне [16..4096].

### Значение по умолчанию

16

### Необходимый уровень привилегий

10

### Командный режим

CONFIG- POLICY-MAP-CLASS

### Пример

```
esr(config-class-policy-map)# fair-queue 200
```

---

## 25.30 *queue-limit*

---

Данной командой определяется предельное количество пакетов для виртуальной очереди.  
Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

```
queue-limit <QUEUE-LIMIT>  
no queue-limit
```

### Параметры

<QUEUE-LIMIT> – предельное количество пакетов в виртуальной очереди, принимает значения в диапазоне [2..4096].

### Значение по умолчанию

127

### Необходимый уровень привилегий

10

### Командный режим

CONFIG- POLICY-MAP-CLASS

### Пример

```
esr(config-class-policy-map)# queue-limit 200
```

---

## 25.31 *random-detect*

---

Данной командой определяются параметры RED.  
Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

```
random-detect <LIMIT> <MAX> <MIN> <PROBABILITY>  
no random-detect
```

### Параметры

<LIMIT> – предельный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<MAX> – максимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<MIN> – минимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<PROBABILITY> – вероятность отбрасывания пакетов, принимает значения [0..100].

При указании значений должны выполняться следующие правила:

```
<MAX> > 2 * <MIN>  
<LIMIT> > 3 * <MAX>
```

### Необходимый уровень привилегий

10



## Командный режим

CONFIG- POLICY-MAP-CLASS

### Пример

```
esr(config-class-policy-map)# random-detect 9000 1500 3000 10
```

## 25.32 *random-detect precedence*

Данной командой определяются параметры GRED.

Использование отрицательной формы команды (no) отменяет назначение.

### Синтаксис

random-detect precedence <PRECEDENCE> <LIMIT> <MAX> <MIN> <PROBABILITY>

no random-detect precedence <PRECEDENCE>

### Параметры

<PRECEDENCE> – значение IP Precedence [0..7];

<LIMIT> – предельный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<MAX> – максимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<MIN> – минимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<PROBABILITY> – вероятность отбрасывания пакетов, принимает значения [0..100].

При указании значений должны выполняться следующие правила:

<MAX> > 2 \* <MIN>

<LIMIT> > 3 \* <MAX>

### Необходимый уровень привилегий

10

## Командный режим

CONFIG- POLICY-MAP-CLASS

### Пример

```
esr(config-class-policy-map)# random-detect precedence 2 9000 1500 3000 10
```

## 25.33 *show qos interface*

Данная команда показывает параметры QoS сетевых интерфейсов.

### Синтаксис

show qos interface shapers <IF>

### Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе 3.3.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show qos interface shapers gigabitethernet 1/0/2
gigabitethernet 1/0/2
Committed rate: 100000 Kbps
Committed burst: 1600 KBytes
```

## 25.34 *show qos tunnel*

Данная команда показывает параметры QoS-туннелей.

## Синтаксис

show qos tunnel shapers <TUN>

## Параметры

<TUN> – имя туннеля, задаётся в виде, описанном в разделе 3.4.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show qos tunnel shapers vti 2
vti 2
qid    Target          Target
       Committed       Committed
       Rate [Kbps]     Burst [KBytes]
---    -
1      10000           128
2      6000            128
```

## 25.35 *show qos statistics*

Данная команда выводит статистику по переданным и отброшенным пакетам. Команда актуальна только для Basic QoS-режима интерфейса.

## Синтаксис

show qos statistics [ <IF> | <TUN> ]

## Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе 3.3;

<TUN> – имя туннеля, задаётся в виде, описанном в разделе 3.4.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```
esr# show qos statistics vti 2
vti 2
```

| Queue | Bytes     | Packets | Drops   |
|-------|-----------|---------|---------|
| 1     | 0         | 0       | 0       |
| 2     | 0         | 0       | 0       |
| 3     | 0         | 0       | 0       |
| 4     | 0         | 0       | 0       |
| 5     | 0         | 0       | 0       |
| 6     | 964073836 | 1413598 | 0       |
| 7     | 121389180 | 177990  | 1235497 |
| 8     | 0         | 0       | 0       |

## 25.36 *show qos policy binding*

Данная команда используется для просмотра привязанной QoS-политики к заданному интерфейсу для классификации входящего (**input**) или приоритезации исходящего (**output**) трафика.

### Синтаксис

show qos policy statistics [ <IF> | <TUN> ]

### Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе 3.3;

<TUN> – имя туннеля, задаётся в виде, описанном в разделе 3.4.

### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```
esr# show qos policy binding gil/0/1
gigabitethernet 1/0/1
Output: parent
  Class: class1
    Policy: child1
      Class: class3
  Class: class2
    Policy: child2
```

## 25.37 *show qos policy configuration*

Данная команда выводит конфигурацию заданной политики QoS.

### Синтаксис

show qos policy configuration <NAME>

### Параметры

<NAME> – имя политики, задается строкой до 31 символа.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример 1

```
esr# show qos policy configuration parent
Policy-map: parent
  Shape average:
    Bandwidth:      --
    Burst:          128
  Class: class1
    Set:
      COS:          --
      DSCP:         --
      IP-Precedence: --
      Queue:        --
    Match access group: --
    Shape average:
      Bandwidth:      --
      Burst:          128
    Shape peak:
      Bandwidth:      --
      Burst:          128
    Mode:            FIFO
    Priority class:   8
    Queue limit:     127
    Service policy:  child1
  Class: class2
    Set:
      COS:          --
      DSCP:         --
      IP-Precedence: --
      Queue:        --
    Match access group: --
    Shape average:
      Bandwidth:      --
      Burst:          128
    Shape peak:
      Bandwidth:      --
      Burst:          128
    Mode:            FIFO
    Priority class:   8
    Queue limit:     127
    Service policy:  child2
```

## 25.38 *show qos policy statistics*

Данная команда выводит статистику по переданным и отброшенным пакетам. Команда актуальна только для Policy-based QoS режима интерфейса.

### Синтаксис

```
show qos policy statistics [ <IF> | <TUN> ]
```

### Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе 3.3;

<TUN> – имя туннеля, задаётся в виде, описанном в разделе 3.4.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример 1

```
esr# show qos policy statistics vti 2
```

## 25.39 *show qos map dscp-queue*

Данная команда показывает информацию о соответствии кодов DSCP в пакетах и выходных очередях, используемых в QoS.

## Синтаксис

show qos map dscp-queue

## Параметры

Команда не содержит аргументов.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

## Пример

```
esr# show qos map dscp-queue
d1 : d2  0  1  2  3  4  5  6  7  8  9
-----
0        01 01 01 01 01 01 01 01 02 02
1        02 02 02 02 02 02 03 03 03 03
2        03 03 03 03 04 04 04 04 04 04
3        04 04 05 05 05 05 05 05 05 05
4        06 06 06 06 06 06 06 06 07 07
5        07 07 07 07 07 07 08 08 08 08
6        08 08 08 08
```

## 25.40 *show qos map cos-queue*

Данная команда показывает информацию о соответствии кодов 802.1p в пакетах и выходных очередях, используемых в QoS.

## Синтаксис

show qos map cos-queue

## Параметры

Команда не содержит аргументов.

## Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```
esr# show qos map dscp-queue
d1 : d2   0  1  2  3  4  5  6  7
-----
0         01 02 03 04 05 06 07 08
```

## 25.41 *show qos map dscp-mutation*

Данная команда показывает информацию о соответствии кодов DSCP в пакетах и кодов DSCP после изменений.

### Синтаксис

```
show qos map dscp-mutation
```

### Параметры

Команда не содержит аргументов.

### Необходимый уровень привилегий

1

## Командный режим

ROOT

### Пример

```
esr# show qos map dscp-mutation
d1 : d2   0  1  2  3  4  5  6  7  8  9
-----
0         00 01 02 03 04 05 06 07 08 09
1         10 11 12 13 14 15 16 17 18 19
2         20 21 22 23 24 25 26 27 28 29
3         30 31 32 33 34 35 36 37 38 39
4         40 41 42 43 44 45 46 47 48 49
5         50 51 52 53 54 55 56 57 58 59
6         60 61 62 63
```

## 26 УПРАВЛЕНИЕ NETFLOW

### 26.1 netflow collector

Данная команда используется для создания коллектора Netflow и перехода в командный режим CONFIG-NETFLOW-HOST.

Использование отрицательной формы команды (no) удаляет сконфигурированный коллектор Netflow.

#### Синтаксис

[no] netflow collector <ADDR>

#### Параметры

<ADDR> – IP-адрес коллектора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

```
esr(config)# netflow collector 10.100.100.1
esr(config-netflow-host)#
```

### 26.2 port

Данной командой определяется порт Netflow-сервиса на сервере сбора статистики.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

port <PORT>  
no port

#### Параметры

<PORT> – номер UDP-порта, указывается в диапазоне [1..65535].

#### Значение по умолчанию

2055

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-NETFLOW-HOST

#### Пример

```
esr(config-netflow-host)# port 5555
```

---

## 26.3 netflow version

---

Данной командой задаётся версия Netflow-протокола.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

netflow version <VERSION>

no netflow version

### Параметры

<VERSION> – версия Netflow-протокола: 5, 9 и 10.

### Значение по умолчанию

9

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# netflow version 10
```

---

## 26.4 netflow max-flows

---

Данной командой задаётся максимальное количество наблюдаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

netflow max-flows <COUNT>

no netflow max-flows

### Параметры

<COUNT> – количество наблюдаемых сессий, принимает значение [10000..2000000].

### Значение по умолчанию

512000

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# netflow max-flows 300000
```



---

## 26.5 netflow inactive-timeout

---

Данной командой задаётся интервал, по истечении которого информация об устаревших сессиях экспортируются на коллектор.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

netflow inactive-timeout <TIMEOUT>

no netflow inactive-timeout

### Параметры

<TIMEOUT> – задержка перед отправкой информации об устаревших сессиях, задается в секундах, принимает значение [0..240].

### Значение по умолчанию

15 секунд

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# netflow inactive-timeout 30
```

---

## 26.6 netflow refresh-rate

---

Данной командой задаётся частота отправки статистики на Netflow-коллектор.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

netflow refresh-rate <RATE>

no netflow refresh-rate

### Параметры

<RATE> – частота отправки статистики, задается в пакетах на поток, принимает значение [1..10000].

### Значение по умолчанию

10

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# netflow refresh-rate 100
```

---

## **26.7 netflow enable**

---

Данной командой активируется Netflow на маршрутизаторе.

Использование отрицательной формы команды (no) деактивирует Netflow на маршрутизаторе.

### **Синтаксис**

[no] netflow enable

### **Параметры**

Команда не содержит аргументов.

### **Значение по умолчанию**

Процесс выключен.

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG

### **Пример**

```
esr(config)# netflow enable
```

---

## **26.8 ip netflow export**

---

Данная команда используется для включения экспорта статистики Netflow на сетевом интерфейсе. Функция Netflow на сетевом интерфейсе может быть включена, если на интерфейсе выключена функция Firewall (раздел 19.4), в ином случае экспорт статистики Netflow настраивается в правиле Firewall (раздел 19.20).

Использование отрицательной формы команды (no) отключает экспорт статистики Netflow на сетевом интерфейсе.

### **Синтаксис**

[no] ip netflow export

### **Параметры**

Команда не содержит аргументов.

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-VTI

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ip netflow export
```

## 26.9 show netflow configuration

Командой выполняется просмотр параметров конфигурации Netflow-агента.

### Синтаксис

```
show netflow configuration
```

### Параметры

Команда не содержит аргументов.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

## Пример

```
esr# show netflow configuration
Netflow configuration:
Global state:      Enabled
Version:           9
Maxflows:          10001
Refresh rate:      10
Inactive timeout:  15

Host: 115.0.0.10  Port: 2055
```

## 26.10 show netflow statistics

Команда для просмотра текущей информации о работе Netflow.

### Синтаксис

```
show netflow statistics
```

### Параметры

Команда не содержит аргументов.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

## Пример

```
esr# show netflow statistics
Flows: active 9 (peak 34 reached 1d4h20m ago), mem 3841K
Hash: size 491496 (mem 3839K). InHash: 760 pkt, 339 K, InPDU 4, 160.

Processed rate  Bits/s          Packets/s
-----
```

```
Current          5142          2
1 Min Avg        4921          0
5 Min Avg        4874          0
Export: Rate 0 bytes/s; Total 3952 pkts, 3 MB, 28818 flows; Errors 2 pkts; Traffic
lost 0 pkts, 0 Kbytes, 0 flows.
```

## 26.11 *show netflow statistics cpu*

Команда для просмотра статистики по распределению информации о нагрузке Netflow на ЦПУ.

### Синтаксис

```
show netflow statistics
```

### Параметры

Команда не содержит аргументов.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show netflow statistics cpu
cpu      PPS      Traffic,   Traffic,   Drop,      Drop,
      PPS      Packets   MBytes    Packets    KBytes
-----
Total    1          215224    61         0          0
cpu0     0           0         0         0          0
cpu1     0        10485     0         0          0
cpu2     0         2676     0         0          0
cpu3     0        12893     0         0          0
cpu4     0           0         0         0          0
cpu5     1       106264    53         0          0
cpu6     0         2684     0         0          0
cpu7     0        10213     0         0          0
cpu8     0         6770     0         0          0
cpu9     0         5424     0         0          0
cpu10    0         2505     0         0          0
cpu11    0        10919     1         0          0
cpu12    0       13395     0         0          0
cpu13    0         2769     0         0          0
cpu14    0        14050     0         0          0
cpu15    0        14177     1         0          0
```

## 27 УПРАВЛЕНИЕ SFLOW

### 27.1 sflow collector

Данная команда используется для создания коллектора sFlow и перехода в командный режим CONFIG-SFLOW-HOST.

Использование отрицательной формы команды (no) удаляет сконфигурированный коллектор sFlow.

#### Синтаксис

[no] netflow collector <ADDR>

#### Параметры

<ADDR> – IP-адрес коллектора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

```
esr(config)# sflow collector 10.100.100.1
esr(config-sflow-host)#
```

### 27.2 port

Данной командой определяется порт sFlow-сервиса на сервере сбора статистики.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

port <PORT>

no port

#### Параметры

<PORT> – номер UDP-порта, указывается в диапазоне [1..65535].

#### Значение по умолчанию

6343

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-SFLOW-HOST

#### Пример

```
esr(config-sflow-host)# port 5556
```

---

## 27.3 *sflow poll-interval*

---

Данной командой задаётся интервал, по истечении которого происходит получение информации о счетчиках сетевого интерфейса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

sflow poll-interval <TIMEOUT>

no sflow poll-interval

### Параметры

<TIMEOUT> – интервал, по истечении которого происходит получение информации о счетчиках сетевого интерфейса, принимает значение [1..10000].

### Значение по умолчанию

10 секунд

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# sflow poll-interval 30
```

---

## 27.4 *sflow sampling-rate*

---

Данной командой задаётся частота отправки пакетов пользовательского трафика в неизменном виде на sFlow-коллектор.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

sflow sampling-rate <RATE>

no sflow sampling-rate

### Параметры

<RATE> – частота отправки пакетов пользовательского трафика на коллектор, принимает значение [1..10000000]. При значении частоты 10 на коллектор будет отправлен один пакет из десяти.

### Значение по умолчанию

1000

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

## Пример

```
esr(config)# sflow sampling-rate 100
```

## 27.5 sflow enable

Данной командой активируется sFlow на маршрутизаторе.

Использование отрицательной формы команды (no) деактивирует sFlow на маршрутизаторе.

### Синтаксис

```
[no] sflow enable
```

### Параметры

Команда не содержит аргументов.

### Значение по умолчанию

Процесс выключен.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

## Пример

```
esr(config)# sflow enable
```

## 27.6 ip sflow export

Данная команда используется для включения экспорта статистики sFlow на сетевом интерфейсе. Функция sFlow на сетевом интерфейсе может быть включена, если на интерфейсе выключена функция Firewall (раздел 19.4), в ином случае экспорт статистики sFlow настраивается в правиле Firewall (раздел 19.20).

Использование отрицательной формы команды (no) отключает экспорт статистики sFlow на сетевом интерфейсе.

### Синтаксис

```
[no] ip sflow export
```

### Параметры

Команда не содержит аргументов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-VTI

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# ip sflow export
```

## ***27.7 show sflow configuration***

Командой выполняется просмотр параметров конфигурации sFlow-агента.

### Синтаксис

show sflow configuration

### Параметры

Команда не содержит аргументов.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show sflow configuration
sFlow configuration:
Global state:   Enabled
Sample rate:    1000
Poll interval:  10
Host: 115.0.0.10 Port: 6800
Host: 115.0.0.20 Port: 6343
Host: 115.0.0.30 Port: 6343
```



## 28 МОНИТОРИНГ И УПРАВЛЕНИЕ

### 28.1 Настройка SNMP

#### 28.1.1 *snmp-server*

Данной командой включается SNMP-сервер. Использование отрицательной формы команды (no) выключает SNMP-сервер.

##### Синтаксис

```
[no] snmp-server [ vrf <VRF> ]
```

##### Параметры

<VRF> – имя экземпляра VRF, в рамках которого будет работать доступ по SNMP протоколу, задается строкой до 31 символа.

##### Значение по умолчанию

Выключен

##### Необходимый уровень привилегий

10

##### Командный режим

CONFIG

##### Пример

```
esr(config)# snmp-server
```

#### 28.1.2 *snmp-server host*

Данной командой включается передача SNMP уведомлений на указанный IP адрес и осуществляется переход в режим настройки SNMP уведомлений.

Использование отрицательной формы команды (no) отключает передачу уведомлений на указанный коллектор SNMP уведомлений.

##### Синтаксис

```
snmp-server host { <ADDR> | <IPv6-ADDR> } [vrf <VRF>]  
no snmp-server host { <ADDR> | <IPv6-ADDR> } [ vrf <VRF> ]
```

##### Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPv6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<VRF> – имя экземпляра VRF, в котором находится коллектор SNMP уведомлений, задается строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# snmp host 192.168.2.2
```

### 28.1.3 *port*

Данной командой определяется порт коллектора SNMP уведомлений на удаленном сервере.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

port <PORT>  
no port

## Параметры

<PORT> – номер UDP-порта, указывается в диапазоне [1..65535].

## Значение по умолчанию

162

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SNMP-HOST

## Пример

```
esr(config-snmp-host)# port 5555
```

### 28.1.4 *snmp-server community*

Данной командой определяется сообщество для доступа по протоколу SNMP.

Использование отрицательной формы команды (no) удаляет настройки сообщества.

## Синтаксис

[no] snmp-server community <COMMUNITY> [ <TYPE> ] [ { <ADDR> | <IPV6-ADDR> } ] [ vrf <VRF> ]

## Параметры

<COMMUNITY> – сообщество для доступа по протоколу SNMP;

<TYPE> – уровень доступа:

ro – доступ только для чтения;

rw – доступ для чтения и записи.

<ADDR> – IP-адрес клиента, которому предоставлен доступ, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPv6-ADDR> – IPv6-адрес клиента, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<VRF> – имя экземпляра VRF, из которого будет разрешен доступ, задается строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# snmp-server community public rw
```

### 28.1.5 *snmp-server filter*

Данная команда устанавливает фильтрацию на отправляемые SNMP уведомления, запрещая отправку указанных.

Использование отрицательной формы команды (no) отменяет фильтрацию указанных типов уведомлений.

## Синтаксис

```
[no] snmp-server filter { enviroment <ENV> | links <LINK> }
```

## Параметры

- enviroment – запрещает отправку всех уведомлений об изменении параметров окружения, если не указан ни один из конкретизирующих атрибутов;
- links - запрещает отправку всех уведомлений об изменениях состояния линка, если не указан ни один из конкретизирующих атрибутов.

<ENV> – типы фильтров параметров окружения:

- pwrin – отказ БП;
- pwrin-insert – БП установлен;
- pwrin-remove – отсутствие БП;
- fan – отказ вентилятора;
- fan-speed – изменение скорости вентиляторов;
- low-space - маленький объем NAND;
- cpu-load – высокая нагрузка ЦПУ;
- low-ram – маленький объем RAM;
- cpu-temp – высокая температура ЦПУ;
- switch-temp - высокая температура коммутаторного процессора;
- sensor-temp – высокое значение других датчиков температуры.

<LINK>— типы фильтров состояние порта:

- status – состояние порта;
- flapping – быстрая смена состояний порта .

### Значение по умолчанию

61

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# snmp-server filter pwrin
```

## 28.1.6 *snmp-server dscp*

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов SNMP-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

### Синтаксис

snmp-server dscp <DSCP>

no snmp-server dscp

### Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

### Значение по умолчанию

61

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

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

## 28.1.7 *snmp-server user*

Данной командой создается SNMPv3-пользователь.

Использование отрицательной формы команды (no) удаляет SNMPv3-пользователя.

### Синтаксис

[no] snmp-server user <NAME>

## Параметры

<NAME> – имя пользователя, задаётся строкой до 63 символов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# snmp-server user admin  
esr(snmp-user)#
```

### 28.1.8 *access*

Данной командой определяется уровень доступа по протоколу SNMPv3.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

access <TYPE>

no access

## Параметры

<TYPE> – уровень доступа:

ro – доступ только для чтения;

rw – доступ для чтения и записи.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SNMP-USER

## Пример

```
esr(snmp-user)# access rw
```

### 28.1.9 *authentication access*

Данной командой определяется режим безопасности.

Использование отрицательной формы команды (no) отключает аутентификацию.

## Синтаксис

authentication access <TYPE>

no authentication access

## Параметры

<TYPE> – режим безопасности:

auth – используется только аутентификация;

priv – используется аутентификация и шифрование данных.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SNMP-USER

### Пример

```
esr(snmp-user)# authentication algorithm auth
```

## 28.1.10 authentication algorithm

Данная команда определяет алгоритм аутентификации SNMPv3-запросов.

Использование отрицательной формы команды (no) отключает аутентификацию.

### Синтаксис

authentication algorithm <ALGORITHM>

no authentication algorithm

### Параметры

<ALGORITHM> – алгоритм шифрования:

md5 – пароль шифруется по алгоритму md5;

sha1 – пароль шифруется по алгоритму sha1.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SNMP-USER

### Пример

```
esr(snmp-user)# authentication algorithm md5
```

## 28.1.11 authentication key

Данная команда устанавливает пароль для аутентификации SNMPv3-запросов.

Использование отрицательной формы команды (no) удаляет пароль.

### Синтаксис

authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }

no authentication key

### Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов;

encrypted – при указании команды задается зашифрованный пароль:

ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SNMP-USER

### Пример

```
esr (snmp-user) # authentication key ascii-text 123456789
esr (snmp-user) # authentication key ascii-text encrypted CDE65039E5591FA3F1
```

## 28.1.12 *enable*

Данной командой активируется SNMPv3-пользователь.

Использование отрицательной формы команды (no) деактивирует SNMPv3-пользователя.

### Синтаксис

[no] enable

### Параметры

Команда не содержит аргументов.

### Значение по умолчанию

Процесс выключен.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-SNMP-USER

### Пример

```
esr (snmp-user) # enable
```

## 28.1.13 *privacy algorithm*

Данная команда определяет алгоритм шифрования передаваемых данных.

Использование отрицательной формы команды (no) отключает шифрование.

### Синтаксис

privacy algorithm <ALGORITHM>

no privacy algorithm

### Параметры

<ALGORITHM> – алгоритм шифрования:

aes128 – использовать алгоритм шифрования AES-128;

des – использовать алгоритм шифрования DES.

---

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SNMP-USER

## Пример

```
esr(snmp-user)# privacy algorithm des
```

---

### 28.1.14 *privacy key*

---

Данная команда устанавливает пароль для шифрования передаваемых данных.

Использование отрицательной формы команды (no) удаляет пароль.

## Синтаксис

privacy key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }

no privacy key

## Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-SNMP-USER

## Пример

```
esr(snmp-user)# privacy key ascii-text 123456789
esr(snmp-user)# privacy key ascii-text encrypted CDE65039E5591FA3F1
```



## 28.2 Управление SYSLOG

---

### 28.2.1 *syslog console*

---

Данной командой устанавливаются уровни syslog-сообщений, которые будут отображаться в консоли. Отображаются сообщения, имеющие уровень важности, заданный в команде или более высокий.

Использование отрицательной формы команды (no) устанавливает уровень отображаемых сообщений по умолчанию.

#### Синтаксис

syslog console <SEVERITY>

no syslog console

#### Параметры

<SEVERITY> – уровень важности сообщения, принимает значения (в порядке убывания важности):

emerg – в системе произошла критическая ошибка, система неработоспособна;

alert – сигналы тревоги, необходимо немедленное вмешательство персонала;

crit – критическое состояние системы, сообщение о событии;

error – сообщения об ошибках;

warning – предупреждения, неаварийные сообщения;

notice – сообщения о важных системных событиях;

info – информационные сообщения системы;

debug – отладочные сообщения, предоставляют пользователю информацию для корректной настройки системы;

none – отключает вывод syslog-сообщений в консоль.

#### Значение по умолчанию

info

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

```
esr(config)# syslog console info
```

---

## **28.2.2      *syslog monitor***

---

Данной командой устанавливается уровень syslog-сообщений, которые будут отображаться при удаленных подключениях (Telnet, SSH). Отображаются сообщения, имеющие уровень важности, заданный в команде или более высокий.

Использование отрицательной формы команды (no) устанавливает уровень отображаемых сообщений по умолчанию.

### **Синтаксис**

syslog monitor <SEVERITY>

no syslog monitor

### **Параметры**

<SEVERITY> – уровень важности сообщения, возможные значения приведены в разделе 28.2.1.

### **Значение по умолчанию**

info

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG

### **Пример**

```
esr(config)# syslog monitor info
```

---

## **28.2.3      *syslog cli-commands***

---

Данной командой включается процесс логирования введенных команд пользователя на локальный syslog-сервер.

Использование отрицательной формы команды (no) выключает логирование команд.

### **Синтаксис**

[no] syslog cli-commands

### **Параметры**

Команда не содержит аргументов.

### **Необходимый уровень привилегий**

10

### **Командный режим**

CONFIG

### **Пример**

```
esr(config)# syslog cli-commands
```

---

## 28.2.4 *syslog file*

---

Данной командой включается сохранение сообщений syslog заданного уровня важности в указанный файл журнала. Сохраняются сообщения, имеющие уровень важности, заданный в команде, или более высокий.

Использование отрицательной формы команды (no) отключает сохранение сообщений syslog в указанный файл.

### Синтаксис

syslog file <NAME> <SEVERITY>

no syslog file <NAME>

### Параметры

<NAME> – имя файла, в который будет производиться запись сообщений заданного уровня, задается строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут отключено сохранение во все сконфигурированные syslog-файлы;

<SEVERITY> – уровень важности сообщения, возможные значения приведены в разделе 28.2.1.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# syslog file esr info
```

---

## 28.2.5 *syslog file-size*

---

Командой устанавливается максимальный размер файла журнала. По превышении указанного размера будет автоматически производиться ротация файлов.

Использование отрицательной формы команды (no) устанавливает значение размера файла журнала в значение по умолчанию.

### Синтаксис

syslog file-size <SIZE>

no syslog file-size

### Параметры

<SIZE> – размер файла, принимает значение [10..10000000] кбайт.

### Значение по умолчанию

500 кбайт

### Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример

```
esr(config)# syslog file-size 10000
```

## 28.2.6 *syslog max-files*

Данная команда устанавливает максимальное количество файлов, сохраняемых при ротации.

Использование отрицательной формы команды (no) устанавливает количество хранимых файлов журнала в значение по умолчанию.

### Синтаксис

syslog max-files <NUM>

no syslog max-files

### Параметры

<NUM> – максимальное количество файлов, принимает значения [1 .. 1000].

### Значение по умолчанию

1

### Необходимый уровень привилегий

10

## Командный режим

CONFIG

### Пример

```
esr(config)# syslog max-files 100
```

## 28.2.7 *syslog host*

Данной командой включается передача сообщений syslog заданного уровня важности на удаленный syslog-сервер. Передаются сообщения, имеющие уровень важности, заданный в команде или более высокий.

Использование отрицательной формы команды (no) отключает передачу сообщений syslog на удаленный syslog-сервер.

### Синтаксис

syslog host <HOSTNAME> { <ADDR> | <IPV6-ADDR> } [ <SEVERITY> ] [ <TRANSPORT> ] [ <PORT> ]  
[ vrf <VRF> ]

no syslog host <HOSTNAME>

### Параметры

<HOSTNAME> – наименование syslog-сервера, задаётся строкой до 31 символа. Используется только для идентификации сервера при конфигурировании. Значение «all» используется в команде **no syslog host** для удаления всех syslog-серверов;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<SEVERITY> – уровень важности сообщения, опциональный параметр, возможные значения приведены в разделе 28.2.1;

<TRANSPORT> – протокол передачи данных, опциональный параметр, принимает значения:

TCP – передача данных осуществляется по протоколу TCP;

UDP – передача данных осуществляется по протоколу UDP;

<PORT> – номер TCP/UDP-порта, опциональный параметр, принимает значения [1..65535], по умолчанию 514;

<VRF> – имя экземпляра VRF, в котором находится удаленный syslog-сервер, задается строкой до 31 символа.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# syslog host eltex 192.168.2.2
```

## 28.2.8 *clear log*

Команда позволяет удалить log-файлы, хранимые в локальной памяти устройства.

### Синтаксис

```
clear log [ file [ <FILE> ] ]
```

### Параметры

file – при использовании команды без параметров удаляются все файлы;

<FILE> – имя или список имён log-файлов для удаления, опциональный параметр.

При вызове команды без параметра будут удалены log-файлы, которые не входят в текущую ротацию Syslog-сервера.

### Необходимый уровень привилегий

10

### Командный режим

ROOT

### Пример

```
esr# clear log file esr.2
```

Команда для просмотра текущей информации о конфигурации syslog-журнала, списка созданных log-файлов, а также для просмотра log-файлов с возможностью фильтрации с помощью регулярных выражений.

### Синтаксис

```
show syslog [ { configuration | <FILE> [ grep <TEXT> ] } ]
```

### Параметры

<FILE> – имя файла, задаётся строкой до 31 символа;

grep <TEXT> – фильтрация файла по указанному регулярному выражению, задаётся строкой до 31 символа;

configuration – команда, при указании которой выводится информация о конфигурации syslog-журнала.

При выполнении команды без параметров будет отображена информация о созданных log-файлах.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show syslog

  Log files
  ~~~~~~
Name Size in bytes Date of last modification
---- -
1 debug 371681 Thu Jan 1 16:17:04 1970
2 debug.1 524222 Thu Jan 1 01:48:13 1970
3 esr 97259 Thu Jan 1 16:17:01 1970
---- -
Total files: 4

esr# show syslog configuration

SYSLOG

File size: 512 (kiB)
Number of logs: 3
Console: info

 Files:
  ~~~~~~
ID      Name                      Severity
--  -
0      esr                        info
```

---

## 28.3 Настройка доступа SSH, Telnet

---

### 28.3.1 *ip ssh server*

---

Данной командой включается SSH-сервер на маршрутизаторе.

Использование отрицательной формы команды (no) отключает SSH-сервер.

#### Синтаксис

```
[no] ip ssh server [ vrf <VRF>]
```

#### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать SSH-сервер.

#### Значение по умолчанию

SSH-сервер выключен.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

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

### 28.3.2 *ip ssh port*

---

Данной командой определяется порт SSH-сервера на маршрутизаторе.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
ip ssh port <PORT>
```

```
no ip ssh port
```

#### Параметры

<PORT> – номер порта, указывается в диапазоне [1..65535].

#### Значение по умолчанию

22

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

```
esr(config)# ip ssh port 3001
```

---

### **28.3.3**      *ip ssh dscp*

---

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов SSH сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

#### **Синтаксис**

ip ssh dscp <DSCP>

no ip ssh dscp

#### **Параметры**

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

#### **Значение по умолчанию**

32

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG

#### **Пример**

```
esr(config)# ip ssh dscp 40
```

---

### **28.3.4**      *ip ssh client username*

---

Данной командой определяется имя пользователя по умолчанию для операций копирования по протоколу SCP.

Использование отрицательной формы команды (no) удаляет имя пользователя.

#### **Синтаксис**

ip ssh client username <NAME>

no ssh client username

#### **Параметры**

<NAME> – имя пользователя, задаётся строкой до 31 символа.

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

CONFIG

#### **Пример**

```
esr(config)# ip ssh client username tester
```



### 28.3.5 *ip ssh client password*

Данной командой определяется пароль по умолчанию для операций копирования по протоколу SCP.

Использование отрицательной формы команды (no) удаляет пароль.

#### Синтаксис

```
ip ssh client password { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no ssh client password
```

#### Параметры

<CLEAR-TEXT> – пароль, задаётся строкой [1 .. 16] символов, принимает значения [0-9a-fA-F];

<ENCRYPTED-TEXT > – зашифрованный пароль, задаётся строкой [2..32] символов.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

```
esr(config)# ip ssh client password test132
```

### 28.3.6 *crypto key generate*

Данной командой генерируется новый криптографический ключ для установления соединения по протоколу SSH.

#### Синтаксис

```
crypto key generate <OPTIONS>
```

#### Параметры

<OPTIONS> – алгоритм генерации нового криптографического ключа:

- dsa – алгоритм DSA;
- ecdsa – алгоритм ECDSA;
- ed25519 – алгоритм ED25519;
- rsa – алгоритм RSA.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

```
esr(config)# crypto key generate ecdsa
```

---

## 28.3.7 *show crypto key mypubkey*

---

Команда используется для просмотра открытых ключей устройства, используемых при установлении соединения по протоколу SSH.

### Синтаксис

show crypto key mypubkey <OPTIONS>

### Параметры

<OPTIONS> – алгоритм генерации нового криптографического ключа:

- dsa – алгоритм DSA;
- ecdsa – алгоритм ECDSA;
- ed25519 – алгоритм ED25519;
- rsa – алгоритм RSA.

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show crypto key mypubkey rsa
Key data
-----
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDz750sWCQrnNufg1yhukstFYCYdEfg
JZ9tWUvcssAZhCJWMewprXBuZMABzFmfBg157pgapxn2qjXJ8ESMV7X7gPfy
xQQah6l376z3SFcpKvwudNgwHiS5HCYPRQWx2Xdaz/nJtYr5NpYgLPba68NC
iXcqEp7EPR5GojDVxpuDuk0hPFcihzmt5Yx8ZptJRzRtsuDQYlowv0Qa24kd
OlQ90/1qKfbAhB6XI60l+dK5VEj7giBESarcRn69/e/YVbdGBdTE93QWFPKI
bm63imfbxRwWtcwsFdIH8Blv9ZqDqqF/IO3TkIKa3lhV9GnsawlAXi/IdyY
bYPboHRdcTlH/ root@esr-1000
```

---

## 28.3.8 *ip telnet server*

---

Данной командой включается Telnet-сервер на маршрутизаторе.

Использование отрицательной формы команды (no) отключает Telnet-сервер.

### Синтаксис

[no] ip telnet server [vrf <VRF>]

### Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать Telnet-сервер.

### Значение по умолчанию

Telnet-сервер выключен.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

## Пример

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

### 28.3.9 *ip telnet port*

Данной командой определяется порт Telnet-сервера на маршрутизаторе.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
ip telnet port <PORT>
```

```
no ip telnet port
```

#### Параметры

<PORT> – номер порта, принимает значения [1..65535].

#### Значение по умолчанию

23

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

## Пример

```
esr(config)# ip telnet port 2001
```

### 28.3.10 *ip telnet dscp*

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов Telnet-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

#### Синтаксис

```
ip telnet dscp <DSCP>
```

```
no ip telnet dscp
```

#### Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

#### Значение по умолчанию

32

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

## Пример

```
esr(config)# ip telnet dscp 40
```

### **28.3.11**      *ip ftp client username*

Данной командой определяется имя пользователя по умолчанию для операций копирования по протоколу FTP.

Использование отрицательной формы команды (no) удаляет имя пользователя.

#### Синтаксис

```
ip ftp client username <NAME>
```

```
no ftp client username
```

#### Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

## Пример

```
esr(config)# ip ftp client username test
```

### **28.3.12**      *ip ftp client password*

Данной командой определяется пароль по умолчанию для операций копирования по протоколу FTP.

Использование отрицательной формы команды (no) удаляет пароль.

#### Синтаксис

```
ip ftp client password { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
```

```
[no] ftp client password
```

#### Параметры

<CLEAR-TEXT> – пароль, задаётся строкой [1 .. 16] символов, принимает значения [0-9a-fA-F];

<ENCRYPTED-TEXT> – зашифрованный пароль, задаётся строкой [2..32] символов.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

## Пример

```
esr(config)# ip ftp client password test
```



## 29 НАСТРОЙКА DHCP

### 29.1 Управление DHCP-клиентом

---

#### 29.1.1 *ip address dhcp*

---

Данной командой включается получение динамического IP-адреса конфигурируемого интерфейса по протоколу DHCP.

Использование отрицательной формы команды (no) выключает получение динамического IP-адреса по протоколу DHCP.

##### Синтаксис

[no] ip address dhcp

##### Параметры

Команда не содержит аргументов.

##### Значение по умолчанию

Выключен.

##### Необходимый уровень привилегий

10

##### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

##### Пример

```
esr(config-if-gi)# ip address dhcp
```

#### 29.1.2 *ip dhcp server address*

---

Данной командой устанавливается IP-адрес DHCP-сервера, у которого будет запрашиваться IP-адрес.

Использование отрицательной формы команды (no) удаляет установленный IP-адрес DHCP-сервера.

##### Синтаксис

ip dhcp server address <ADDR>

no ip dhcp server address

##### Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

##### Необходимый уровень привилегий

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ip dhcp server address 10.10.10.1
```

### 29.1.3 *ip dhcp client lease-time*

Данной командой устанавливается запрашиваемое время аренды сетевого адреса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

ip dhcp client lease-time <TIME>

no ip dhcp client lease-time

## Параметры

<TIME> – запрашиваемое время аренды, задаётся в виде DD:HH:MM, где:

DD – дни, принимает значение [0..364];

HH – часы, принимает значение [0..23];

MM – минуты, принимает значение [0..59].

## Значение по умолчанию

1 день

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ip dhcp client lease-time 00:12:00
```

### 29.1.4 *ip dhcp client timeout*

Данной командой задаётся интервал, по истечении которого клиент считает, что DHCP-сервер недоступен. Если в базе данных IP-адресов клиента есть какие-либо арендованные адреса, срок аренды которых еще не истек, то клиент будет проверять последовательно каждый из них и, если

найдет корректную, то IP-адрес из неё будет присвоен интерфейсу. Если нет действующих аренд в базе данных, то клиент будет повторно запрашивать IP-адрес по истечении интервала повтора (dhcp retry). Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
ip dhcp client timeout <SEC>
no ip dhcp client timeout
```

#### Параметры

<SEC> – период времени в секундах, принимает значения [1 .. 600].

#### Значение по умолчанию

60 секунд

#### Необходимый уровень привилегий

10

#### Командный режим

```
CONFIG-GI
CONFIG-SUBIF
CONFIG-PORT-CHANNEL
CONFIG-BRIDGE
```

#### Пример

```
esr(config-if-gi)# ip dhcp client timeout 300
```

### 29.1.5 *ip dhcp client retry*

Данной командой задаётся интервал, через который DHCP-клиент возобновит попытки получить IP-адрес, если было установлено, что DHCP-сервер не отвечает.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
ip dhcp client retry <SEC>
no ip dhcp client retry
```

#### Параметры

<SEC> – период времени в секундах, принимает значение [1..600].

#### Значение по умолчанию

300 секунд

#### Необходимый уровень привилегий

10

#### Командный режим

```
CONFIG-GI
CONFIG-SUBIF
CONFIG-PORT-CHANNEL
```



## Пример

```
esr(config-if-gi)# ip dhcp client retry 180
```

### 29.1.6 *ip dhcp client reboot*

---

Данной командой задаётся время, в течение которого DHCP-клиент будет пытаться получить старый IP-адрес, перед тем как начать получать новый.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
ip dhcp client reboot <SEC>
```

```
no ip dhcp client reboot
```

#### Параметры

<SEC> – период времени в секундах, принимает значение [1..600].

#### Значение по умолчанию

10 секунд

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ip dhcp client reboot 60
```

### 29.1.7 *ip dhcp client select-timeout*

---

Данной командой задаётся период времени, в течение которого DHCP-клиент будет выбирать среди предложений по аренде от серверов, если такие существуют. Это используется в сетях с несколькими DHCP-серверами – в этом случае клиенту в ответ на запрос IP-адреса может быть отправлено несколько предложений. Возможно, что одно из этих предложений предпочтительнее другого (например, одно предложение может иметь адрес, который клиент использовал ранее).

Клиент ждет указанный период времени после того, как отправил запрос на получение IP-адреса, предполагая, что он получит несколько предложений от сервера. Если дополнительных предложений за указанный период времени не поступало, то клиент принимает одно из предложений.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

```
ip dhcp client select-timeout <SEC>
```

```
no ip dhcp client select-timeout
```

## Параметры

<SEC> – период времени в секундах, принимает значение [0..300].

## Значение по умолчанию

0 секунд – клиент примет первое пришедшее предложение.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ip dhcp client select-timeout 30
```

## 29.1.8 *ip dhcp client vendor-class-id*

Данной командой устанавливается значение DHCP Опции 60 для получения дополнительных настроек по DHCP Опции 43.

Использование отрицательной формы команды (no) отключает запрос данной опции.

## Синтаксис

ip dhcp client vendor-class-id <NAME>

no ip dhcp client vendor-class-id

## Параметры

<NAME> – идентификатор класса поставщика, задаётся строкой до 31 символа.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ip dhcp client vendor-class-id ELTEX
```

## 29.1.9 *ip dhcp client ignore*

Данной командой указываются DHCP опции, которые будут игнорироваться клиентом.

## Синтаксис

```
ip dhcp client ignore <OPTION>
```

```
no ip dhcp client ignore
```

## Параметры

<OPTION> – принимает следующие значения:

dns-nameserver – DHCP-опция 6, Список DNS-серверов;

domain-name – DHCP-опция 15, Имя домена;

netbios-nameserver – DHCP-опция 44, Список NetBios-серверов;

router – DHCP-опция 3, Список шлюзов по умолчанию;

static-route – DHCP-опция 121, Список бесклассовых статических маршрутов;

tftp-server-address – DHCP-опция 66, Имя TFTP-сервера;

vendor-specific – DHCP-опция 43, Информация определенная производителем.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ip dhcp client ignore router
```

## 29.2 Управление IPv6 DHCP-клиентом

### 29.2.1 *ipv6 address dhcp*

Данной командой включается получение динамического IPv6-адреса конфигурируемого интерфейса по протоколу IPv6 DHCP.

Использование отрицательной формы команды (no) выключает получение динамического IPv6-адреса по протоколу IPv6 DHCP.

## Синтаксис

```
[no] ipv6 address dhcp
```

## Параметры

Команда не содержит аргументов.

## Значение по умолчанию

Выключен.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ipv6 address dhcp
```

### 29.2.2 *ipv6 dhcp server address*

Данной командой устанавливается IPv6-адрес DHCP-сервера, у которого будет запрашиваться IPv6-адрес.

Использование отрицательной формы команды (no) удаляет установленный IPv6-адрес DHCP-сервера.

## Синтаксис

ipv6 dhcp server address <IPv6-ADDR>

no ipv6 dhcp server address

## Параметры

<IPv6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ipv6 dhcp server address fc00::1
```

### 29.2.3 *ipv6 dhcp client lease-time*

Данной командой устанавливается запрашиваемое время аренды сетевого адреса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

ipv6 dhcp client lease-time <TIME>

no ipv6 dhcp client lease-time

## Параметры

<TIME> – запрашиваемое время аренды, задаётся в виде DD:HH:MM, где:

DD – дни, принимает значение [0..364];

HH – часы, принимает значение [0..23];

MM – минуты, принимает значение [0..59].

## Значение по умолчанию

1 день.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ipv6 dhcp client lease-time 00:12:00
```

## 29.2.4 *ipv6 dhcp client timeout*

Данной командой задаётся интервал, по истечении которого клиент считает, что IPv6 DHCP-сервер недоступен. Если в базе данных IPv6-адресов клиента есть какие-либо арендованные адреса, срок аренды которых ещё не истек, то клиент будет проверять последовательно каждый из них и, если найдет корректную, то IPv6-адрес из неё будет присвоен интерфейсу. Если нет действующих аренд в базе данных, то клиент будет повторно запрашивать IPv6-адрес по истечении интервала повтора (dhcp retry).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

## Синтаксис

ipv6 dhcp client timeout <SEC>

no ipv6 dhcp client timeout

## Параметры

<SEC> – период времени в секундах, принимает значения [1 .. 600].

## Значение по умолчанию

60 секунд.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

---

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# ipv6 dhcp client timeout 300
```

## 29.2.5 *ipv6 dhcp client retry*

---

Данной командой задаётся интервал, через который IPv6 DHCP-клиент возобновит попытки получить IPv6-адрес, если было установлено, что IPv6 DHCP-сервер не отвечает.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

ipv6 dhcp client retry <SEC>

no ipv6 dhcp client retry

### Параметры

<SEC> – период времени в секундах, принимает значение [1..600].

### Значение по умолчанию

300 секунд.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

### Пример

```
esr(config-if-gi)# ipv6 dhcp client retry 180
```

## 29.2.6 *ipv6 dhcp client reboot*

---

Данной командой задаётся время, в течение которого IPv6 DHCP-клиент будет пытаться получить старый IPv6-адрес перед тем, как начать получать новый.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

### Синтаксис

ipv6 dhcp client reboot <SEC>

no ipv6 dhcp client reboot

### Параметры

<SEC> – период времени в секундах, принимает значение [1..600].

### Значение по умолчанию

10 секунд

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

#### Пример

```
esr(config-if-gi)# ipv6 dhcp client reboot 60
```

### 29.2.7 *ipv6 dhcp client select-timeout*

Данной командой задаётся период времени, в течение которого IPv6 DHCP-клиент будет выбирать среди предложений по аренде от серверов, если такие существуют. Это используется в сетях с несколькими IPv6 DHCP-серверами – в этом случае клиенту в ответ на запрос IPv6-адреса может быть отправлено несколько предложений. Возможно, что одно из этих предложений предпочтительнее другого (например, одно предложение может иметь адрес, который клиент использовал ранее).

Клиент ждет указанный период времени после того, как отправил запрос на получение IPv6-адреса, предполагая, что он получит несколько предложений от сервера. Если дополнительных предложений за указанный период времени не поступало, то клиент принимает одно из предложений.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

#### Синтаксис

ipv6 dhcp client select-timeout <SEC>

no ipv6 dhcp client select-timeout

#### Параметры

<SEC> – период времени в секундах, принимает значение [0..300].

#### Значение по умолчанию

0 секунд – клиент примет первое пришедшее предложение.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

#### Пример

```
esr(config-if-gi)# ipv6 dhcp client select-timeout 30
```

### **29.2.8**      ***ipv6 dhcp client vendor-class-id***

---

Данной командой устанавливается значение DHCP Опции 60 для получения дополнительных настроек по DHCP Опции 43.

Использование отрицательной формы команды (no) отключает запрос данной опции.

#### **Синтаксис**

```
ipv6 dhcp client vendor-class-id <NAME>  
no ipv6 dhcp client vendor-class-id
```

#### **Параметры**

<NAME> – идентификатор класса поставщика, задаётся строкой до 31 символа.

#### **Необходимый уровень привилегий**

10

#### **Командный режим**

```
CONFIG-GI  
CONFIG-SUBIF  
CONFIG-PORT-CHANNEL  
CONFIG-BRIDGE
```

#### **Пример**

```
esr(config-if-gi)# ipv6 dhcp client vendor-class-id ELTEX
```

### **29.2.9**      ***ipv6 dhcp client ignore***

---

Данной командой указываются DHCP-опции, которые будут игнорироваться клиентом.

#### **Синтаксис**

```
ipv6 dhcp client ignore <OPTION>  
no ipv6 dhcp client ignore
```

#### **Параметры**

<OPTION> – принимает следующие значения:

dns-nameserver – DHCP-опция 6, Список DNS-серверов;

domain-name – DHCP-опция 15, Имя домена;

netbios-nameserver – DHCP-опция 44, Список NetBios-серверов;

static-route – DHCP-опция 121, Список бесклассовых статических маршрутов;

tftp-server-address – DHCP-опция 66, Имя TFTP-сервера;

vendor-specific – DHCP-опция 43, Информация, определенная производителем.



## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

## Пример

```
esr(config-if-gi)# ipv6 dhcp client ignore dns-nameserver
```

## 29.3 Управление DHCP Relay агентом

### 29.3.1 *ip dhcp-relay*

Данная команда включает DHCP-агент.

Использование отрицательной формы команды (no) выключает DHCP-агент.

#### Синтаксис

[no] ip dhcp-relay

#### Параметры

Команда не содержит аргументов.

#### Значение по умолчанию

Выключен

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ip dhcp-relay
```

### 29.3.2 *ip helper-address*

Данной командой указывается IP-адрес DHCP-сервера, которому будут отправляться DHCP Discover пакеты, перехваченные DHCP Relay агентом.

Использование отрицательной формы команды (no) удаляет IP-адрес из списка DHCP-серверов для DHCP Relay агента.

#### Синтаксис

ip helper-address <IP>

no ip helper-address

## Параметры

<IP> – IP-адрес DHCP-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно указать до 8 IP-адресов, список задаётся через запятую.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

## Пример

```
esr(config-if-gi)# ip helper-address 10.10.10.1
```

## 29.4 Управление IPv6 DHCP Relay агентом

### 29.4.1 *ipv6 dhcp-relay*

Данная команда включает IPv6 DHCP-агент.

Использование отрицательной формы команды (no) выключает IPv6 DHCP-агент.

## Синтаксис

[no] ipv6 dhcp-relay

## Параметры

Команда не содержит аргументов.

## Значение по умолчанию

Выключен

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ipv6 dhcp-relay
```

### 29.4.2 *ipv6 dhcp-relay interface*

Данной командой указывается исходящий интерфейс и IPv6-адрес DHCP-сервера, которому будут отправляться DHCP Solicit-пакеты, перехваченные IPv6 DHCP Relay-агентом.

Использование отрицательной формы команды (no) удаляет исходящий интерфейс из списка интерфейсов для IPv6 DHCP Relay агента.

#### Синтаксис

```
ipv6 dhcp-relay interface <IF> [ <IPV6-ADDR> ]
```

```
no ipv6 dhcp-relay interface <IF>
```

#### Параметры

<IF> – наименование интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе 3.3.

<IPV6-ADDR> – IPv6-адрес DHCP-сервера, задаётся в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Если адрес не задан, то пакеты отправляются на ff02::1:2. Все IPv6 DHCP-сервера и агенты в локальном сетевом сегменте.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG-GI

CONFIG-SUBIF

CONFIG-PORT-CHANNEL

CONFIG-BRIDGE

#### Пример

```
esr(config-if-gi)# ipv6 dhcp-relay interface gigabitethernet 1/0/2 fc00::1
```

## 29.5 Настройка и мониторинг DHCP-сервера

### 29.5.1 *ip dhcp-server*

Данная команда включает DHCP-сервер.

Использование отрицательной формы команды (no) выключает DHCP-сервер.

#### Синтаксис

```
[no] ip dhcp-server
```

#### Параметры

Команда не содержит аргументов.

#### Значение по умолчанию

Выключен

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

### 29.5.2 *ip dhcp-server dscp*

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов DHCP-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

#### Синтаксис

```
ip dhcp-server dscp <DSCP>
no ip dhcp-server dscp
```

#### Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

#### Значение по умолчанию

61

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

```
esr(config)# ip dhcp-server dscp 40
```

### 29.5.3 *ip dhcp-server pool*

Команда используется для создания пула IP-адресов DHCP-сервера и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный пул IP-адресов.

#### Синтаксис

```
[no] ip dhcp-server pool <NAME>
```

#### Параметры

<NAME> – имя пула IP-адресов DHCP-сервера, задаётся строка до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все пулы IP-адресов.

#### Необходимый уровень привилегий

10

#### Командный режим

CONFIG

#### Пример

```
esr(config)# ip dhcp-server pool lan
```

## 29.5.4 *network*

Данная команда задает IP-адрес и маску для подсети, из которой будет выделен пул IP-адресов. Использование отрицательной формы команды (no) удаляет настройки подсети в пуле.

### Синтаксис

```
network <ADDR/LEN>
```

```
no network
```

### Параметры

<ADDR/LEN> – IP-адрес и префикс подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DHCP-SERVER

### Пример

```
esr(config-dhcp-server)# network 192.168.3.0/24
```

## 29.5.5 *address-range*

Данная команда позволяет добавить диапазон IP-адресов к пулу адресов, конфигурируемого DHCP-сервера.

Использование отрицательной формы команды (no) удаляет указанный диапазон IP-адресов.

### Синтаксис

```
address-range <FROM-ADDR>-<TO-ADDR>
```

```
no address-range { <FROM-ADDR>-<TO-ADDR> | all }
```

### Параметры

<FROM-ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Можно указать до 32 диапазонов IP-адресов, список задаётся через запятую.

all – удалить все сконфигурированные диапазоны IP-адресов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DHCP-SERVER

### Пример

```
esr(config-dhcp-server)# address-range 192.168.3.1-192.168.3.20,192.168.3.24
```

## 29.5.6 *address*

---

Данная команда позволяет добавить IP-адрес для определенного физического адреса к пулу адресов конфигурируемого DHCP-сервера.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес.

### Синтаксис

address <ADDR> mac-address <MAC>

no address <ADDR>

### Параметры

<ADDR> – IP-адрес клиента, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Если использовать команду для удаления, то при указании значения «all» будут удалены все IP-адреса;

<MAC> – MAC-адрес клиента, которому будет выдан IP-адрес, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DHCP-SERVER

### Пример

```
esr(config-dhcp-server)# address 192.168.3.21 mac-address A8:F9:4B:AA:00:40
```

## 29.5.7 *default-router*

---

Данная команда позволяет задать список IP-адресов шлюзов по умолчанию, которые DHCP-сервер будет сообщать клиентам, используя DHCP опцию 3.

Использование отрицательной формы команды (no) удаляет указанные адреса из списка шлюзов.

### Синтаксис

[no] default-router <ADDR>

### Параметры

<ADDR> – IP-адрес шлюза по умолчанию, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно указать до 8 IP-адресов, список задаётся через запятую. Если использовать команду для удаления, то при указании значения «all» будут удалены все шлюзы по умолчанию.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DHCP-SERVER

### Пример

```
esr(config-dhcp-server)# default-router 192.168.3.1,192.168.3.2
```

## 29.5.8 *domain-name*

Данная команда позволяет задать DNS-имя сетевого домена. Имя домена передаётся клиентам в составе DHCP-опций.

Использование отрицательной формы команды (no) удаляет установленное имя домена.

### Синтаксис

domain-name <NAME>

no domain-name

### Параметры

<NAME> – DNS-имя домена клиента, задаётся строкой до 255 символов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DHCP-SERVER

### Пример

```
esr(config-dhcp-server)# domain-name eltex.loc
```

## 29.5.9 *dns-server*

Данная команда позволяет задать список IP-адресов DNS-серверов. Список передаётся клиентам в составе DHCP-опций.

Использование отрицательной формы команды удаляет указанный DNS-сервер из списка.

### Синтаксис

dns-server <ADDR>

no dns-server { <ADDR> | all }

### Параметры

<ADDR> – IP-адрес DNS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно указать до 8 IP-адресов, список задаётся через запятую. Если использовать команду для удаления, то при указании значения «all» будут удалены все DNS-серверы.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DHCP-SERVER

### Пример

```
esr(config-dhcp-server)# dns-server 8.8.8.8,8.8.4.4
```

---

## 29.5.10 *max-lease-time*

---

Данная команда позволяет задать максимальное время аренды IP-адресов. Если DHCP-клиент запрашивает время аренды, превосходящее максимальное значение, то будет установлено время, заданное этой командой. Использование отрицательной формы команды устанавливает значение по умолчанию.

### Синтаксис

max-lease-time <TIME>

no max-lease-time

### Параметры

<TIME> – максимальное время аренды IP-адреса, задаётся в формате DD:HH:MM, где:

DD – количество дней, принимает значения [0..364];

HH – количество часов, принимает значения [0..23];

MM – количество минут, принимает значения [0..59].

### Значение по умолчанию

1 день

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-DHCP-SERVER

### Пример

```
esr(config-dhcp-server)# max-lease-time 00:16:00
```

---

## 29.5.11 *default-lease-time*

---

Данная команда позволяет задать время аренды, на которое клиенту будет выдан IP-адрес, если клиент не запрашивал определенное время аренды. Использование отрицательной формы команды устанавливает значение по умолчанию.

### Синтаксис

default-lease-time <TIME>

no default-lease-time

### Параметры

<TIME> – время аренды IP-адреса, в формате DD:HH:MM, где:

DD – количество дней, принимает значения [0..364];

HH – количество часов, принимает значения [0..23];

MM – количество минут, принимает значения [0..59].

### Значение по умолчанию

12 часов



## Необходимый уровень привилегий

10

## Командный режим

CONFIG-DHCP-SERVER

## Пример

```
esr(config-dhcp-server)# default-lease-time 00:04:00
```

### 29.5.12 *ip dhcp-server vendor-class-id*

Данная команда необходима для создания идентификатора класса поставщика (DHCP Опция 60) и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный идентификатор класса поставщика.

## Синтаксис

[no] ip dhcp-server vendor-class-id <NAME>

## Параметры

<NAME> – идентификатор класса поставщика, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все идентификаторы класса поставщика.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ip dhcp-server vendor-class-id ELTEX
```

### 29.5.13 *vendor-specific-options*

Данная команда позволяет задать специфическую информацию поставщика (DHCP Опция 43). Использование отрицательной формы команды (no) удаляет специфическую информацию поставщика.

## Синтаксис

vendor-specific-options <HEX>  
no vendor-specific-options

## Параметры

<HEX> – специфическая информация поставщика, задаётся в шестнадцатеричном формате до 128 символов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-DHCP-VENDOR-ID

## Пример

```
esr(config-dhcp-vendor-id)# vendor-specific-options 0b0931302e312e39302e320
```

### 29.5.14 netbios-name-server

Данная команда позволяет сконфигурировать 44 опцию DHCP (задает IP-адрес NetBIOS-сервера).

Использование отрицательной формы команды (no) выключает передачу IP-адреса NetBIOS-сервера (44 опцию).

#### Синтаксис

```
no netbios-name-server <ADDR>
no netbios-name-server { <ADDR> | all }
```

#### Параметры

<ADDR> – IP-адрес NetBIOS-сервера задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно задать до 4 IP-адресов. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IP-адреса;

- all – команда удаляет все IP-адреса NetBIOS-серверов.

#### Необходимый уровень привилегий

1

#### Командный режим

ROOT

#### Пример

```
esr(config-dhcp-server)# netbios-name-server 192.168.45.15
```

### 29.5.15 show ip dhcp binding

Данная команда позволяет посмотреть выданные DHCP-сервером IP-адреса.

#### Синтаксис

```
show ip dhcp binding [ <ADDR> ]
```

#### Параметры

<ADDR> – IP-адрес, опциональный параметр, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Если в команде задан параметр <ADDR>, то будет отображена информация связанная только с указанным адресом.

#### Необходимый уровень привилегий

1

#### Командный режим

ROOT

#### Пример

```
esr# show ip dhcp binding
Allocated          MAC address          Binding      Lease expires at
```

| address     |                   | type    |                              |
|-------------|-------------------|---------|------------------------------|
| -----       | -----             | -----   | -----                        |
| 192.168.1.3 | 50:46:5d:a5:3f:91 | dynamic | Thursday 2014/01/01 12:42:12 |

### 29.5.16 *show ip dhcp server dscp*

Данная команда позволяет посмотреть значение DSCP для сообщений DHCP-сервера.

#### Синтаксис

```
show ip dhcp server dscp
```

#### Параметры

Команда не содержит аргументов.

#### Необходимый уровень привилегий

1

#### Командный режим

ROOT

#### Пример

```
esr # show ip dhcp server dscp
DSCP:      32
```

### 29.5.17 *show ip dhcp server pool*

Данная команда позволяет посмотреть настроенные пулы IP-адресов. При указании имени выводится информация только для заданного пула.

#### Синтаксис

```
show ip dhcp server pool [ <POOL_NAME> ]
```

#### Параметры

<POOL\_NAME> – имя пула, опциональный параметр.

#### Необходимый уровень привилегий

1

#### Командный режим

ROOT

#### Пример

```
esr# show ip dhcp server pool lan-pool
name:                lan-pool
network:              192.168.1.0/24
address-ranges:       192.168.1.2-192.168.1.254
default-router:        192.168.1.1
max lease time:        1:0:0 (day:hour:min)
default lease time:    0:12:0 (day:hour:min)
```

---

## 29.5.18 *show ip dhcp server vendor-specific*

---

Данная команда позволяет посмотреть настроенные DHCP опции 43 и 60.

### Синтаксис

```
show ip dhcp server vendor-specific
```

### Параметры

Команда не содержит аргументов

### Необходимый уровень привилегий

1

### Командный режим

ROOT

### Пример

```
esr# show ip dhcp server vendor-specific
Vendor ID      Vendor options
-----
ELTEX          0x0b0931302e312e39302e32
```

---

## 29.6 *Настройка и мониторинг IPv6 DHCP-сервера*

---

### 29.6.1 *ipv6 dhcp-server*

---

Данная команда включает IPv6 DHCP-сервер.

Использование отрицательной формы команды (no) выключает IPv6 DHCP-сервер.

### Синтаксис

```
[no] ipv6 dhcp-server
```

### Параметры

Команда не содержит аргументов.

### Значение по умолчанию

Выключен

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ipv6 dhcp-server
```

## 29.6.2 *ipv6 dhcp-server pool*

Команда используется для создания пула IPv6-адресов DHCP-сервера и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный пул IPv6-адресов.

### Синтаксис

```
[no] ipv6 dhcp-server pool <NAME>
```

### Параметры

<NAME> – имя пула IPv6-адресов DHCP-сервера, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все пулы IPv6-адресов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG

### Пример

```
esr(config)# ipv6 dhcp-server pool lan
```

## 29.6.3 *network*

Данная команда задает IPv6-адрес и маску для подсети, из которой будет выделен пул IP-адресов.

Использование отрицательной формы команды (no) удаляет настройки подсети в пуле.

### Синтаксис

```
network <IPV6-ADDR/LEN>  
no network
```

### Параметры

<IPV6-ADDR/LEN> – IP-адрес и префикс подсети, задаётся в виде X:X:X:X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IPV6-DHCP-SERVER

### Пример

```
esr(config-ipv6-dhcp-server)# network fc00::/110
```

## 29.6.4 *address-range*

Данная команда позволяет добавить диапазон IPv6-адресов к пулу адресов конфигурируемого DHCP-сервера.

Использование отрицательной формы команды (no) удаляет указанный диапазон IPv6-адресов.

### Синтаксис

[no] address-range <FROM-ADDR>-<TO-ADDR>

no address-range all

### Параметры

<FROM-ADDR> – начальный IPv6-адрес диапазона, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Можно указать до 32 диапазонов IPv6-адресов, список задаётся через запятую.

all – удалить все сконфигурированные диапазоны IPv6-адресов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IPV6-DHCP-SERVER

### Пример

```
esr(config-ipv6-dhcp-server)# address-range fc00::1-fc00:12,fc00::15-fc00::25
```

## 29.6.5 address

Данная команда позволяет добавить IPv6-адрес для определенного физического адреса к пулу адресов конфигурируемого DHCP-сервера.

Использование отрицательной формы команды (no) удаляет указанный IPv6-адрес.

### Синтаксис

address <IPV6-ADDR> mac-address <MAC>

no address <IPV6-ADDR>

### Параметры

<IPV6-ADDR> – IPv6-адрес клиента, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Если использовать команду для удаления, то при указании значения «all» будут удалены все IPv6-адреса;

<MAC> – MAC-адрес клиента, которому будет выдан IPv6-адрес, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IPV6-DHCP-SERVER

### Пример

```
esr(config-ipv6-dhcp-server)# address fc00::2 mac-address A8:F9:4B:AA:00:40
```

## 29.6.6 *domain-name*

Данная команда позволяет задать имя сетевого домена. Имя домена передаётся клиентам в составе DHCP-опций.

Использование отрицательной формы команды (no) удаляет установленное имя домена.

### Синтаксис

domain-name <NAME>

no domain-name

### Параметры

<NAME> – имя домена клиента, задаётся строкой до 255 символов.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IPV6-DHCP-SERVER

### Пример

```
esr(config-ipv6-dhcp-server)# domain-name eltex.loc
```

## 29.6.7 *dns-server*

Данная команда позволяет задать список IPv6-адресов DNS-серверов. Список передаётся клиентам в составе DHCP-опций.

Использование отрицательной формы команды удаляет указанный DNS-сервер из списка.

### Синтаксис

dns-server <IPV6-ADDR>

no dns-server { <IPV6-ADDR> | all }

### Параметры

<IPV6-ADDR> – IPv6-адрес DNS-сервера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Можно указать до 8 IPv6-адресов, список задаётся через запятую. Если использовать команду для удаления, то при указании значения «all» будут удалены все DNS-серверы.

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IPV6-DHCP-SERVER

### Пример

```
esr(config-ipv6-dhcp-server)# dns-server 8.8.8.8,8.8.4.4
```

---

## 29.6.8 *max-lease-time*

---

Данная команда позволяет задать максимальное время аренды IPv6-адресов. Если IPv6 DHCP-клиент запрашивает время аренды, превосходящее максимальное значение, то будет установлено время, заданное этой командой. Использование отрицательной формы команды устанавливает значение по умолчанию.

### Синтаксис

max-lease-time <TIME>

no max-lease-time

### Параметры

<TIME> – максимальное время аренды IP-адреса, задаётся в формате DD:HH:MM, где:

DD – количество дней, принимает значения [0..364];

HH – количество часов, принимает значения [0..23];

MM – количество минут, принимает значения [0..59].

### Значение по умолчанию

1 день

### Необходимый уровень привилегий

10

### Командный режим

CONFIG-IPV6-DHCP-SERVER

### Пример

```
esr(config-ipv6-dhcp-server)# max-lease-time 00:16:00
```

---

## 29.6.9 *default-lease-time*

---

Данная команда позволяет задать время аренды, на которое клиенту будет выдан IPv6-адрес, если клиент не запрашивал определенное время аренды.

Использование отрицательной формы команды устанавливает значение по умолчанию.

### Синтаксис

default-lease-time <TIME>

no default-lease-time

### Параметры

<TIME> – время аренды IP-адреса, в формате DD:HH:MM, где:

DD – количество дней, принимает значения [0..364];

HH – количество часов, принимает значения [0..23];

MM – количество минут, принимает значения [0..59].

### Значение по умолчанию

12 часов

### Необходимый уровень привилегий



## Командный режим

CONFIG-IPV6-DHCP-SERVER

## Пример

```
esr(config-ipv6-dhcp-server)# default-lease-time 00:04:00
```

### 29.6.10 *ipv6 dhcp-server vendor-class-id*

Данная команда необходима для создания идентификатора класса поставщика (DHCP Опция 60) и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный идентификатор класса поставщика.

## Синтаксис

```
[no] ipv6 dhcp-server vendor-class-id <NAME>
```

## Параметры

<NAME> – идентификатор класса поставщика, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все идентификаторы класса поставщика.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG

## Пример

```
esr(config)# ipv6 dhcp-server vendor-class-id ELTEX
```

### 29.6.11 *vendor-specific-options*

Данная команда позволяет задать специфическую информацию поставщика (DHCP Опция 43).

Использование отрицательной формы команды (no) удаляет специфическую информацию поставщика.

## Синтаксис

```
vendor-specific-options <HEX>
```

```
no vendor-specific-options
```

## Параметры

<HEX> – специфическая информация поставщика, задаётся в шестнадцатеричном формате до 128 символов.

## Необходимый уровень привилегий

10

## Командный режим

CONFIG-IPV6-DHCP-VENDOR-ID

## Пример

```
esr(config-dhcp-vendor-id)# vendor-specific-options 0b0931302e312e39302e320
```

### 29.6.12 *show ipv6 dhcp binding*

Данная команда позволяет посмотреть выданные DHCP-сервером IPv6-адреса.

#### Синтаксис

```
show ipv6 dhcp binding [ <IPV6-ADDR> ]
```

#### Параметры

<IPV6-ADDR> – IPv6-адрес, опциональный параметр, задаётся в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Если в команде задан параметр <IPV6-ADDR>, то будет отображена информация, связанная только с указанным адресом.

#### Необходимый уровень привилегий

1

#### Командный режим

ROOT

#### Пример

```
esr# show ipv6 dhcp binding
IPv6 address                               Lease expires at
-----
2001:db8:0:1::18                          Tuesday 2015/06/09 23:39:45
```

### 29.6.13 *show ipv6 dhcp server pool*

Данная команда позволяет посмотреть настроенные пулы IPv6-адресов. При указании имени выводится информация только для заданного пула.

#### Синтаксис

```
show ipv6 dhcp server pool [ <POOL_NAME> ]
```

#### Параметры

<POOL\_NAME> – имя пула, опциональный параметр, задается строкой до 31 символа.

#### Необходимый уровень привилегий

1

#### Командный режим

ROOT

#### Пример

```
esr# show ipv6 dhcp server pool lan-pool-ipv6
name:                lan-pool-ipv6
network:             fc00::/110
domain name:         example.com
address-ranges:      fc00::1-fc00::2
                    fc00::3
addresses:           --
dns-server:          fc00:2::3
```

```
max lease time:      001:00:00 (day:hour:min)
default lease time:  000:12:00 (day:hour:min)
```

### **29.6.14**      ***show ipv6 dhcp server vendor-specific***

---

Данная команда позволяет посмотреть настроенные DHCP-опции 43 и 60.

#### **Синтаксис**

```
show ipv6 dhcp server vendor-specific
```

#### **Параметры**

Команда не содержит аргументов

#### **Необходимый уровень привилегий**

1

#### **Командный режим**

ROOT

#### **Пример**

```
esr# show ipv6 dhcp server vendor-specific
Vendor ID      Vendor options
-----
ELTEX          0x0b0931302e312e39302e32
```

Для получения технической консультации по вопросам эксплуатации оборудования ООО «Предприятие «ЭЛТЕКС» Вы можете обратиться в Сервисный центр компании:

Российская Федерация ,630020, г. Новосибирск, ул. Окружная, дом 29 в.

Телефон:

+7(383) 274-47-87

+7(383) 272-83-31

E-mail: [techsupp@eltex.nsk.ru](mailto:techsupp@eltex.nsk.ru)

На официальном сайте компании Вы можете найти техническую документацию и программное обеспечение для продукции ООО «Предприятие «ЭЛТЕКС», обратиться к базе знаний, проконсультироваться у инженеров Сервисного центра на техническом форуме:

Официальный сайт компании: **<http://eltex.nsk.ru>**

Технический форум: **<http://eltex.nsk.ru/forum>**

База знаний: **<http://eltex.nsk.ru/support/knowledge>**

Центр загрузок: **<http://eltex.nsk.ru/support/downloads>**