



Маршрутизаторы серии RTT R100, R200, R800

Руководство по эксплуатации, версия ПО 1.37

Содержание

1.	ВВЕДЕНИЕ	11
1.1.	Аннотация	11
1.2.	Целевая аудитория	11
1.3.	Условные обозначения	11
2.	ОПИСАНИЕ ИЗДЕЛИЯ	13
2.1.	Назначение	13
2.2.	Функции	13
2.2.1.	Функции интерфейсов	13
2.2.2.	Функции при работе с MAC-адресами	14
2.2.3.	Функции второго уровня сетевой модели OSI	14
2.2.4.	Функции третьего уровня сетевой модели OSI	15
2.2.5.	Функции туннелирования трафика	16
2.2.6.	Функции управления и конфигурирования	16
2.2.7.	Функции сетевой защиты	17
2.3.	Основные технические характеристики	17
	Общие параметры	17
2.4.	Конструктивное исполнение	20
2.4.1.	Конструктивное исполнение R800	20
2.4.2.	Конструктивное исполнение R100, R200	22
2.4.3.	Световая индикация	24
2.5.	Комплект поставки	28
3.	УСТАНОВКА И ПОДКЛЮЧЕНИЕ	30
3.1.	Крепление кронштейнов	30
3.2.	Установка устройства в стойку	30
3.3.	Установка модулей питания R800	32
3.4.	Подключение питающей сети	32
3.5.	Установка и удаление SFP-трансиверов	33
3.5.1.	Установка трансивера	33
3.5.2.	Удаление трансивера	33
4.	ИНТЕРФЕЙСЫ УПРАВЛЕНИЯ	35
4.1.	Интерфейс командной строки (CLI)	35
4.2.	Типы и порядок именования интерфейсов маршрутизатора	36
4.3.	Типы и порядок именования туннелей маршрутизатора	37
5.	НАЧАЛЬНАЯ НАСТРОЙКА МАРШРУТИЗАТОРА	39
5.1.	Заводская конфигурация маршрутизатора RTT	39
5.1.1.	Описание заводской конфигурации	39
5.2.	Подключение и конфигурирование маршрутизатора	40
5.2.1.	Подключение к маршрутизатору	40
5.2.2.	Применение изменения конфигурации	41
5.2.3.	Базовая настройка маршрутизатора	42
6.	ОБНОВЛЕНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	46
6.1.	Обновление программного обеспечения средствами системы	46
6.2.	Обновление программного обеспечения из начального загрузчика	48
6.3.	Обновление вторичного загрузчика (U-Boot)	49
7.	РЕКОМЕНДАЦИИ ПО БЕЗОПАСНОЙ НАСТРОЙКЕ	51
7.1.	Общие рекомендации	51
7.2.	Настройка системы логирования событий	51
7.2.1.	Рекомендации	51

7.2.2.	Предупреждения.....	52
7.2.3.	Пример настройки	52
7.3.	Настройка политики использования паролей.....	52
7.3.1.	Рекомендации.....	53
7.3.2.	Пример настройки	53
7.4.	Настройка политики AAA.....	54
7.4.1.	Рекомендации.....	54
7.4.2.	Предупреждения.....	54
7.4.3.	Пример настройки	54
7.5.	Настройка удалённого управления	56
7.5.1.	Рекомендации.....	56
7.5.2.	Пример настройки	56
7.6.	Настройка механизмов защиты от сетевых атак.....	57
7.6.1.	Рекомендации.....	57
7.6.2.	Пример настройки	57
8.	УПРАВЛЕНИЕ ИНТЕРФЕЙСАМИ	59
8.1.	Настройка физического интерфейса	59
8.1.1.	Алгоритм настройки.....	59
8.1.2.	Алгоритм настройки режима L3	60
8.1.3.	Пример настройки в режиме L3	60
8.2.	Настройка терминации на саб-интерфейсе.....	61
8.2.1.	Алгоритм настройки.....	61
8.2.2.	Пример настройки саб-интерфейса	63
8.3.	Настройка терминации на Q-in-Q интерфейсе	64
8.3.1.	Алгоритм настройки.....	64
8.3.2.	Пример настройки Q-in-Q интерфейса.....	67
8.4.	Настройка USB-модемов	67
8.4.1.	Алгоритм настройки USB-модемов	68
8.4.2.	Пример настройки	70
8.5.	Настройка PPP через E1	71
8.5.1.	Алгоритм настройки.....	71
8.5.2.	Пример конфигурации.....	75
8.6.	Настройка MLPPP.....	76
8.6.1.	Алгоритм настройки.....	76
8.6.2.	Пример настройки	79
8.6.3.	Фрагментация трафика	80
9.	УПРАВЛЕНИЕ ТУННЕЛИРОВАНИЕМ	82
9.1.	Настройка GRE-туннелей	82
9.1.1.	Алгоритм настройки.....	82
9.1.2.	Пример настройки IP-GRE-туннеля	86
9.2.	Настройка DMVPN	88
9.2.1.	Алгоритм настройки.....	88
9.2.2.	Пример настройки 1	91
9.2.3.	Пример настройки 2	97
9.3.	Настройка L2TPv3-туннелей	102
9.3.1.	Алгоритм настройки.....	102
9.3.2.	Пример настройки L2TPv3-туннеля	104
9.4.	Настройка IPsec VPN.....	106
9.4.1.	Алгоритм настройки Route-based IPsec VPN	106

9.4.2.	Пример настройки Route-based IPsec VPN	115
9.4.3.	Алгоритм настройки Policy-based IPsec VPN	119
9.4.4.	Пример настройки Policy-based IPsec VPN с аутентификацией по общему известному ключу	125
9.4.5.	Пример настройки Policy-based IPsec VPN с аутентификацией сертификатам X.509, выписываемых PKI-клиентом.....	129
9.4.6.	Алгоритм настройки Remote Access IPsec VPN	135
9.4.7.	Пример настройки Remote Access IPsec VPN	143
9.4.8.	Пример настройки DPD (Dead Peer Detection).....	148
9.5.	Настройка LT-туннелей	150
9.5.1.	Алгоритм настройки.....	150
9.5.2.	Пример настройки	151
10.	УПРАВЛЕНИЕ ФУНКЦИЯМИ ВТОРОГО УРОВНЯ (L2)	153
10.1.	Настройка физического интерфейса	153
10.1.1.	Алгоритм настройки.....	153
10.1.2.	Пример настройки режима L2	154
10.2.	Настройка VLAN.....	155
10.2.1.	Алгоритм настройки.....	156
10.2.2.	Манипуляции с VLAN на интерфейсе	158
10.2.3.	Пример настройки 1	158
10.2.4.	Пример настройки 2	159
10.2.5.	Разрешение обработки VLAN в тегированном и нетегированном режимах.....	160
10.2.6.	Пример настройки 1	160
10.2.7.	Пример настройки 2	161
10.2.8.	Пример настройки Private Vlan	163
10.3.	Настройка LLDP.....	164
10.3.1.	Алгоритм настройки.....	164
10.3.2.	Пример настройки	165
10.4.	Настройка LLDP MED	166
10.4.1.	Алгоритм настройки.....	166
10.4.2.	Пример настройки Voice VLAN	168
10.5.	Настройка протоколов семейства STP	169
10.5.1.	Настройка протоколов STP и RSTP	169
10.5.2.	Настройка протокола STP и RSTP в рамках bridge	172
10.5.3.	Настройка протокола MSTP	176
10.5.4.	Настройка BPDU Guard.....	179
10.6.	Настройка Bridge	181
10.6.1.	Алгоритм настройки.....	181
10.6.2.	Пример настройки bridge для VLAN и L2TPv3-туннеля	185
10.6.3.	Пример настройки bridge для VLAN	186
10.6.4.	Пример настройки добавления/удаления второго VLAN-тега	188
10.7.	Настройка Dual-Homing	188
10.7.1.	Алгоритм настройки.....	189
10.7.2.	Пример настройки	189
10.8.	Настройка зеркалирования (SPAN/RSPAN)	190
10.8.1.	Алгоритм настройки.....	191
10.8.2.	Пример настройки	191
10.9.	Настройка LACP	192
10.9.1.	Алгоритм настройки.....	193
10.9.2.	Пример настройки	195

11.	УПРАВЛЕНИЕ QOS	198
11.1.	Базовый QoS	198
11.1.1.	Алгоритм настройки.....	198
11.1.2.	Пример настройки	201
11.1.3.	Пример расчета пропускной способности для взвешенных очередей	203
11.2.	Расширенный QoS.....	204
11.2.1.	Алгоритм настройки.....	204
11.2.2.	Пример настройки	216
11.2.3.	Механизм работы полисера	218
11.3.	MPLS QoS.....	219
12.	УПРАВЛЕНИЕ МАРШРУТИЗАЦИЕЙ	220
12.1.	Политика фильтрации маршрутной информации	220
12.1.1.	Протокол RIP	223
12.1.2.	Протокол OSPF.....	223
12.1.3.	Протокол IS-IS	224
12.1.4.	Протокол iBGP.....	225
12.1.5.	Протокол eBGP	225
12.2.	Конфигурирование статических маршрутов	226
12.2.1.	Алгоритм настройки.....	226
12.2.2.	Пример настройки	228
12.3.	Конфигурирование статических multipath-маршрутов	229
12.3.1.	Алгоритм настройки.....	230
12.3.2.	Пример настройки	231
12.4.	Настройка RIP	233
12.4.1.	Алгоритм настройки.....	234
12.4.2.	Пример настройки	238
12.5.	Настройка RIPng	239
12.5.1.	Алгоритм настройки.....	240
12.5.2.	Пример настройки	243
12.6.	Настройка OSPF	244
12.6.1.	Алгоритм настройки.....	244
12.6.2.	Пример настройки	255
12.6.3.	Пример настройки OSPF stub area	257
12.6.4.	Пример настройки Virtual link	257
12.7.	Настройка BGP.....	259
12.7.1.	Алгоритм настройки.....	259
12.7.2.	Пример настройки	272
12.7.3.	Политика выбора лучшего маршрута в протоколе BGP	274
12.7.4.	Условное анонсирование маршрутной информации (Conditional Advertisement) ..	276
12.7.5.	Быстрая деактивация пиринговых сессий	286
12.7.6.	Настройка политик маршрутизации Route-map.....	292
12.7.7.	Конфедерация	312
12.8.	Настройка Policy-Based Routing.....	320
12.8.1.	Алгоритм настройки.....	321
12.8.2.	Пример настройки	322
12.9.	Настройка BFD	324
12.9.1.	Настройка таймеров	325
12.9.2.	Алгоритм настройки.....	327
12.9.3.	Пример настройки	330

12.10.	Настройка VRF	332
12.10.1.	Алгоритм настройки	332
12.10.2.	Пример настройки	333
12.11.	Настройка MultiWAN	334
12.11.1.	Алгоритм настройки	335
12.11.2.	Пример настройки	337
12.12.	Настройка IS-IS	339
12.12.1.	Алгоритм настройки	340
12.12.2.	Пример настройки	348
13.	УПРАВЛЕНИЕ ТЕХНОЛОГИЕЙ MPLS	351
13.1.	Настройка протокола LDP	351
13.1.1.	Алгоритм настройки	351
13.1.2.	Пример настройки	353
13.2.	Конфигурирование параметров сессии в протоколе LDP	355
13.2.1.	Алгоритм настройки параметров Hello holdtime и Hello interval в глобальной конфигурации LDP	357
13.2.2.	Алгоритм настройки параметров Hello holdtime и Hello interval для address family	357
13.2.3.	Алгоритм настройки параметра Keepalive holdtime в глобальной конфигурации LDP	358
13.2.4.	Алгоритм настройки параметра Keepalive holdtime для определенного соседа	358
13.2.5.	Пример настройки	358
13.3.	Конфигурирование параметров сессии в протоколе targeted-LDP	359
13.3.1.	Алгоритм настройки параметров Hello holdtime, Hello interval и Keepalive holdtime для процесса LDP	361
13.3.2.	Алгоритм настройки параметров Hello holdtime, Hello interval и Keepalive holdtime для определенного соседа	362
13.3.3.	Пример настройки	362
13.4.	Настройка фильтрации LDP-меток	363
13.4.1.	Метод на основе Advertise-labels	364
13.4.2.	Метод на основе Prefix-list	365
13.5.	Настройка сервиса L2VPN Martini mode	367
13.5.1.	Алгоритм настройки L2VPN VPWS	368
13.5.2.	Пример настройки L2VPN VPWS	369
13.5.3.	Алгоритм настройки L2VPN VPLS	372
13.5.4.	Пример настройки L2VPN VPLS	373
13.6.	Настройка сервиса L2VPN Kompella mode	377
13.6.1.	Алгоритм настройки L2VPN VPLS	377
13.6.2.	Пример настройки L2VPN VPLS	379
13.7.	Настройка сервиса L3VPN	391
13.7.1.	Алгоритм настройки	392
13.7.2.	Пример настройки	394
13.8.	Балансировка трафика MPLS	405
13.8.1.	Пример настройки	406
13.9.	Работа с бридж-доменом в рамках MPLS	406
13.10.	Назначение MTU при работе с MPLS	408
13.11.	Inter-AS Option A	414
13.11.1.	L2VPN	414
13.11.2.	L3VPN	423
13.12.	Inter-AS Option B	435
13.12.1.	L3VPN	435

13.13.	Inter-AS Option C.....	445
13.13.1.	L3VPN	446
13.14.	MPLS over GRE	448
13.14.1.	L2VPN	449
13.14.2.	L3VPN	453
14.	УПРАВЛЕНИЕ БЕЗОПАСНОСТЬЮ	461
14.1.	Настройка AAA.....	461
14.1.1.	Алгоритм настройки локальной аутентификации.....	461
14.1.2.	Алгоритм настройки AAA по протоколу RADIUS.....	465
14.1.3.	Алгоритм настройки AAA по протоколу TACACS.....	468
14.1.4.	Алгоритм настройки AAA по протоколу LDAP.....	472
14.1.5.	Пример настройки аутентификации по Telnet через RADIUS-сервер.....	477
14.2.	Настройка привилегий команд.....	477
14.2.1.	Алгоритм настройки.....	478
14.2.2.	Пример настройки привилегий команд.....	478
14.3.	Настройка логирования и защиты от сетевых атак.....	478
14.3.1.	Алгоритм настройки.....	478
14.3.2.	Описание механизмов защиты от атак	481
14.3.3.	Пример настройки логирования и защиты от сетевых атак.....	483
14.4.	Конфигурирование Firewall.....	485
14.4.1.	Алгоритм настройки.....	485
14.4.2.	Пример настройки Firewall.....	495
14.4.3.	Пример настройки Firewall по доменным именам.....	497
14.4.4.	Пример настройки фильтрации приложений (DPI).....	499
14.5.	Настройка списков доступа (ACL)	501
14.5.1.	Алгоритм настройки.....	501
14.5.2.	Пример настройки списка доступа	504
14.6.	Проксирование HTTP/HTTPS-трафика	504
14.6.1.	Алгоритм настройки.....	504
14.6.2.	Пример настройки HTTP-прокси.....	507
14.7.	Настройка IPS/IDS.....	510
14.7.1.	Алгоритм базовой настройки.....	510
14.7.2.	Алгоритм настройки автообновления правил IPS/IDS из внешних источников	512
14.7.3.	Рекомендуемые открытые источники обновления правил	513
14.7.4.	Пример настройки IPS/IDS с автообновлением правил	515
14.7.5.	Алгоритм настройки базовых пользовательских правил	516
14.7.6.	Пример настройки базовых пользовательских правил	524
14.7.7.	Алгоритм настройки расширенных пользовательских правил.....	525
14.7.8.	Пример настройки расширенных пользовательских правил	526
15.	УПРАВЛЕНИЕ СЕРТИФИКАТАМИ И КЛЮЧАМИ.....	528
15.1.	Автоматическое распространение ключей и сертификатов X.509	528
15.1.1.	Общее описание инфраструктуры открытых ключей	528
15.1.2.	Планирование инфраструктуры открытых ключей	529
15.1.3.	Настройка PKI-сервера в роли корневого удостоверяющего центра	530
15.1.4.	Настройка PKI-клиента.....	534
15.1.5.	Процесс автоматического перевыпуска сертификата PKI-клиента.....	540
15.1.6.	Процесс автоматического перевыпуска сертификата PKI-сервера.....	541
15.2.	Ручная генерация и распространение ключей и сертификатов X.509	541
15.2.1.	Алгоритм генерации ключей и запросов на сертификацию	541

15.2.2.	Пример ручного выпуска сертификата через внешний удостоверяющий центр.....	543
16.	УПРАВЛЕНИЕ РЕЗЕРВИРОВАНИЕМ	548
16.1.	Настройка VRRP	548
16.1.1.	Алгоритм настройки.....	548
16.1.2.	Пример настройки 1	551
16.1.3.	Пример настройки 2	552
16.2.	Настройка tracking	554
16.2.1.	Алгоритм настройки.....	554
16.2.2.	Пример настройки	561
16.3.	Настройка Firewall/NAT failover	562
16.3.1.	Алгоритм настройки.....	563
16.3.2.	Пример настройки	564
16.4.	Настройка DHCP failover	568
16.4.1.	Алгоритм настройки.....	568
16.4.2.	Пример настройки	569
17.	УПРАВЛЕНИЕ КЛАСТЕРИЗАЦИЕЙ	574
17.1.	Настройка кластера	574
17.1.1.	Алгоритм настройки.....	574
17.1.2.	Пример настройки кластера	578
17.2.	Подключение сервисов	584
17.2.1.	Настройка System prompt	584
17.2.2.	Настройка Port-channel U/N	587
17.2.3.	Настройка MultiWAN.....	590
17.2.4.	Настройка IPsec VPN.....	595
17.2.5.	Настройка firewall/NAT failover	608
17.2.6.	Настройка DHCP failover.....	630
17.2.7.	Настройка SNMP	640
17.2.8.	Настройка Source NAT	643
17.2.9.	Настройка Destination NAT	647
17.2.10.	Настройка BGP	652
17.2.11.	Настройка DMVPN	664
18.	УПРАВЛЕНИЕ УДАЛЕННЫМ ДОСТУПОМ	687
18.1.	Настройка сервера удаленного доступа к корпоративной сети по RPTP-протоколу.....	687
18.1.1.	Алгоритм настройки.....	687
18.1.2.	Пример настройки	689
18.2.	Настройка сервера удаленного доступа к корпоративной сети по L2TP over IPsec протоколу.....	692
18.2.1.	Алгоритм настройки.....	692
18.2.2.	Пример настройки	695
18.3.	Настройка сервера удаленного доступа к корпоративной сети по OpenVPN-протоколу	697
18.3.1.	Алгоритм настройки.....	697
18.3.2.	Пример настройки	701
18.4.	Настройка сервера удаленного доступа к корпоративной сети по WireGuard-протоколу.....	704
18.4.1.	Алгоритм настройки.....	704
18.4.2.	Пример настройки	706
18.4.3.	Пример настройки правил Firewall для совместной работы с WireGuard-сервером	708
18.5.	Настройка клиента удаленного доступа по протоколу PPPoE	710
18.5.1.	Алгоритм настройки.....	710

18.5.2.	Пример настройки	712
18.6.	Настройка клиента удаленного доступа по протоколу PPTP	714
18.6.1.	Алгоритм настройки.....	714
18.6.2.	Пример настройки	716
18.7.	Настройка клиента удаленного доступа по протоколу L2TP.....	717
18.7.1.	Алгоритм настройки.....	717
18.7.2.	Пример настройки	719
18.8.	Настройка клиента удаленного доступа по протоколу WireGuard.....	721
18.8.1.	Алгоритм настройки.....	721
18.8.2.	Пример настройки	723
19.	УПРАВЛЕНИЕ СЕРВИСАМИ	727
19.1.	Настройка DHCP-сервера	727
19.1.1.	Алгоритм настройки.....	727
19.1.2.	Пример настройки	730
19.2.	Конфигурирование Destination NAT	732
19.2.1.	Алгоритм настройки.....	732
19.2.2.	Пример настройки Destination NAT	735
19.3.	Конфигурирование Source NAT.....	737
19.3.1.	Алгоритм настройки.....	738
19.3.2.	Пример настройки 1	741
19.3.3.	Пример настройки 2	743
19.4.	Конфигурирование Static NAT.....	745
19.4.1.	Алгоритм настройки.....	745
19.4.2.	Пример настройки Static NAT	745
19.5.	Настройка NTP.....	747
19.5.1.	Алгоритм настройки.....	747
19.5.2.	Пример настройки	749
20.	МОНИТОРИНГ	751
20.1.	Настройка Netflow	751
20.1.1.	Алгоритм настройки.....	751
20.1.2.	Пример настройки	752
20.2.	Настройка sFlow	753
20.2.1.	Алгоритм настройки.....	753
20.2.2.	Пример настройки	754
20.3.	Настройка SNMP.....	755
20.3.1.	Алгоритм настройки.....	755
20.3.2.	Пример настройки	760
20.4.	Настройка Zabbix-agent/proxy.....	761
20.4.1.	Алгоритм настройки.....	762
20.4.2.	Пример настройки zabbix-agent.....	763
20.4.3.	Пример настройки zabbix-server	764
20.5.	Настройка Syslog	767
20.5.1.	Алгоритм настройки.....	767
20.5.2.	Пример настройки	772
20.6.	Проверка целостности.....	773
20.6.1.	Процесс настройки.....	774
20.6.2.	Пример конфигурации.....	774
20.7.	Настройка архивации конфигурации маршрутизатора.....	774
20.7.1.	Процесс настройки.....	774

20.7.2.	Пример конфигурации.....	775
20.8.	Настройка SLA.....	776
20.8.1.	Алгоритм настройки SLA-теста	776
20.8.2.	Настройка SLA-responder	783
20.8.3.	Пример настройки ICMP-режима тестирования	784
20.8.4.	Пример настройки UDP-режима тестирования.....	785
20.8.5.	Алгоритм настройки параметров аутентификации	787
20.8.6.	Пример конфигурации UDP-теста с аутентификацией по ключ-строке	791
20.8.7.	Пример конфигурации UDP-теста с аутентификацией по связке ключей.....	793
20.8.8.	Настройка пороговых значений.....	794
20.8.9.	Измерение характеристик канала связи	796
21.	ЧАСТО ЗАДАВАЕМЫЕ ВОПРОСЫ	798
22.	ПРИЛОЖЕНИЕ А. PACKET FLOW	801
22.1.	Порядок обработки входящего/исходящего трафика сетевыми службами маршрутизаторов RTT	801
22.2.	Порядок обработки транзитного трафика сетевыми службами маршрутизаторов RTT ..	802

1. ВВЕДЕНИЕ

1.1. Аннотация

В настоящее время осуществляются масштабные проекты по построению сетей связи. Одной из основных задач при реализации крупных мультисервисных сетей является создание надежных и высокопроизводительных транспортных сетей, которые являются опорными в многослойной архитектуре сетей следующего поколения.

Маршрутизаторы серии RTT могут использоваться на сетях крупных предприятий и предприятий малого и среднего бизнеса (SMB), в операторских сетях. Устройства обеспечивают высокую производительность, высокую пропускную способность и поддерживают функции защиты передаваемых данных.

В данном руководстве по эксплуатации изложены назначение, технические характеристики, функции, конструктивное исполнение, порядок установки, рекомендации по начальной настройке и обновлению программного обеспечения маршрутизатора серии RTT (далее устройство).

1.2. Целевая аудитория

Данное руководство пользователя предназначено для технического персонала, выполняющего установку, настройку и мониторинг устройств посредством интерфейса командной строки (CLI), а также процедуры по обслуживанию системы и обновлению ПО. Квалификация технического персонала предполагает знание основ работы стеков протоколов TCP/IP, принципов построения Ethernet-сетей.

1.3. Условные обозначения

Обозначение	Описание
<i>Курсив Calibri</i>	Курсивом Calibri указываются переменные или параметры, которые необходимо заменить соответствующим словом или строкой.
Полужирный курсив	Полужирным шрифтом выделены примечания и предупреждения.
<Полужирный курсив>	В угловых скобках указываются названия клавиш на клавиатуре.
Courier New	Полужирным Шрифтом Courier New записаны примеры ввода команд.
Courier New	Шрифтом Courier New в рамке с тенью указаны результаты выполнения команд.
[]	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 устройству или человеку, привести к некорректной работе устройства или потере данных.

2. ОПИСАНИЕ ИЗДЕЛИЯ

2.1. Назначение

Устройства серии RTT являются высокопроизводительными многоцелевыми сетевыми маршрутизаторами. Устройство объединяет в себе традиционные сетевые функции и комплексный многоуровневый подход к безопасности маршрутизации, что позволяет обеспечить надежную защиту для корпоративной среды.

Устройство поддерживает функции межсетевого экрана для защиты своей сетевой инфраструктуры и сочетает в себе новейшие средства обеспечения безопасности данных, шифрования, аутентификации и защиты от вторжений.

Устройство содержит в себе средства для программной и аппаратной обработки данных. За счет оптимального распределения функций обработки данных между частями достигается максимальная производительность.

2.2. Функции

2.2.1. Функции интерфейсов

В таблице 1 приведен список функций интерфейсов устройства.

Таблица 1 – Функции интерфейсов устройства

Определение полярности подключения кабеля (Auto MDI/MDIX)	Автоматическое определение типа кабеля - перекрестный кабель или кабель прямого подключения. <ul style="list-style-type: none"> – MDI (Medium Dependent Interface – прямой) – стандарт кабелей для подключения оконечных устройств; – MDIX (Medium Dependent Interface with Crossover – перекрестный) – стандарт кабелей для подключения концентраторов и коммутаторов.
Поддержка обратного давления (Back pressure)	Метод обратного давления используется на полудуплексных соединениях для регулирования потока данных от встречного устройства путем создания коллизий. Метод позволяет избежать переполнения буферной памяти устройства и потери данных.
Управление потоком (IEEE 802.3X)	Управление потоком позволяет соединять низкоскоростное устройство с высокоскоростным. Для предотвращения переполнения буфера низкоскоростное устройство имеет возможность отправлять пакет PAUSE, тем самым информируя высокоскоростное устройство о необходимости сделать паузу при передаче пакетов.
Агрегирование каналов (LAG, Link aggregation)	Агрегирование (объединение) каналов позволяет увеличить пропускную способность канала связи и повысить его надежность. Маршрутизатор поддерживает статическое и динамическое агрегирование каналов. При динамическом агрегировании используется протокол LACP для управления группой каналов.

2.2.2. Функции при работе с MAC-адресами

В таблице 2 приведены функции устройства при работе с MAC-адресами.

Таблица 2 – Функции работы с MAC-адресами

Таблица MAC-адресов	Таблица MAC-адресов устанавливает соответствие между MAC-адресами и интерфейсами устройства и используется для маршрутизации пакетов данных. Маршрутизаторы имеют таблицу емкостью до 16K MAC-адресов и резервируют определенные MAC-адреса для использования системой.
Режим обучения	MAC-таблица может содержать либо статические адреса, либо адреса, изученные при прохождении пакетов данных через устройство. Изучение происходит за счет регистрации MAC-адресов отправителей пакетов с привязкой их к портам и VLAN. Впоследствии эти данные используются для маршрутизации встречных пакетов. Время хранения зарегистрированных MAC-адресов ограничено, его продолжительность может настраиваться администратором. Если MAC-адрес получателя, указанный в принятом устройством пакете, отсутствует в таблице, то такой пакет отправляется далее, как широковещательный в пределах L2 сегмента сети.

2.2.3. Функции второго уровня сетевой модели OSI

В таблице 3 приведены функции и особенности второго уровня (уровень 2 OSI).

Таблица 3 – Описание функций второго уровня (уровень 2 OSI)

Поддержка VLAN	VLAN (Virtual Local Area Network) – это средство разделения сети на изолированные сегменты на уровне L2. Использование VLAN позволяет повысить устойчивость работы крупных сетей за счет деления их на более мелкие сети, изолировать разнородный трафик данных между собой и решить многие другие задачи. Маршрутизаторы поддерживают различные способы организации VLAN: <ul style="list-style-type: none"> – VLAN на базе меток пакетов данных, в соответствии с IEEE 802.1Q; – VLAN на базе портов устройства (port-based); – VLAN на базе использования правил классификации данных (policy-based).
Протокол связующего дерева (Spanning Tree Protocol)¹	Задачей протокола Spanning Tree является исключение избыточных сетевых соединений и приведение топологии сети к древовидной. Основные применения протокола связаны с предотвращением заикливания сетевого трафика и с организацией резервных каналов связи.

¹ В текущей версии ПО данный функционал поддерживается только на маршрутизаторе R-800

2.2.4. Функции третьего уровня сетевой модели OSI

В таблице 4 приведены функции третьего уровня (уровень 3 OSI).

Таблица 4 – Описание функций третьего уровня (Layer 3)

Статические IP-маршруты	Администратор маршрутизатора имеет возможность добавлять и удалять статические записи в таблицу маршрутизации.
Динамическая маршрутизация	Протоколы динамической маршрутизации позволяют устройству обмениваться маршрутной информацией с соседними маршрутизаторами и автоматически составлять таблицу маршрутов. Маршрутизатор поддерживает следующие протоколы: RIP, OSPFv2, OSPFv3, BGP.
Таблица ARP	ARP (Address Resolution Protocol) – протокол для выяснения соответствия адресов сетевого и канального уровней. Таблица ARP содержит информацию об изученном соответствии. Соответствие устанавливается на основе анализа ответов от сетевых устройств, адреса устройств запрашиваются с помощью широковещательных пакетов.
Клиент DHCP	Протокол DHCP (Dynamic Host Configuration Protocol) даёт возможность автоматизировать управление сетевыми устройствами. Клиент DHCP позволяет маршрутизатору получать сетевой адрес и дополнительные параметры от внешнего DHCP-сервера. Как правило, этот способ используется для получения сетевых настроек оператора публичной сети (WAN).
Сервер DHCP	Сервер DHCP предназначен для автоматизации и централизации конфигурирования сетевых устройств. Размещение DHCP-сервера на маршрутизаторе позволяет получить законченное решение для поддержки локальной сети. DHCP-сервер, входящий в состав маршрутизатора, позволяет назначать IP-адреса сетевым устройствам и передавать дополнительные сетевые параметры – адреса серверов, адреса шлюзов сети и другие необходимые параметры.
DHCP Relay	Функция DHCP Relay предназначена для перенаправления широковещательных DHCP Discover-пакетов из одного широковещательного домена в одноадресные (unicast) DHCP Discover-пакеты в другом широковещательном домене.
Трансляция сетевых адресов (NAT, Network Address Translation)	Трансляция сетевых адресов – это механизм, который позволяет преобразовывать IP-адреса и номера портов транзитных пакетов. Функция NAT позволяет использовать меньшее количество IP-адресов, транслируя несколько IP-адресов внутренней сети в один внешний публичный IP-адрес. Использование NAT позволяет увеличить защищённость локальной сети за счёт скрытия её внутренней структуры. Маршрутизаторы поддерживают следующие варианты NAT: <ul style="list-style-type: none"> – Source NAT (SNAT) – выполняется замена адреса, а также номера порта источника при прохождении пакета в одну сторону и обратной замене адреса назначения в ответном пакете; – Destination NAT (DNAT) – когда обращения извне транслируются межсетевым экраном на компьютер пользователя в локальной сети,

	имеющий внутренний адрес и потому недоступный извне сети непосредственно (без NAT).
--	---

2.2.5. Функции туннелирования трафика

Таблица 5 – Функции туннелирования трафика

Протоколы туннелирования	<p>Туннелирование – это способ преобразования пакетов данных при передаче их по сети, при котором происходит замена, модификация или добавление нового сетевого заголовка пакета. Такой способ может быть использован для согласования транспортных протоколов при прохождении данных через транзитную сеть, для создания защищенных соединений, при которых туннелированные данные подвергаются шифрованию.</p> <p>Маршрутизаторы поддерживают следующие виды туннелей:</p> <ul style="list-style-type: none"> – GRE - инкапсуляция IP-пакета в другой IP-пакет с добавлением GRE (General Routing Encapsulation) заголовка; – IPv4-IPv4 – туннель, использующий инкапсуляцию исходных IP-пакетов в IP-пакеты с другими сетевыми параметрами; – L2TPv3 – туннель для передачи L2-трафика с помощью IP-пакетов; – IPsec – туннель с шифрованием передаваемых данных; – L2TP, PPTP – туннели, использующиеся для организации удаленного доступа клиент-сервер.
---------------------------------	--

2.2.6. Функции управления и конфигурирования

Таблица 6 – Основные функции управления и конфигурирования

Загрузка и выгрузка файла настройки	Параметры устройства сохраняются в файле настройки, который содержит данные конфигурации как всей системы в целом, так и определенного порта устройства. Для передачи файлов могут использоваться протоколы TFTP, FTP, SCP.
Интерфейс командной строки (CLI)	Управление посредством CLI осуществляется локально через последовательный порт RS-232 либо удаленно через Telnet, SSH. Интерфейс командной строки консоли (CLI) является промышленным стандартом. Интерпретатор CLI предоставляет список команд и ключевых слов для помощи пользователю и сокращению объема вводимых данных.
Syslog	Протокол Syslog обеспечивает передачу информационных сообщений о происходящих в системе событиях и ведение журнала событий.
Сетевые утилиты ping, traceroute	Утилиты ping и traceroute – предназначены для проверки доступности сетевых устройств и для определения маршрутов передачи данных в IP-сетях.
Управление контролируемым доступом – уровни привилегий	Маршрутизаторы поддерживают управление уровнем доступа пользователей к системе. Уровни доступа позволяют управлять зонами ответственности администраторов устройств. Уровни доступа нумеруются от 1 до 15, уровень 15 соответствует полному доступу к управлению устройством.

Аутентификация	<p>Аутентификация – это процедура проверки подлинности пользователя. Маршрутизаторы поддерживают следующие методы аутентификации:</p> <ul style="list-style-type: none"> – локальная – для аутентификации используется локальная база данных пользователей, хранящаяся на самом устройстве; – групповая – база данных пользователей хранится на сервере аутентификации. Для взаимодействия с сервером используются протоколы RADIUS и TACACS.
Сервер SSH Сервер Telnet	Функции сервера SSH и Telnet позволяют установить соединение с устройством для управления им.
Автоматическое восстановление конфигурации	Устройство поддерживает автоматическую систему восстановления конфигурации, которая предотвращает ситуации потери удаленного доступа к устройству после смены конфигурации. Если в течение заданного времени после изменения конфигурации не было введено подтверждение – произойдет автоматический откат конфигурации до предыдущего использовавшегося состояния.

2.2.7. Функции сетевой защиты

В таблице приведены функции сетевой защиты, выполняемые устройством.

Таблица 7 – Функции сетевой защиты

Зоны безопасности	Все интерфейсы маршрутизатора распределяются по зонам безопасности. Для каждой пары зон настраиваются правила, определяющие возможность или невозможность прохождения данных между зонами, правила фильтрации трафика данных.
Фильтрация данных	Для каждой пары зон безопасности составляется набор правил, которые позволяют управлять фильтрацией данных, проходящих через маршрутизатор. Командный интерфейс устройства предоставляет средства для детальной настройки правил классификации трафика и для назначения результирующего решения о пропуске трафика.

2.3. Основные технические характеристики

Основные технические параметры маршрутизатора приведены в таблице 8.

Таблица 8 – Основные технические характеристики

Общие параметры		
Пакетный процессор	R800	Broadcom XLP316L
	R200	Broadcom XLP204
	R100	Broadcom XLP104

Общие параметры		
Интерфейсы	R800	24 x Ethernet 10/100/1000Base-T 2 x 10GBase-R/1000Base-X (SFP+/SFP) 1 x Console RS-232 (RJ-45) 2 x USB 2.0 1 x Слот для SD-карты
	R200	4 x Ethernet 10/100/1000Base-T / 1000Base-X Combo 4 x Ethernet 10/100/1000Base-T 1 x Console RS-232 (RJ-45) 1 x USB 3.0 1 x USB 2.0 1 x Слот для SD-карты
	R100	4 x Ethernet 10/100/1000Base-T / 1000Base-X Combo 1 x Console RS-232 (RJ-45) 1 x USB 3.0 1 x USB 2.0 1 x Слот для SD-карты
Типы оптических трансиверов	R800	1000BASE-X SFP, 10GBASE-R SFP+
	R100	1000BASE-X SFP
	R200	
Дуплексный и полудуплексный режимы интерфейсов		- дуплексный и полудуплексный режим для электрических портов - дуплексный режим для оптических портов
Максимальная пропускная способность маршрутизатора (при аппаратной коммутации)	R800	88 Гбит/с
Скорость передачи данных	R800	- электрические интерфейсы 10/100/1000 Мбит/с - оптические интерфейсы 1/10 Гбит/с
	R100	- электрические интерфейсы 10/100/1000 Мбит/с - оптические интерфейсы 1 Гбит/с
	R200	
Количество VPN-туннелей	R800	500
	R100	250
	R200	
Максимальное количество конкурентных сессий	R800	3,13M
	R200	2,250M
	R100	1,570M
Таблица MAC-адресов	R800	16K записей
Поддержка VLAN		до 4K активных VLAN в соответствии с 802.1Q
Количество интерфейсов L3	R800 R200 R100	2000
Количество маршрутов BGPv4/BGPv6	R800	5M
	R100	2,5M
	R200	
Количество маршрутов OSPFv2/OSPFv3/IS-IS	R800	500K
	R100	300K
	R200	

Общие параметры		
Количество маршрутов RIP/RIPng		10K
Количество статических маршрутов		11K
VRF		32
Количество L3-интерфейсов		4000
Размер базы FIB	R800	1,7M
	R100	1,4M
	R200	
Соответствие стандартам		IEEE 802.3 10BASE-T Ethernet IEEE 802.3u 100BASE-T Fast Ethernet IEEE 802.3ab 1000BASE-T Gigabit Ethernet IEEE 802.3z Fiber Gigabit Ethernet ANSI/IEEE 802.3 автоопределение скорости IEEE 802.3x контроль потоков данных IEEE 802.3ad объединение каналов LACP IEEE 802.1Q виртуальные локальные сети VLAN IEEE 802.1v IEEE 802.3ac IEEE 802.3ae IEEE 802.1D IEEE 802.1w IEEE 802.1s
Управление		
Локальное управление		CLI
Удаленное управление		TELNET, SSH
Физические характеристики и условия окружающей среды		
Источники питания	R800	сеть переменного тока: 220В+/-20%, 50 Гц сеть постоянного тока: -36 .. - 72В варианты питания: - один источник питания постоянного или переменного тока; - два источника питания постоянного или переменного тока, с возможностью горячей замены.
	R100 R200	сеть переменного тока: 220В+/-20%, 50 Гц
Максимально потребляемая мощность	R800	75 Вт
	R100	20 Вт
	R200	25 Вт
Масса	R800	не более 3,6 кг
	R100 R200	не более 2,5 кг
Габаритные размеры	R800	430x352x44 мм

Общие параметры		
	R100 R200	310x240x44 мм
Интервал рабочих температур	R800 R100 R200	от -10 до +45 °С
Интервал температуры хранения		от -40 до +70 °С
Относительная влажность при эксплуатации (без образования конденсата)		не более 80%
Относительная влажность при хранении (без образования конденсата)		от 10% до 95%
Средний срок службы		не менее 15 лет

2.4. Конструктивное исполнение

В данном разделе описано конструктивное исполнение устройства. Представлены изображения передней, задней и боковых панелей устройства. Описаны разъемы, светодиодные индикаторы и органы управления.

Устройство выполнено в металлическом корпусе с возможностью установки в 19" конструктив, высота корпуса 1U.

2.4.1. Конструктивное исполнение R800

2.4.1.1. Передняя панель устройства R800

Внешний вид передней панели показан на рисунке 1.

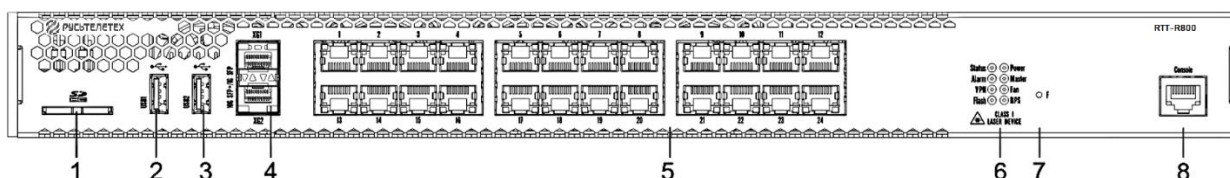


Рисунок 1– Передняя панель R800

В таблице 9 приведен перечень разъемов, светодиодных индикаторов и органов управления, расположенных на передней панели устройства R800.

Таблица 9 – Описание разъемов, индикаторов и органов управления передней панели R800

№	Элемент панели передней	Описание
1	SD	Разъем для установки SD-карт памяти.

№	Элемент панели передней	Описание
2	USB1	Порт для подключения USB-устройств.
3	USB2	Порт для подключения USB-устройств.
4	XG1, XG2	Слоты для установки трансиверов 10G SFP+/ 1G SFP.
5	[1 .. 24]	24 порта Gigabit Ethernet 10/100/1000 Base-T (RJ-45).
6	Status	Индикатор текущего состояния устройства.
	Alarm	Индикатор наличия и уровня аварии устройства.
	VPN	Индикатор наличия активных VPN-сессий.
	Flash	Индикатор активности обмена с накопителем данных - SD-картой или USB Flash.
	Power	Индикатор питания устройства.
	Master	Индикатор работы устройства в failover-режимах.
	Fan	Индикатор аварии вентиляторов.
	RPS	Индикатор резервного источника электропитания.
7	F	Функциональная кнопка для перезагрузки устройства и сброса к заводским настройкам: <ul style="list-style-type: none"> — при удержании кнопки менее 10 секунд происходит перезагрузка устройства; — при удержании кнопки более 10 секунд происходит перезагрузка устройства и сброс к заводским настройкам.
8	Console	Консольный порт RS-232 для локального управления устройством.

2.4.1.2. Задняя панель устройств R800

Внешний вид задней панели устройств R800 приведен на рисунке 2².

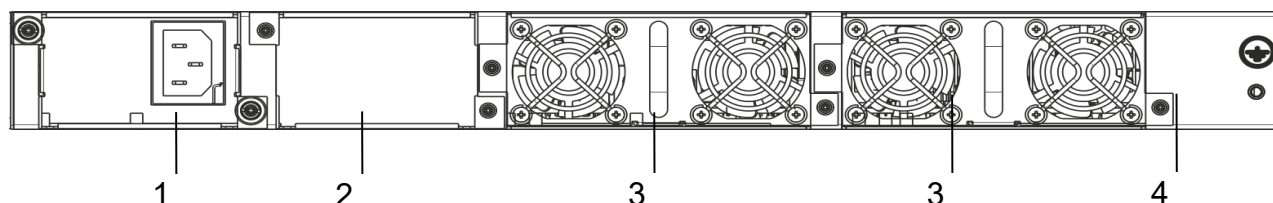


Рисунок 2 – Задняя панель R800

В таблице приведен перечень разъемов, расположенных на задней панели маршрутизатора.

² На рисунке показана комплектация маршрутизатора с одним источником питания переменного тока.

Таблица 10 – Описание разъемов задней панели маршрутизатора

№	Описание
1	Основной источник питания.
2	Место для установки резервного источника питания.
3	Съемные вентиляционные модули с возможностью горячей замены.
4	Клемма для заземления устройства.

2.4.1.3. Боковые панели устройства

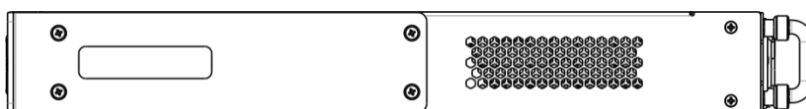


Рисунок 3 – Правая боковая панель маршрутизаторов R800

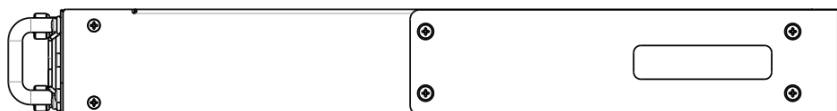


Рисунок 4 – Левая боковая панель маршрутизаторов R800

На боковых панелях устройства расположены вентиляционные решетки, которые служат для отвода тепла. Не закрывайте вентиляционные отверстия посторонними предметами. Это может привести к перегреву компонентов устройства и вызвать нарушения в его работе. Рекомендации по установке устройства расположены в разделе «Установка и подключение».

2.4.2. Конструктивное исполнение R100, R200

2.4.2.1. Передняя панель устройств R100, R200

Внешний вид передней панели R100 показан на рисунке 5.

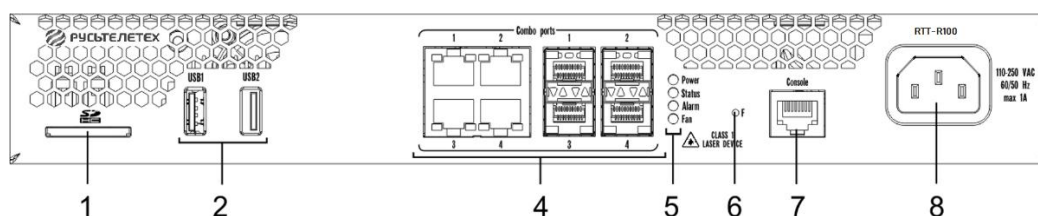


Рисунок 5 – Передняя панель R100

Внешний вид передней панели R200 показан на рисунке 6.

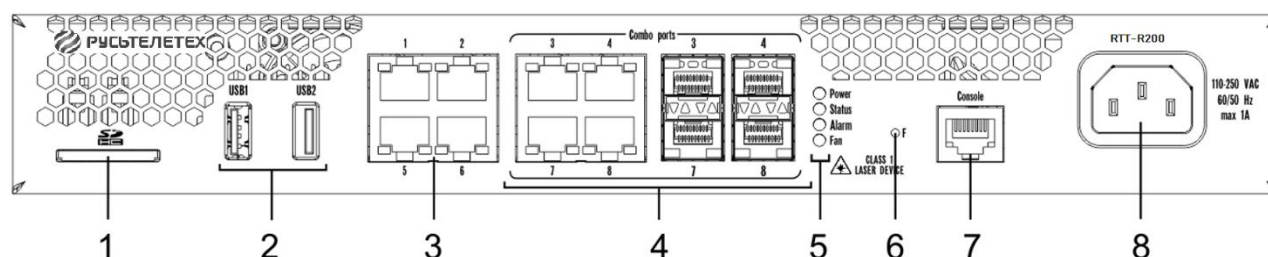


Рисунок 6 – Передняя панель R200

В таблице 11 приведен перечень разъемов, светодиодных индикаторов и органов управления, расположенных на передней панели устройств R100, R200.

Таблица 11 – Описание разъемов, индикаторов и органов управления передней панели

№	Элемент панели передней	Описание
1	SD	Разъем для установки SD-карт памяти.
2	USB1, USB2	2 порта для подключения USB-устройств.
3	[1 .. 4]	4 порта Gigabit Ethernet 10/100/1000 Base-T (RJ-45).
4	Combo Ports	4 порта Gigabit Ethernet 10/100/1000 Base-X (SFP).
5	Power	Индикатор питания устройства.
	Status	Индикатор текущего состояния устройства.
	Alarm	Индикатор наличия и уровня аварии устройства.
	Fan	Индикатор аварии вентиляторов.
6	F	Функциональная кнопка для перезагрузки устройства и сброса к заводским настройкам: <ul style="list-style-type: none"> – при удержании кнопки менее 10 секунд происходит перезагрузка устройства; – при удержании кнопки более 10 секунд происходит перезагрузка устройства и сброс к заводским настройкам.
7	Console	Консольный порт RS-232 для локального управления устройством.
8	110-250 VAC 60/50 Hz max 1A	Источник питания.

2.4.2.2. Задняя панель устройств R100, R200

Внешний вид задней панели устройств R100, R200 приведен на рисунке 7.

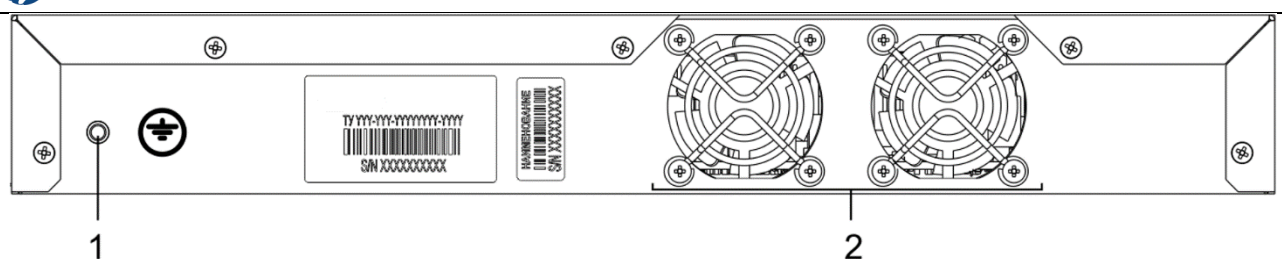


Рисунок 7 – R100, R200 задняя панель

В таблице 12 приведен перечень разъемов, расположенных на задней панели маршрутизатора.

Таблица 12 – Описание разъемов задней панели маршрутизатора

№	Описание
1	Клемма для заземления устройства.
2	Вентиляционный модуль.

2.4.2.3. Боковые панели устройства R100, R200



Рисунок 8 – Правая боковая панель маршрутизатора R100, R200



Рисунок 9 – Левая боковая панель маршрутизатора R100

2.4.3. Световая индикация

2.4.3.1. Световая индикация R800

Состояние медных интерфейсов GigabitEthernet отображается двумя светодиодными индикаторами - *LINK/ACT* зеленого цвета и *SPEED* янтарного цвета. Расположение индикаторов медных интерфейсов показано на рисунке 10. Состояние SFP-интерфейсов отображается двумя индикаторами *RX/ACT* и *TX/ACT* и указано на рисунке 11. Значения световой индикации описаны в таблицах 13 и 14.

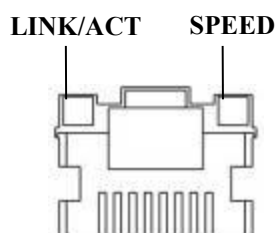


Рисунок 10 – Расположение индикаторов разъема RJ-45

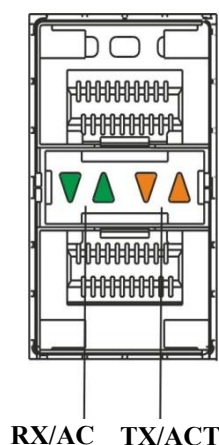


Рисунок 11 – Расположение индикаторов оптических интерфейсов

Таблица 13 – Световая индикация состояния медных интерфейсов

Свечение индикатора SPEED	Свечение индикатора LINK/ACT	Состояние интерфейса Ethernet
Выключен	Выключен	Порт выключен или соединение не установлено
Выключен	Горит постоянно	Установлено соединение на скорости 10 или 100 Мбит/с
Горит постоянно	Горит постоянно	Установлено соединение на скорости 1000 Мбит/с
X	Мигание	Идет передача данных

Таблица 14 – Световая индикация состояния SFP/SFP+ интерфейсов

Свечение индикатора RX/ACT	Свечение индикатора TX/ACT	Состояние интерфейса Ethernet
Выключен	Выключен	Порт выключен или соединение не установлено
Горит постоянно	Горит постоянно	Соединение установлено

Свечение индикатора RX/АСТ	Свечение индикатора TX/АСТ	Состояние интерфейса Ethernet
Мигание	X	Идет прием данных
X	Мигание	Идет передача данных

В следующей таблице приведено описание состояний системных индикаторов устройства и их значений.

Таблица 15 – Состояния системных индикаторов

Название индикатора	Функция индикатора	Состояние индикатора	Состояние устройства
<i>Status</i>	Индикатор текущего состояния устройства.	Зеленый	Устройство работает нормально
		Оранжевый	Устройство находится в состоянии загрузки ПО
<i>Alarm</i>	Индикатор наличия и уровня аварии устройства.	-	-
<i>VPN</i>	Индикатор наличия активных VPN-сессий.	-	-
<i>Flash</i>	Индикатор активности обмена с накопителем данных: SD-картой или USB Flash.	Оранжевый	Выполнение операций чтения/записи по команде «сору»
<i>Power</i>	Индикатор питания устройства.	Зеленый	Питание устройства в норме. Основной источник питания, если он установлен, работает нормально
		Оранжевый	Неработоспособность основного источника питания, авария или отсутствие первичной сети
		Выключен	Отказ внутренних источников питания устройства
<i>Master</i>	Индикатор работы устройства в failover-режимах.	-	-
<i>Fan</i>	Состояние вентилятора охлаждения.	Выключен	Все вентиляторы исправны
		Красный	Отказ одного или более вентиляторов. Причиной возникновения аварии может быть неработоспособность хотя бы одного из вентиляторов – остановка или пониженная частота оборотов.
<i>RPS</i>	Режим работы резервного источника питания.	Зеленый	Резервный источник установлен и исправен
		Выключен	Резервный источник не установлен
		Красный	Отсутствие первичного питания резервного источника или его неисправность

2.4.3.2. Световая индикация R100/R200

Состояние медных интерфейсов GigabitEthernet и SFP-интерфейсов отображается двумя светодиодными индикаторами - *LINK/АСТ* зеленого цвета и *SPEED* янтарного цвета. Расположение

индикаторов медных интерфейсов показано на рисунке 12. Значения световой индикации описаны в таблице 16.

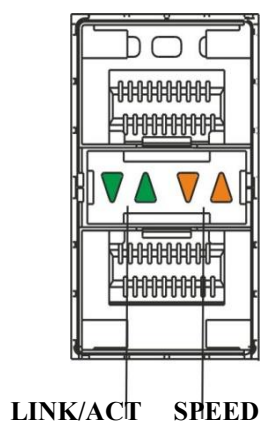


Рисунок 12 – Расположение индикаторов оптических интерфейсов

Таблица 16 – Световая индикация состояния медных интерфейсов и SFP-интерфейсов

Свечение индикатора SPEED	Свечение индикатора LINK/ACT	Состояние интерфейса Ethernet
Выключен	Выключен	Порт выключен или соединение не установлено
Выключен	Горит постоянно	Установлено соединение на скорости 10 или 100 Мбит/с
Горит постоянно	Горит постоянно	Установлено соединение на скорости 1000 Мбит/с
X	Мигание	Идет передача данных

В следующей таблице приведено описание состояний системных индикаторов устройства и их значений.

Таблица 17 – Состояния системных индикаторов

Название индикатора	Функция индикатора	Состояние индикатора	Состояние устройства
Status	Индикатор текущего состояния устройства.	Зеленый	Устройство работает нормально.
		Оранжевый	Устройство находится в состоянии загрузки ПО.
Alarm	Индикатор наличия и уровня аварии устройства ³ .	-	-
Power	Индикатор питания устройства.	Зеленый	Питание устройства в норме. Основной источник питания, если он установлен, работает нормально.
		Оранжевый	Неработоспособность основного источника питания, авария или отсутствие первичной сети.
		Выключен	Отказ внутренних источников питания устройства.
Fan	Состояние вентилятора охлаждения.	Выключен	Все вентиляторы исправны.
		Красный	Отказ одного или более вентиляторов. Причиной возникновения аварии может быть неработоспособность хотя бы одного из вентиляторов – остановка или пониженная частота оборотов.

2.5. Комплект поставки

В базовый комплект поставки R100 входят:

- маршрутизатор R100;
- кабель питания;
- кабель для подключения к порту Console (RJ-45 – DB9F);
- комплект для крепления устройства в стойку 19”;
- документация.

В базовый комплект поставки R200 входят:

- маршрутизатор R200;
- кабель питания;
- кабель для подключения к порту Console (RJ-45 – DB9F);

³ Не поддерживается в текущей версии ПО

- комплект для крепления устройства в стойку 19”;
- документация.

В базовый комплект поставки R800 входят:

- маршрутизатор R800;
- кабель питания;
- кабель для подключения к порту Console (RJ-45 – DB9F);
- комплект для крепления устройства в стойку 19”;
- документация.



По заказу покупателя для R800 в комплект поставки может быть включен модуль питания (PM160-220/12 или PM75-48/12).



По заказу покупателя в комплект поставки могут быть включены SFP/SFP+-трансиверы.

3. УСТАНОВКА И ПОДКЛЮЧЕНИЕ

В данном разделе описаны процедуры установки устройства в стойку и подключения к питающей сети.

3.1. Крепление кронштейнов

В комплект поставки устройства входят кронштейны для установки в стойку и винты для крепления кронштейнов к корпусу устройства. Для установки кронштейнов:

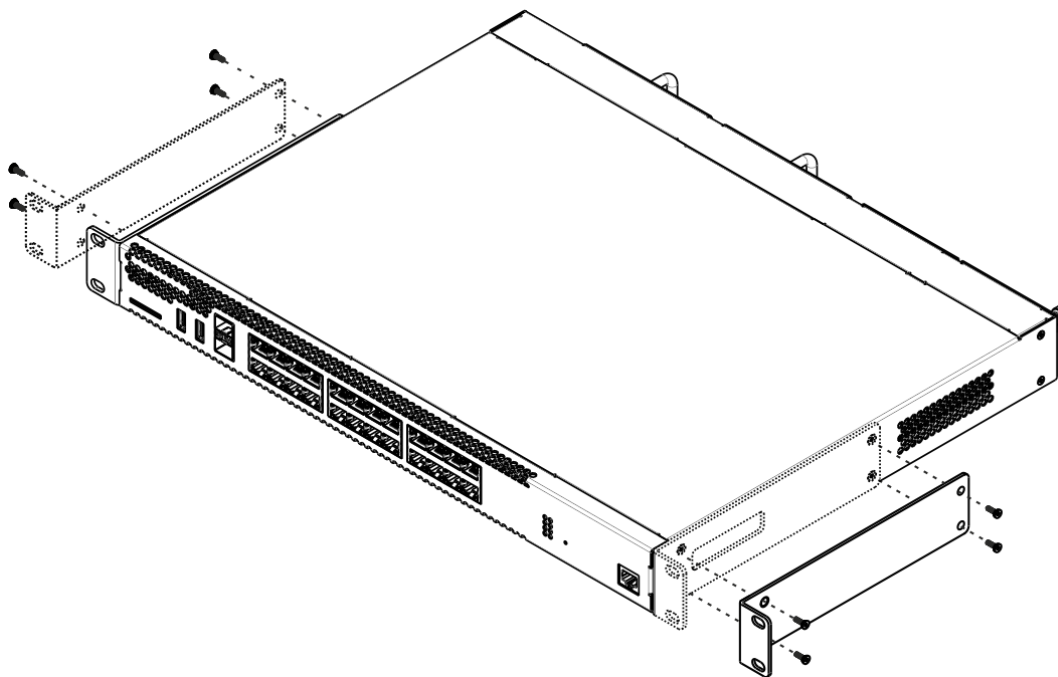


Рисунок 13 – Крепление кронштейнов

1. Совместите четыре отверстия для винтов на кронштейне с такими же отверстиями на боковой панели устройства.
2. С помощью отвертки прикрепите кронштейн винтами к корпусу.
3. Повторите действия 1, 2 для второго кронштейна.

3.2. Установка устройства в стойку

Для установки устройства в стойку:

1. Приложите устройство к вертикальным направляющим стойки.

2. Совместите отверстия кронштейнов с отверстиями на направляющих стойки. Используйте отверстия в направляющих на одном уровне с обеих сторон стойки для того, чтобы устройство располагалось горизонтально.
3. С помощью отвертки прикрепите маршрутизатор к стойке винтами.

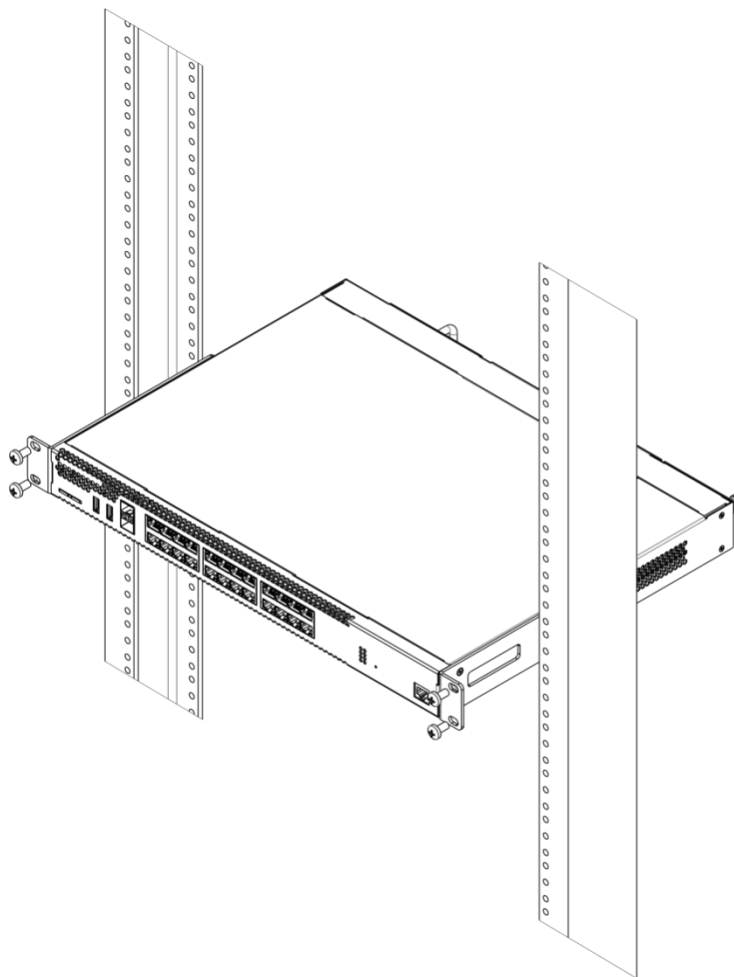


Рисунок 14 – Установка устройства в стойку



Вентиляция устройства организована по схеме фронт-тыл. На передней и боковых панелях устройства расположены вентиляционные отверстия, с задней стороны устройства расположены вентиляционные модули. Не закрывайте входные и выходные вентиляционные отверстия посторонними предметами во избежание перегрева компонентов устройства и нарушения его работы.

3.3. Установка модулей питания R800

Маршрутизаторы R800 могут работать с одним или двумя модулями питания. Установка второго модуля питания необходима в случае использования устройства в условиях, требующих повышенной надежности.

Места для установки модулей питания с электрической точки зрения равноценны. С точки зрения использования устройства, модуль питания, находящийся ближе к краю, считается основным, ближе к центру – резервным. Модули питания могут устанавливаться и извлекаться без выключения устройства. При установке или извлечении дополнительного модуля питания маршрутизатор продолжает работу без перезапуска.

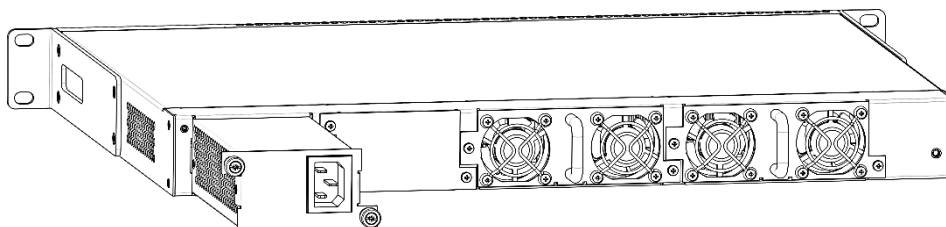


Рисунок 15 – Установка модулей питания

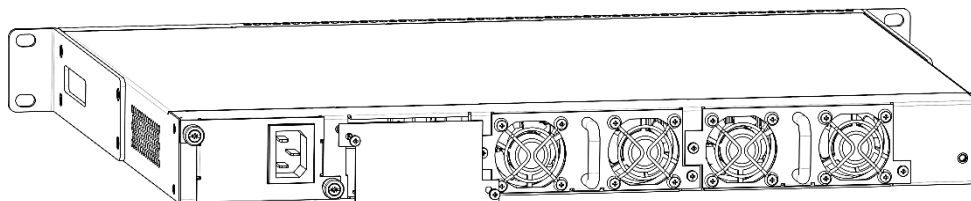


Рисунок 16 – Установка заглушки



Индикация аварии модуля питания может быть вызвана не только отказом модуля, но и отсутствием первичного питания.

Состояние модулей питания может быть проверено по индикации на передней панели маршрутизатора или по диагностике, доступной через интерфейсы управления маршрутизатором.

3.4. Подключение питающей сети

1. Прежде, чем к устройству будет подключена питающая сеть, необходимо заземлить корпус устройства. Заземление необходимо выполнять изолированным многожильным проводом. Устройство заземления и сечение заземляющего провода должны соответствовать требованиями Правил устройства электроустановок (ПУЭ).
2. Если предполагается подключение компьютера или иного оборудования к консольному порту маршрутизатора, это оборудование также должно быть надежно заземлено.

3. Подключите к устройству кабель питания. В зависимости от комплектации устройства, питание может осуществляться от сети переменного тока либо от сети постоянного тока. При подключении сети переменного тока следует использовать кабель, входящий в комплект устройства. Для подключения к сети постоянного тока используйте провод сечением не менее 1 мм².
4. Включите питание устройства и убедитесь в отсутствии аварий по состоянию индикаторов на передней панели.

3.5. Установка и удаление SFP-трансиверов



Установка оптических модулей может производиться как при выключенном, так и при включенном устройстве.

3.5.1. Установка трансивера

1. Вставьте верхний SFP-модуль в слот открытой частью разъема вниз, а нижний SFP-модуль - открытой частью разъема вверх.

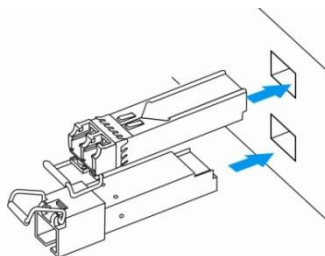


Рисунок 17 – Установка SFP-трансиверов

2. Надавите на модуль по направлению внутрь корпуса устройства до появления характерного щелчка фиксации модуля.

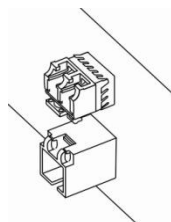


Рисунок 18 – Установленные SFP-трансиверы

3.5.2. Удаление трансивера

1. Откиньте рукоятку модуля, это приведет к разблокированию удерживающей защелки.

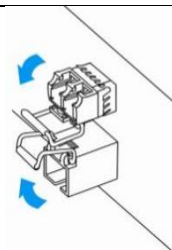


Рисунок 19 – Открытие защелки SFP-трансиверов

2. Извлеките модуль из слота.

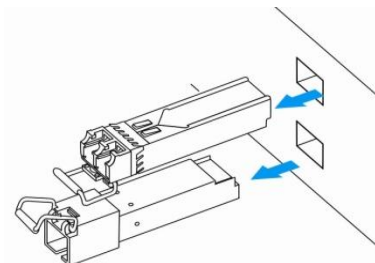


Рисунок 20 – Извлечение SFP-трансиверов

4. ИНТЕРФЕЙСЫ УПРАВЛЕНИЯ

Настройка и мониторинг устройства может осуществляться через различные интерфейсы управления.

Для доступа к устройству может использоваться сетевое подключение по протоколам Telnet и SSH или прямое подключение через консольный порт, соответствующий спецификации RS-232. При доступе по протоколам Telnet, SSH и при подключении через консольный порт для управления устройством используется интерфейс командной строки.



Заводская конфигурация содержит описание доверенной зоны trusted и IP-адрес для доступа к управлению устройством - 192.168.1.1/24.

В доверенную зону входят интерфейсы:

для R100: GigabitEthernet 1/0/2-4;

для R200: GigabitEthernet 1/0/2-8;

для R800: GigabitEthernet 1/0/2-24;

В заводской конфигурации по умолчанию создан пользователь «admin» с паролем «password».

При использовании любого из перечисленных интерфейсов управления действуют единые принципы работы с конфигурацией. Должна соблюдаться определенная, описанная здесь, последовательность изменения и применения конфигурации, позволяющая защитить устройство от некорректного конфигурирования.

4.1. Интерфейс командной строки (CLI)

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).

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

В системе может быть создано необходимое количество пользователей, права доступа задаются индивидуально для каждого из них.


Для обеспечения безопасности командного интерфейса, все команды разделены на две категории – привилегированные и непривилегированные. К привилегированным в основном относятся команды конфигурирования. К непривилегированным – команды мониторинга.

Система позволяет нескольким пользователям одновременно подключаться к устройству.

4.2. 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 18 – 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 1/0/12 Примечание: Допускается использовать сокращенное наименование, например gi1/0/12.
Порты 10Гбит/с	tengigabitethernet <UNIT>/<SLOT>/<PORT> Пример обозначения: tengigabitethernet 1/0/2 Примечание: допускается использовать сокращенное наименование, например te1/0/2.
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 – tengigabitethernet 1/0/2.123 – port-channel 1.6 Примечание: Идентификатор субинтерфейса может принимать значения [1..4094].
Q-in-Q интерфейсы	Обозначение Q-in-Q интерфейса образуется из обозначения базового интерфейса, идентификатора сервисного VLAN и идентификатора пользовательского VLAN, разделенных точкой. Примеры обозначений: – gigabitethernet 1/0/12.100.10 – tengigabitethernet 1/0/2.45.12 – port-channel 1.6.34 Примечание: Идентификатор сервисного и пользовательского VLAN 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

Тип интерфейса	Обозначение
	<ul style="list-style-type: none"> – <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: <ul style="list-style-type: none"> – loopback 4 – bridge 60 – service-port 1



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 tengigabitethernet 1/0/1-2

interface gi1/0/1-3,gi1/0/7,te1/0/1

4.3. 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 19 – 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>

Тип туннеля	Обозначение
	Пример обозначения: gre 1
SoftGRE-туннель	Обозначение SoftGRE-туннеля состоит из обозначения типа, порядкового номера туннеля и, опционально, VLAN ID виртуального интерфейса: softgre <GRE_ID>[.<VLAN>] Примеры обозначения: softgre 1, softgre 1.10
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 туннеля: vti <VTI_ID> Пример обозначения: vti 1
Логический туннель (туннель между VRF)	Обозначение логического туннеля состоит из обозначения типа и порядкового номера туннеля: lt <LT_ID> Пример обозначения: lt 1
PPPoE-туннель	Обозначение PPPoE-туннеля состоит из обозначения типа и порядкового номера туннеля: pppoe <PPPOE_ID> Пример обозначения: pppoe 1
OpenVPN-туннель	Обозначение OpenVPN-туннеля состоит из обозначения типа и порядкового номера туннеля: openvpn <OPENVPN_ID> Пример обозначения: openvpn 1
PPTP-туннель	Обозначение PPTP-туннеля состоит из обозначения типа и порядкового номера туннеля: pptp <PPTP_ID> Пример обозначения: pptp 1



Количество туннелей каждого типа зависит от модели и ПО маршрутизатора.

5. НАЧАЛЬНАЯ НАСТРОЙКА МАРШРУТИЗАТОРА

5.1. Заводская конфигурация маршрутизатора RTT

При отгрузке устройства потребителю на устройство загружена заводская конфигурация, которая включает минимально необходимые базовые настройки. Заводская конфигурация позволяет использовать маршрутизатор в качестве шлюза с функцией SNAT без необходимости применять дополнительные настройки. Кроме того, заводская конфигурация содержит настройки, позволяющие получить сетевой доступ к устройству для выполнения расширенного конфигурирования.

5.1.1. Описание заводской конфигурации

Для подключения к сетям в конфигурации описаны 2 зоны безопасности с наименованиями «Trusted» для локальной сети и «Untrusted» для публичной сети. Все интерфейсы разделены между двух зон безопасности:

1. Зона «Untrusted» предназначена для подключения к публичной сети (WAN). В этой зоне открыты порты DHCP-протокола для получения динамического IP-адреса от провайдера. Все входящие соединения из данной зоны на маршрутизатор запрещены.

В данную зону безопасности входят интерфейсы:

для R100/200: GigabitEthernet 1/0/1;

для R800: GigabitEthernet1/0/1, TengigabitEthernet1/0/1, TengigabitEthernet1/0/2.

Интерфейсы зоны объединены в один L2-сегмент через сетевой мост *Bridge 2*.

2. Зона «Trusted» предназначена для подключения к локальной сети (LAN). В этой зоне открыты порты протоколов Telnet и SSH для удаленного доступа, ICMP-протокола для проверки доступности маршрутизатора, DHCP-протокола для получения клиентами IP-адресов от маршрутизатора. Исходящие соединения из данной зоны в зону «Untrusted» разрешены.

В данную зону безопасности входят интерфейсы:

для R100: GigabitEthernet 1/0/2-4;

для R200: GigabitEthernet1/0/2-8;

для R800: GigabitEthernet1/0/2-24;

Интерфейсы зоны объединены в один L2-сегмент через сетевой мост *Bridge 1*.

На интерфейсе *Bridge 2* включен DHCP-клиент для получения динамического IP-адреса от провайдера. На интерфейсе *Bridge 1* сконфигурирован статический IP-адрес 192.168.1.1/24. Созданный IP-интерфейс выступает в качестве шлюза для клиентов локальной сети. Для клиентов локальной сети

настроен DHCP пул адресов 192.168.1.2-192.168.1.254 с маской 255.255.255.0. Для получения клиентами локальной сети доступа к Internet на маршрутизаторе включен сервис Source NAT.

Политики зон безопасности настроены следующим образом:

Таблица 20 – Описание политик зон безопасности

Зона, из которой идет трафик	Зона, в которую идет трафик	Тип трафика	Действие
Trusted	Untrusted	TCP, UDP, ICMP	разрешен
Trusted	Trusted	TCP, UDP, ICMP	разрешен
Trusted	self	TCP/23(Telnet), TCP/22(SSH), ICMP, UDP/67(DHCP Server), UDP/123(NTP)	разрешен
Untrusted	self	UDP/68(DHCP Client),	разрешен



Для обеспечения возможности конфигурирования устройства при первом включении в конфигурации маршрутизатора создана учётная запись администратора 'admin'. Настоятельно рекомендуется изменить пароль администратора при начальном конфигурировании маршрутизатора.



Для сетевого доступа к управлению маршрутизатором при первом включении в конфигурации задан статический IP-адрес на интерфейсе *Bridge 1* - 192.168.1.1/24.

5.2. Подключение и конфигурирование маршрутизатора

Маршрутизаторы серии RTT предназначены для выполнения функций пограничного шлюза и обеспечения безопасности сети пользователя при подключении ее к публичным сетям передачи данных.

Базовая настройка маршрутизатора должна включать:

- назначение IP-адресов (статических или динамических) интерфейсам, участвующим в маршрутизации данных;
- создание зон безопасности и распределение интерфейсов по зонам;
- создание политик, регулирующих прохождение данных между зонами;
- настройка сервисов, сопутствующих маршрутизации данных (NAT, Firewall и прочие).

Расширенные настройки зависят от требований конкретной схемы применения устройства и легко могут быть добавлены или изменены с помощью имеющихся интерфейсов управления.

5.2.1. Подключение к маршрутизатору

Предусмотрены следующие способы подключения к устройству:

5.2.1.1. Подключение по локальной сети Ethernet



При первоначальном старте маршрутизатор загружается с заводской конфигурацией. Описание заводской конфигурации приведено в разделе 5.1 Заводская конфигурация маршрутизатора данного руководства.

Подключите сетевой кабель передачи данных (патч-корд) к любому порту, входящему в зону «**Trusted**», и к компьютеру, предназначенному для управления.

В заводской конфигурации маршрутизатора активирован DHCP-сервер с пулом IP-адресов в подсети **192.168.1.0/24**.

При подключении сетевого интерфейса управляющего компьютера он должен получить сетевой адрес от сервера.

Если IP-адрес не получен по какой-либо причине, то следует назначить адрес интерфейса вручную, используя любой адрес, кроме 192.168.1.1, в подсети 192.168.1.0/24.

5.2.1.2. Подключение через консольный порт RS-232

При помощи кабеля RJ-45/DBF9, который входит в комплект поставки устройства, соедините порт «**Console**» маршрутизатора с портом RS-232 компьютера.

Запустите терминальную программу (например, HyperTerminal или Minicom) и создайте новое подключение. Должен быть использован режим эмуляции терминала VT100.

Выполните следующие настройки интерфейса RS-232:

Скорость: 115200 бит/с
 Биты данных: 8 бит
 Четность: нет
 Стоповые биты: 1
 Управление потоком: нет

5.2.2. Применение изменения конфигурации

Любые изменения, внесенные в конфигурацию, вступят в действие только после применения команды:

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

После применения данной команды запускается таймер "отката" конфигурации. Для остановки таймера и механизма "отката" используется команда:

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

Значение таймера "отката" по умолчанию – 600 секунд. Для изменения данного таймера используется команда:

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

<TIME> – интервал времени ожидания подтверждения конфигурации, принимает значение в секундах [120..86400].

5.2.3. Базовая настройка маршрутизатора

Процедура настройки маршрутизатора при первом включении состоит из следующих этапов:

- Изменение пароля пользователя «admin».
- Создание новых пользователей.
- Назначение имени устройства (Hostname).
- Установка параметров подключения к публичной сети в соответствии с требованиями провайдера.
- Настройка удаленного доступа к маршрутизатору.
- Применение базовых настроек.

5.2.3.1. Изменение пароля пользователя «admin»

Для защищенного входа в систему необходимо сменить пароль привилегированного пользователя «admin».



Учетная запись techsupport (до версии 1.0.7 - rustel) необходима для удаленного обслуживания сервисным центром;

Учетная запись remote - аутентификация RADIUS, TACACS+, LDAP;

Удалить пользователей admin, techsupport, remote нельзя. Можно только сменить пароль и уровень привилегий.

Имя пользователя и пароль вводится при входе в систему во время сеансов администрирования устройства.

Для изменения пароля пользователя «admin» используются следующие команды:

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

5.2.3.2. Создание новых пользователей

Для создания нового пользователя системы или настройки любого из параметров – имени пользователя, пароля, уровня привилегий, – используются команды:

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

Пример команд для создания пользователя «**fedor**» с паролем «**12345678**» и уровнем привилегий **15** и создания пользователя «**ivan**» с паролем «**password**» и уровнем привилегий **1**:

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

5.2.3.3. Назначение имени устройства

Для назначения имени устройства используются следующие команды:

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

После применения конфигурации приглашение командной строки изменится на значение, заданное параметром **<new-name>**.

5.2.3.4. Настройка параметров публичной сети

Для настройки сетевого интерфейса маршрутизатора в публичной сети необходимо назначить устройству параметры, определённые провайдером сети - IP-адрес, маска подсети и адрес шлюза по умолчанию.

Пример команд настройки статического IP-адреса для субинтерфейса **GigabitEthernet 1/0/2.150** для доступа к маршрутизатору через **VLAN 150**.

Параметры интерфейса:

IP-адрес – 192.168.16.144;
Маска подсети – 255.255.255.0;
IP-адрес шлюза по умолчанию – 192.168.16.1.

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

Для того чтобы убедиться, что адрес был назначен интерфейсу, после применения конфигурации введите следующую команду:

```
RTT# show ip interfaces
```

IP address	Interface	Type
-----	-----	-----
192.168.16.144/24	gigabitethernet 1/0/2.150	static

Провайдер может использовать динамически назначаемые адреса в своей сети. Для получения IP-адреса может использоваться протокол DHCP, если в сети присутствует сервер DHCP.

Пример настройки, предназначенной для получения динамического IP-адреса от DHCP-сервера на интерфейсе **GigabitEthernet 1/0/10**:

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

Для того чтобы убедиться, что адрес был назначен интерфейсу, введите следующую команду после применения конфигурации:

```
RTT# show ip interfaces
```

IP address	Interface	Type
-----	-----	-----
192.168.11.5/25	gigabitethernet 1/0/10	DHCP

5.2.3.5. Настройка удаленного доступа к маршрутизатору

В заводской конфигурации разрешен удаленный доступ к маршрутизатору по протоколам Telnet или SSH из зоны **«trusted»**. Для того чтобы разрешить удаленный доступ к маршрутизатору из других зон, например, из публичной сети, необходимо создать соответствующие правила в firewall.

При конфигурировании доступа к маршрутизатору правила создаются для пары зон:

source-zone – зона, из которой будет осуществляться удаленный доступ;

self – зона, в которой находится интерфейс управления маршрутизатором.

Для создания разрешающего правила используются следующие команды:

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


Пример команд для разрешения пользователям из зоны «untrusted» с IP-адресами 132.16.0.5-132.16.0.10 подключаться к маршрутизатору с IP-адресом 40.13.1.22 по протоколу SSH:

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

6. ОБНОВЛЕНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

6.1. Обновление программного обеспечения средствами системы



Для обновления программного обеспечения понадобится один из следующих серверов: TFTP, FTP, SCP. На сервер должны быть помещены файлы программного обеспечения маршрутизатора, полученные от производителя.

На маршрутизаторе хранится две копии программного обеспечения. Для обеспечения надежности процедуры обновления программного обеспечения доступна для обновления только копия, которая не была использована для последнего старта устройства.



При обновлении программного обеспечения конфигурация маршрутизатора конвертируется в соответствии с новой версией.

При загрузке маршрутизатора с более старой версией программного обеспечения, чем загруженная ранее, конфигурация не конвертируется и впоследствии удаляется.



Обновление средствами системы доступно в версии 1.0.3.69 и последующих. Обновление ПО с более ранних версий можно произвести, воспользовавшись инструкцией, приведенной в разделе 6.2.

Обновление программного обеспечения на устройстве, работающем под управлением операционной системы, выполняется в следующем порядке.

1. Подготовьте для работы выбранный сервер. Должен быть известен адрес сервера, на сервере должен быть размещен файл дистрибутивный файл программного обеспечения.
2. Маршрутизатор должен быть подготовлен к работе в соответствии с требованиями документации. Конфигурация маршрутизатора должна позволять обмениваться данными по протоколам TFTP/FTP/SCP и ICMP с сервером. При этом должна быть учтена принадлежность сервера к зонам безопасности маршрутизатора.
3. Подключитесь к маршрутизатору локально через консольный порт Console или удаленно, используя протоколы Telnet или SSH.

Проверьте доступность сервера для маршрутизатора, используя команду *ping* на маршрутизаторе. Если сервер не доступен – проверьте правильность настроек маршрутизатора и состояние сетевых интерфейсов сервера.

4. Для обновления программного обеспечения маршрутизатора введите следующую команду. В качестве параметра *<server>* должен быть указан IP-адрес используемого сервера. Для обновления с FTP или SCP-сервера потребуется ввести имя пользователя (параметр *<user>*) и пароль (параметр *<password>*). В качестве параметра *<file_name>* укажите имя файла программного обеспечения, помещенного на сервер (при использовании SCP нужно указать

полный путь – параметр *<folder>*). После ввода команды маршрутизатор скопирует файл во внутреннюю память, проверит целостность данных и сохранит его в энергонезависимую память устройства.

TFTP:

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

FTP:

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

SCP:

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

Для примера обновим основное ПО через SCP:

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

- Для того чтобы устройство работало под управлением новой версии программного обеспечения, необходимо произвести переключение активного образа. С помощью команды *show bootvar* следует выяснить номер образа, содержащего обновленное ПО.

```
RTT# show bootvar
```

Image	Version	Date	Status	After
reboot				
1	1.0.4 build 141[f812808]	date 18/02/2015 time 16:12:54	Active	*
2	1.0.4 build 141[f812808]	date 18/02/2015 time 16:12:54	Not Active	

Для выбора образа используйте команду

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

- Для обновления вторичного загрузчика (U-Boot) введите следующую команду. В качестве параметра *<server>* должен быть указан IP-адрес используемого сервера. Для обновления с FTP или SCP-сервера потребуется ввести имя пользователя (параметр *<user>*) и пароль (параметр *<password>*). В качестве параметра *<file_name>* укажите имя файла вторичного загрузчика, помещенного на сервер (при использовании SCP нужно указать полный путь – параметр *<folder>*). После ввода команды маршрутизатор скопирует файл во внутреннюю память, проверит целостность данных и сохранит его в энергонезависимую память устройства.

TFTP:

```
RTT# copy tftp://<server>:<file_name> system:boot
```

FTP:

```
RTT# copy ftp://<server>:<file_name> system:boot
```

SCP:

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

6.2. Обновление программного обеспечения из начального загрузчика

Программное обеспечение маршрутизатора можно обновить из начального загрузчика следующим образом:

1. Остановите загрузку устройства после окончания инициализации маршрутизатора загрузчиком U-Boot, нажав клавишу <Esc>.

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

2. Укажите IP-адрес TFTP-сервера:

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

3. Укажите IP-адрес маршрутизатора:

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

4. Можно сохранить окружение командой «saveenv» для будущих обновлений.

5. Запустите процедуру обновления программного обеспечения:

```
BRCM.XLP316Lite Rev B0.u-boot# run tftp_update_image1
BRCM.XLP316Lite Rev B0.u-boot# run set_bootpart_1
```

```
Using nae-0-3 device
TFTP from server 10.100.100.1; our IP address is 10.100.100.2
Filename 'RTT800/firmware'.
Load address: 0xa800000060000000
Loading: TftpStart:TftpTimeoutMsecs = 10000, TftpTimeoutCountMax = 6
#####
```

```
#####
#####
#####
#####
done
Bytes transferred = 64453909 (3d77d15 hex)
Device 0: MT29F8G08ABBCAH4 ... is now current device

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

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

6. Запустите загруженное программное обеспечение:

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

6.3. Обновление вторичного загрузчика (U-Boot)

Вторичный загрузчик занимается инициализацией NAND и маршрутизатора. При обновлении новый файл вторичного загрузчика сохраняется на flash на месте старого.

Для просмотра текущей версии загрузочного файла, работающего на устройстве, введите команду «**version**» в CLI U-Boot, также версия отображается в процессе загрузки маршрутизатора:

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

Процедура обновления ПО:

1. Остановите загрузку устройства после окончания инициализации маршрутизатора загрузчиком U-Boot, нажав клавишу <Esc>.

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

2. Укажите IP-адрес TFTP-сервера:

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

3. Укажите IP-адрес маршрутизатора:

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

4. Можно сохранить окружение командой «saveenv» для будущих обновлений.

5. Запустите процедуру обновления программного обеспечения:

```
BRCM.XLP316Lite Rev B0.u-boot# run upd_uboot или BRCM.XLP316LiteRevB0.u-boot#  
runftp_update_uboot, в зависимости от версии загрузчика.
```

```
Using nae-1 device  
TFTP from server 10.100.100.1; our IP address is 10.100.100.2  
Filename 'RTT800/u-boot.bin'.  
Load address: 0xa800000078020000  
Loading: #####  
done  
Bytes transferred = 852648 (d02a8 hex)  
SF: Detected MX25L12805D with page size 256, total 16777216 bytes  
16384 KiB MX25L12805D at 0:0 is now current device
```

6. Перезагрузите маршрутизатор:

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

7. РЕКОМЕНДАЦИИ ПО БЕЗОПАСНОЙ НАСТРОЙКЕ

Рекомендации по безопасной настройке носят общий характер и подходят для большинства инсталляций. Настоящие рекомендации в значительной степени повышают безопасность эксплуатации устройства, но не являются исчерпывающими. В зависимости от схемы применения устройства необходимо настраивать и другие параметры безопасности. В некоторых специфических случаях выполнение данных рекомендаций может привести к неработоспособности сети. При настройке устройства стоит в первую очередь следовать техническим требованиям и регламентам сетей, в которых будет эксплуатироваться данное устройство.

7.1. Общие рекомендации

- Рекомендуется всегда отключать неиспользуемые физические интерфейсы с помощью команды *shutdown*.
- Рекомендуется всегда настраивать синхронизацию системных часов с доверенными источниками сетевого времени (NTP). Алгоритм настройки NTP приведён в разделе **Настройка NTP** настоящего руководства.
- Рекомендуется отключать NTP broadcast client, включённый по умолчанию в заводской конфигурации.
- Не рекомендуется использовать команду *ip firewall disable*, отключающую межсетевое экранирование. Следует всегда назначать интерфейсам соответствующие зоны безопасности и настраивать корректные правила межсетевого экрана. Алгоритм настройки межсетевого экрана приведён в разделе **Конфигурирование Firewall** настоящего руководства.

7.2. Настройка системы логирования событий

Алгоритмы настройки системы логирования событий приведены в подразделе **Настройка Syslog** настоящего руководства.

Подробная информация о командах для настройки системы логирования событий приведена в документе «Справочник команд CLI».

7.2.1. Рекомендации

- Рекомендуется настроить хранение сообщений о событиях в файл *syslog* на устройстве и передачу этих событий на внешний *syslog*-сервер.
- Рекомендуется ограничивать размер *syslog*-файла на устройстве.
- Рекомендуется настраивать ротацию *syslog*-файлов на устройстве.
- Рекомендуется включать нумерацию сообщений *syslog*.

7.2.2. Предупреждения

- Данные, хранящиеся в файловой системе tmpsys:syslog, не сохраняются при перезагрузке устройства. Этот тип файловой системы рекомендуется использовать для хранения оперативных логов.
- Не рекомендуется использовать файловую систему flash:syslog для хранения логов, так как это может привести к преждевременному выходу из строя устройства.

7.2.3. Пример настройки

Задача:

Настроить хранение сообщений о событиях уровня info и выше в файл syslog на устройстве и настроить передачу этих событий на внешний syslog-сервер. Ограничить файл размером 512 Кбайт. Включить ротацию 3 файлов. Включить нумерацию сообщений syslog.

Решение:

Настраиваем хранение syslog-сообщений в файле:

```
rtt(config)# syslog file tmpsys:syslog/default
rtt((config-syslog-file)# severity info
rtt((config-syslog-file)# exit
```

Настраиваем ограничение размера и ротацию файлов:

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

Настраиваем передачу сообщений на внешний сервер:

```
rtt(config)# syslog host mylog
rtt(config-syslog-host)# remote-address 92.168.1.2
rtt(config-syslog-host)# transport udp
rtt(config-syslog-host)# port 514
rtt(config-syslog-host)# severity info
rtt(config-syslog-host)# exit
```

Включаем нумерацию сообщений syslog:

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

7.3. Настройка политики использования паролей

Алгоритмы настройки политики использования паролей приведены в разделе **Настройка AAA** настоящего руководства.

Подробная информация о командах для настройки политики использования паролей приведена в документе «Справочник команд CLI».

7.3.1. Рекомендации

- Рекомендуется всегда включать требования на смену пароля по умолчанию пользователя admin.
- Рекомендуется ограничивать время жизни паролей и запрещать повторно использовать, как минимум, предыдущий пароль.
- Рекомендуется выставлять требования минимальной длины пароля больше 8 символов.
- Рекомендуется выставлять требования на использование строчных и прописных букв, цифр и спецсимволов.

7.3.2. Пример настройки

Задача:

- Настроить парольную политику с обязательным требованием смены пароля по умолчанию, временем действия пароля 1 месяц и запретом на использование 12 последних паролей.
- Задать минимальную длину пароля 16 символов, максимальную — 64 символа.
- Пароль должен содержать не менее 3 прописных букв, не менее 5 строчных букв, не менее 4 цифр и не менее 2 спецсимволов. Пароль в обязательном порядке должен содержать все 4 типа символов.

Решение:

Включаем запрос на смену пароля по умолчанию для пользователя admin:

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

Устанавливаем время жизни пароля 30 дней и запрет на использование предыдущих 12 паролей:

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

Устанавливаем ограничения на длину пароля:

```
rtt(config)# security passwords min-length 16  
rtt(config)# security passwords max-length 24
```

Устанавливаем ограничения по минимальному количеству символов соответствующих типов:

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

7.4. Настройка политики AAA

Алгоритмы настройки политики AAA приведены в разделе **Настройка AAA** настоящего руководства.

Подробная информация о командах для настройки политики AAA приведена в документе «Справочник команд CLI».

7.4.1. Рекомендации

- Рекомендуется использовать ролевую модель доступа на устройство.
- Рекомендуется использовать персональные учетные записи для аутентификации на устройстве.
- Рекомендуется включать логирование вводимых пользователем команд.
- Рекомендуется использовать несколько методов аутентификации для входа на устройства через консоль, удалённого входа на устройства и повышения привилегий. Оптимальной считается комбинация из аутентификации по одному из протоколов RADIUS/TACACS/LDAP и локальной аутентификации.
- Рекомендуется отключить встроенную учётную запись **admin**.
- Рекомендуется настроить логирование изменений локальных учётных записей.
- Рекомендуется настроить логирование изменений политики AAA.

7.4.2. Предупреждения

- Встроенную учётную запись **admin** удалить нельзя, только отключить авторизацию для неё командой **no admin login enable**.
- Команда **no username admin** не удаляет пользователя **admin**, сбрасывает его конфигурацию в значения по умолчанию. После применения этой команды пользователь **admin** не будет отображаться в конфигурации.
- Команда **no password** для пользователя **admin** также не удаляет пароль пользователя **admin**, а сбрасывает его в значение по умолчанию. После применения этой команды пароль пользователя **admin** перестает отображаться в конфигурации и становится **'password'**.
- Перед отключением авторизации для пользователя **admin** в конфигурацию устройства необходимо настроить пользователя с уровнем привилегий 15 или задать ENABLE-пароль для уровня привилегий 15.

7.4.3. Пример настройки

Задача:

Настроить политику AAA:

- Для удалённого входа по протоколу SSH использовать аутентификации через RADIUS.
- Для входа через локальную консоль использовать аутентификации через RADIUS, в случае отсутствия связи с RADIUS-серверами использовать локальную аутентификацию.
- Использовать ENABLE-пароль, заданный через RADIUS, в случае отсутствия связи с RADIUS-серверами использовать локальный ENABLE-пароль.

- Установить пользователю admin пониженный уровень привилегий.
- Настроить логирование изменений локальных учётных записей.
- Настроить логирование изменений политик AAA.
- Настроить логирование вводимых команд.

Решение:

Создаем локального пользователя **local-operator** с уровнем привилегий 8:

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

Задаём локальный ENABLE-пароль:

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

Далее необходимо отключить авторизацию у пользователя admin:

```
rtt(config)# no admin login enable
```

Настраиваем связь с двумя RADIUS-серверами, основным 192.168.1.11 и резервным 192.168.2.12:

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

Настраиваем политику AAA:

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

Настраиваем логирование:

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

7.5. Настройка удалённого управления

Подробная информация о командах настройки удалённого доступа приведена в документе «Справочник команд CLI».

7.5.1. Рекомендации

- Не рекомендуется включать удалённое управление по протоколу Telnet.
- Рекомендуется использовать криптостойкие алгоритмы аутентификации sha2-512 и отключить все остальные.
- Рекомендуется использовать криптостойкие алгоритмы шифрования aes256ctr и отключить все остальные.
- Рекомендуется использовать криптостойкий алгоритм обмена ключами шифрования dh-group-exchange-sha256 и отключить все остальные.
- Рекомендуется использовать криптостойкий алгоритм верификации Host-Key для SSH rsa и отключить все остальные.
- Рекомендуется разрешить доступ к удалённому управлению устройством только с определённых IP-адресов.
- Перед началом эксплуатации рекомендуется регенерировать ключи шифрования.

7.5.2. Пример настройки

Задача:

Сгенерировать новые ключи шифрования. Использовать криптостойкие алгоритмы.

Решение:

Отключаем устаревшие и не криптостойкие алгоритмы:

```
rtt(config)# ip ssh server
rtt(config)# ip ssh authentication algorithm md5 disable
rtt(config)# ip ssh authentication algorithm md5-96 disable
rtt(config)# ip ssh authentication algorithm ripemd160 disable
rtt(config)# ip ssh authentication algorithm sha1 disable
rtt(config)# ip ssh authentication algorithm sha1-96 disable
rtt(config)# ip ssh authentication algorithm sha2-256 disable
rtt(config)# ip ssh encryption algorithm 3des disable
rtt(config)# ip ssh encryption algorithm aes128 disable
rtt(config)# ip ssh encryption algorithm aes128ctr disable
rtt(config)# ip ssh encryption algorithm aes192 disable
rtt(config)# ip ssh encryption algorithm aes192ctr disable
rtt(config)# ip ssh encryption algorithm aes256 disable
rtt(config)# ip ssh encryption algorithm arcfour disable
rtt(config)# ip ssh encryption algorithm arcfour128 disable
rtt(config)# ip ssh encryption algorithm arcfour256 disable
rtt(config)# ip ssh encryption algorithm blowfish disable
rtt(config)# ip ssh encryption algorithm cast128 disable
rtt(config)# ip ssh key-exchange algorithm dh-group-exchange-sha1 disable
rtt(config)# ip ssh key-exchange algorithm dh-group1-sha1 disable
```

```
rtt(config)# ip ssh key-exchange algorithm dh-group14-sha1 disable
rtt(config)# ip ssh key-exchange algorithm ecdh-sha2-nistp256 disable
rtt(config)# ip ssh key-exchange algorithm ecdh-sha2-nistp384 disable
rtt(config)# ip ssh key-exchange algorithm ecdh-sha2-nistp521 disable
rtt(config)# ip ssh host-key algorithm dsa disable
rtt(config)# ip ssh host-key algorithm ecdsa256 disable
rtt(config)# ip ssh host-key algorithm ecdsa384 disable
rtt(config)# ip ssh host-key algorithm ecdsa521 disable
rtt(config)# ip ssh host-key algorithm ed25519 disable
```

Генерируем новые ключи шифрования:

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

7.6. Настройка механизмов защиты от сетевых атак

Алгоритмы настройки механизмов защиты от сетевых атак приведены в разделе **Настройка логирования и защиты от сетевых атак** настоящего руководства.

Подробная информация о командах для настройки политики использования паролей приведена в документе «Справочник команд CLI».

7.6.1. Рекомендации

- Рекомендуется всегда включать защиту от ip spoofing.
- Рекомендуется всегда включать защиту от TCP-пакетов с неправильно выставленными флагами.
- Рекомендуется всегда включать защиту от фрагментированных TCP-пакетов с выставленным флагом SYN.
- Рекомендуется всегда включать защиту от фрагментированных ICMP-пакетов.
- Рекомендуется всегда включать защиту ICMP-пакетов большого размера.
- Рекомендуется всегда включать защиту от незарегистрированных IP-протоколов.
- Рекомендуется включать логирование механизма защиты от сетевых атак.

7.6.2. Пример настройки

Задача:

Настроить механизм защиты от сетевых атак в соответствии с рекомендациями.

Решение:

Включаем защиту от ip spoofing и логирование механизма защиты:

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

Включаем защиту от TCP-пакетов с неправильно выставленными флагами и логирование механизма защиты:

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

Включаем защиту от фрагментированных ICMP-пакетов и логирование механизма защиты:

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

Включаем защиту от ICMP-пакетов большого размера и логирование механизма защиты:

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

Включаем защиту от незарегистрированных IP-протоколов и логирование механизма защиты:

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

8. УПРАВЛЕНИЕ ИНТЕРФЕЙСАМИ

8.1. Настройка физического интерфейса

8.1.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Переход в режим конфигурирования функционала.	<pre> rtt(config)# interface gigabitethernet rtt(config)# interface tengigabitethernet rtt(config)# interface fourtygigabitethernet rtt(config)# interface twentyfivegigabitethernet rtt(config)# interface hundredgigabitethernet rtt(config)# interface port- channel { <ID> <UNIT>/<ID> } </pre>	<p><UNIT> – номер устройства в группе устройств [1..4].</p> <p><ID> – порядковый номер группы агрегации каналов, принимает значения [1..12].</p>
2	Включить/отключить интерфейс.	<pre> rtt(config-if-gi)# shutdown/ no shutdown </pre>	
3	Задать описание (необязательно).	<pre> rtt(config-if-gi)# description <text> </pre>	<text> – до 255 символов.
4	Задать MTU (необязательно).	<pre> rtt(config-if-gi)# mtu <count> </pre>	<p><count> – 552–10000.</p> <p>Значение по умолчанию: 1500.</p>
5	Задать скорость (необязательно).	<pre> rtt(config-if-gi)# speed 1000M/100M/10M/10G/auto </pre>	Значение по умолчанию: auto.
6	Задать MAC-адрес (необязательно).	<pre> rtt(config-if-gi)# mac-address <ADDR> </pre>	<p><ADDR> – MAC-адрес сетевого моста, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].</p>

8.1.2. Алгоритм настройки режима L3

Шаг	Описание	Команда	Ключи
1.1	Задать IP-адрес. Получить IP-адрес от DHCP-сервера.	<pre>rtt(config-if-gi)# ip address <ADDR/LEN></pre> <p>ИЛИ</p> <pre>rtt(config-if-gi)# ip address <ADDR/LEN> secondary</pre>	<p><ADDR/LEN> – IP-адрес и длина маски подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].</p> <p>Ключ secondary указывает, что настроенный адрес является дополнительным IP-адресом. Если это ключевое слово отсутствует, настроенный адрес является основным IP-адресом. Возможно указать до 7 дополнительных IP-адресов.</p>
1.2		<pre>rtt(config-if-gi)# ip address dhcp</pre>	
2.1	Задать IPv6-адрес. Получить IPv6-адрес от DHCP-сервера.	<pre>rtt(config-if-gi)# ipv6 address <ADDR/P></pre>	<p><ADDR/P> – IP-адрес и длина маски подсети, задаётся в виде <X:X:X:X::X/N> – где каждая буква X – это шестнадцатеричные значения шести 16-битных элементов адреса и N – длина префикса, принимает значения [1..128].</p>
2.2		<pre>rtt(config-if-gi)# ipv6 address dhcp</pre>	
<p>Также для физического интерфейса в режиме L3 возможно настроить:</p> <ul style="list-style-type: none"> • QoS в базовом или расширенном режимах (см. раздел Управление QoS); • проху (см. раздел Проксирование HTTP/HTTPS-трафика); • мониторинг трафика (см. разделы Настройка Netflow и Настройка sFlow); • функционал протоколов маршрутизации (см. раздел Управление маршрутизацией); • протокол VRRF (см. раздел Управление резервированием); • функционал IDS/IPS (см. раздел Настройка IPS/IDS). 			



Для использования firewall необходимо произвести его настройку (см. в разделе **Конфигурирование Firewall**).

8.1.3. Пример настройки в режиме L3

Задача:

Настроить интерфейс для прохождения трафика.



Решение:

Перейдите в режим конфигурирования, включите интерфейс, отключите firewall и задайте IPv4-адрес из диапазона 192.0.2.0/24:

```
rtt# configure
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# no shutdown
rtt(config-if-gi)# ip firewall disable
rtt(config-if-gi)# ip address 192.0.2.1/24
```

Сохраните изменения:

```
rtt(config)# commit
rtt(config)# confirm
```

На противоположной стороне выдается адрес из той же подсети.

8.2. Настройка терминции на саб-интерфейсе

Для терминирования Ethernet-фреймов конкретного VLAN на определенном физическом интерфейсе необходимо создать саб-интерфейс с указанием номера VLAN, фреймы которого будут терминироваться. При создании двух саб-интерфейсов с одинаковыми VLAN, но на разных физических/агрегированных интерфейсах, коммутация Ethernet-фреймов между данными саб-интерфейсами будет невозможна, т. к. сегменты за пределами саб-интерфейсов будут являться отдельными широковещательными доменами. Для обмена данными между абонентами разных саб-интерфейсов (даже с одинаковым VLAN-ID) будет использоваться маршрутизация, т. е. обмен данными будет происходить на третьем уровне модели OSI.

8.2.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать саб-интерфейс физического интерфейса (возможно только если физический интерфейс в режиме routerport или hybrid).	<pre>rtt(config)# interface gigabitethernet <PORT>.<S-VLAN></pre> <p>ИЛИ</p> <pre>interface tengigabitethernet <PORT>.<S-VLAN></pre> <p>ИЛИ</p> <pre>interface port-channel { <CH> <UNIT>}/<CH> } .<S-VLAN></pre>	<p><PORT> – номер физического интерфейса.</p> <p><UNIT> – номер устройства в группе устройств [1..4].</p> <p><CH> – номер агрегированного интерфейса.</p> <p><S-VLAN> – идентификатор создаваемого S-VLAN.</p> <p>Если физический интерфейс включен в bridge-group, создать саб-интерфейс будет невозможно.</p>

Шаг	Описание	Команда	Ключи
2	Задать описание саб-интерфейса (необязательно).	<code>rtt(config-if-sub) # description <DESCRIPTION></code>	<DESCRIPTION> – описание интерфейса, задаётся строкой до 255 символов.
3	Указать экземпляр VRF, в котором будет работать данный саб-интерфейс (необязательно).	<code>rtt(config-if-sub) # ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.
4	Указать IPv4/IPv6-адрес и маску подсети для конфигурируемого интерфейса или включить получение IP-адреса динамически.	<code>rtt(config-if-sub) # ip address <ADDR/LEN></code> или <code>rtt(config-if-sub) # ip address <ADDR/LEN> secondary</code>	<p><ADDR/LEN> – IP-адрес и длина маски подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].</p> <p>Ключ secondary указывает, что настроенный адрес является дополнительным IP-адресом. Если это ключевое слово отсутствует, настроенный адрес является основным IP-адресом. Возможно указать до 7 дополнительных IP-адресов.</p> <p>Дополнительные функции IPv4-адресации см. в документе «Справочник команд CLI».</p>
		<code>rtt(config-if-sub) # ipv6 address <IPV6- ADDR/LEN></code>	<p><IPV6-ADDR/LEN> – IP-адрес и префикс подсети, 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].</p> <p>Дополнительные функции IPv6-адресации см. в документе «Справочник команд CLI».</p> <p>Можно указать несколько IPv4/IPv6-адресов перечислением через запятую. Может быть назначено до 8 IPv4/IPv6-адресов на интерфейс.</p>
		<code>rtt(config-if-sub) # ip address dhcp</code>	Дополнительные функции при работе DHCP-клиента см. в документе «Справочник команд CLI».
5		<code>rtt(config-if-sub) # ip firewall disable</code>	

Шаг	Описание	Команда	Ключи
	Отключить на интерфейсе функции Firewall или включить интерфейс в зону безопасности (см. раздел Конфигурирование Firewall).	<code>rtt(config-if-sub) # security-zone <NAME></code>	<NAME> – имя зоны безопасности, задаётся строкой до 31 символа.
6	Установить интервал времени, в течение которого собирается статистика о нагрузке на саб-интерфейс (необязательно).	<code>rtt(config-if-sub) # load-average <TIME></code>	<TIME> – интервал в секундах, принимает значения [5..150].
7	Установить время жизни IPv4/IPv6 записей в ARP-таблице, изученных на данном интерфейсе (необязательно).	<code>rtt(config-if-sub) # ip arp reachable-time <TIME></code> или <code>rtt(config-if-sub) # ipv6 nd reachable-time <TIME></code>	<TIME> – время жизни динамических MAC-адресов, в миллисекундах. Допустимые значения от 5000 до 100000000 миллисекунд. Реальное время обновления записи варьируется от $[0,5;1,5] * \text{<TIME>}$.
8	Изменить размер MTU (MaximumTransmissionUnit). MTU более 1500 будет активно только если применена команда "system jumbo-frames" (необязательно).	<code>rtt(config-if-sub) # mtu <MTU></code>	<MTU> – значение MTU в байтах. Значение по умолчанию: 1500.
9	Включить запись статистики использования текущего интерфейса (необязательно).	<code>rtt(config-if-sub) # history statistics</code>	
10	Переопределить значение поля MSS (Maximum segment size) во входящих TCP-пакетах (необязательно).	<code>rtt(config-if-sub) # ip tcp adjust-mss <MSS></code> <code>rtt(config-if-sub) # ipv6 tcp adjust-mss <MSS></code>	<MSS> – значение MSS, принимает значения в диапазоне [500..1460]. Значение по умолчанию: 1460.
<p>Также для саб-интерфейса возможно настроить:</p> <ul style="list-style-type: none"> • QoS в базовом или расширенном режимах (см. раздел Управление QoS); • проху (см. раздел Проксирование HTTP/HTTPS-трафика); • мониторинг трафика (см. разделы Настройка Netflow и Настройка sFlow); • функционал протоколов маршрутизации (см. раздел Управление маршрутизацией); • протокол VRRF (см. раздел Управление резервированием); • функционал IDS/IPS (см. раздел Настройка IPS/IDS). 			

8.2.2. Пример настройки саб-интерфейса

Задача:

Настроить терминацию подсети 192.0.2.1/24 в VLAN: 828 на физическом интерфейсе gigabitethernet 1/0/1.

Решение:

Создадим саб-интерфейс для VLAN: 828:

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

Настроим IP-адрес из необходимой подсети:

```
rtt(config)# interface gigabitethernet 1/0/1.828
rtt(config-if-sub)# ip address 192.0.2.1/24
rtt(config-if-sub)# exit
```



Помимо назначения IP-адреса, на саб-интерфейсе необходимо либо отключить firewall, либо настроить соответствующую зону безопасности.

8.3. Настройка терминации на Q-in-Q интерфейсе

Q-in-Q — технология передачи пакетов с двумя 802.1q-тегами. Данная технология используется для расширения количества используемых VLAN в сети передачи данных. Внутренним тегом (InnerTag) называется 802.1q-заголовок ближе к payload. Также внутренний тег называют C-VLAN (Customer VLAN). Внешний тег (OuterTag) — это 802.1q-заголовок, добавленный к изначальному 802.1q-пакетом, также называется S-VLAN (Service VLAN). Использование двойных меток в Ethernet-фреймах описывается протоколом 802.1ad.

8.3.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать саб-интерфейс физического интерфейса (возможно только если физический интерфейс в режиме routepoint или hybrid).	<pre>rtt(config)# interface gigabitethernet <PORT>.<S-VLAN></pre> <p>или</p> <pre>interface tengigabitethernet <PORT>.<S-VLAN></pre> <p>или</p> <pre>interface port-channel { <CH> <UNIT> }/<CH> } .<S-VLAN></pre>	<p><PORT> — номер физического интерфейса.</p> <p><UNIT> — номер устройства в группе устройств [1..4].</p> <p><CH> — номер агрегированного интерфейса.</p> <p><S-VLAN> — идентификатор создаваемого S-VLAN.</p> <p>Если физический интерфейс включен в bridge-group, создать саб-интерфейс будет невозможно.</p>

Шаг	Описание	Команда	Ключи
2	Создать Q-in-Q интерфейс.	<pre>rtt(config)# interface gigabitethernet <PORT>.<S-VLAN>.<C- VLAN></pre> <p>или</p> <pre>rtt(config)# interface tengigabitethernet <PORT>.<S-VLAN>.<C- VLAN></pre> <p>или</p> <pre>rtt(config)# interface port-channel { <CH> <UNIT>/<CH> } .<S- VLAN>.<C-VLAN></pre>	<p><PORT> – номер физического интерфейса.</p> <p><UNIT> – номер устройства в группе устройств [1..4].</p> <p><CH> – номер агрегированного интерфейса.</p> <p><S-VLAN> – идентификатор создаваемого S-VLAN.</p> <p><C-VLAN> – идентификатор создаваемого C-VLAN.</p> <p>Если физический или саб-интерфейс включен в bridge-group, создать саб-интерфейс будет невозможно.</p>
3	Задать описание Q-in-Q интерфейс (необязательно).	<pre>rtt(config-if-qinq) # description <DESCRIPTION></pre>	<DESCRIPTION> – описание интерфейса, задаётся строкой до 255 символов.
4	Указать экземпляр VRF, в котором будет работать данный Q-in-Q интерфейс (необязательно).	<pre>rtt(config-if-qinq) # ip vrf forwarding <VRF></pre>	<VRF> – имя VRF, задается строкой до 31 символа.
5	Указать IPv4/IPv6-адрес и маску подсети для конфигурируемого интерфейса или включить получение IP-адреса динамически.	<pre>rtt(config-if-qinq) # ip address <ADDR/LEN></pre> <p>или</p> <pre>rtt(config-if-qinq) # ip address <ADDR/LEN> secondary</pre>	<p><ADDR/LEN> – IP-адрес и длина маски подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].</p> <p>Ключ secondary указывает, что настроенный адрес является дополнительным IP-адресом. Если это ключевое слово отсутствует, настроенный адрес является основным IP-адресом. Возможно указать до 7 дополнительных IP-адресов.</p> <p>Дополнительные функции IPv4-адресации см. в документе «Справочник команд CLI».</p>

Шаг	Описание	Команда	Ключи
		<code>rtt(config-if-qinq)# ipv6 address <IPV6-ADDR/LEN></code>	<p><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].</p> <p>Дополнительные функции IPv6-адресации см. в документе «Справочник команд CLI».</p> <p>Можно указать несколько UIPv4/IPv6-адресов перечислением через запятую. Может быть назначено до 8 IPv4/IPv6-адресов на интерфейс.</p>
		<code>rtt(config-if-qinq)# ip address dhcp</code>	Дополнительные функции при работе DHCP-клиента см. в документе «Справочник команд CLI».
6	Отключить на интерфейсе функции Firewall или включить интерфейс в зону безопасности (см. раздел Конфигурирование Firewall).	<code>rtt(config-if-qinq)# ip firewall disable</code>	
		<code>rtt(config-if-qinq)# security-zone <NAME></code>	<NAME> – имя зоны безопасности, задаётся строкой до 31 символа.
7	Установить интервал времени, в течение которого собирается статистика о нагрузке на саб-интерфейс (необязательно).	<code>rtt(config-if-sub)# load-average <TIME></code>	<TIME> – интервал в секундах, принимает значения [5..150].
8	Установить время жизни IPv4/IPv6 записей в ARP-таблице, изученных на данном интерфейсе (необязательно).	<code>rtt(config-if-sub)# ip arp reachable-time <TIME></code> или <code>rtt(config-if-sub)# ipv6 nd reachable-time <TIME></code>	<p><TIME> – время жизни динамических MAC-адресов, в миллисекундах.</p> <p>Допустимые значения от 5000 до 100000000 миллисекунд. Реальное время обновления записи варьируется от [0,5;1,5]*<TIME>.</p>
9	Изменить размер MTU (MaximumTransmissionUnit). MTU более 1500 будет активно только если применена команда "system jumbo-frames" (необязательно).	<code>rtt(config-if-sub)# mtu <MTU></code>	<p><MTU> – значение MTU в байтах.</p> <p>Значение по умолчанию: 1500.</p>
10	Включить запись статистики использования текущего интерфейса (необязательно).	<code>rtt(config-if-sub)# history statistics</code>	

Шаг	Описание	Команда	Ключи
11	Переопределить значение поля MSS (Maximum segment size) во входящих TCP-пакетах (необязательно).	<pre>rtt(config-if-sub)# ip tcp adjust-mss <MSS> rtt(config-if-sub)# ipv6 tcp adjust-mss <MSS></pre>	<p><MSS> – значение MSS, принимает значения в диапазоне [500..1460].</p> <p>Значение по умолчанию: 1460.</p>
<p>Также для Q-in-Q интерфейса возможно настроить:</p> <ul style="list-style-type: none"> • QoS в базовом или расширенном режимах (см. раздел Управление QoS); • проху (см. раздел Проксирование HTTP/HTTPS-трафика); • мониторинг трафика (см. разделы Настройка Netflow и Настройка sFlow); • функционал протоколов маршрутизации (см. раздел Управление маршрутизацией); • протокол VRRF (см. раздел Управление резервированием); • функционал IDS/IPS (см. раздел Настройка IPS/IDS). 			

8.3.2. Пример настройки Q-in-Q интерфейса

Задача:

Настроить терминацию подсети 192.0.2.1/24 комбинации C-VLAN: 741, S-VLAN: 828 на физическом интерфейсе gigabitethernet 1/0/1.

Решение:

Создадим саб-интерфейс для S-VLAN: 828:

```
rtt(config)# interface gigabitethernet 1/0/1.828
rtt(config-if-sub)# exit
```

Создадим Q-in-Q-интерфейс для S-VLAN: 741 и настроим IP-адрес из необходимой подсети:

```
rtt(config)# interface gigabitethernet 1/0/1.828.741
rtt(config-if-qinq)# ip address 192.0.2.1/24
rtt(config-if-qinq)# exit
```



Помимо назначения IP-адреса, на Q-in-Q саб-интерфейсе необходимо либо отключить firewall, либо настроить соответствующую зону безопасности.

8.4. Настройка USB-модемов

Использование USB-модемов позволяет организовать дополнительный канал связи для работы маршрутизатора. При подключении USB-модемов возможно использовать USB-концентраторы. Одновременно в системе может быть сконфигурировано до 10 USB-модемов.

8.4.1. Алгоритм настройки USB-модемов

Шаг	Описание	Команда	Ключи
1	После подключения USB-модема дождаться, когда система обнаружит подключенное устройство.		
2	Определить, какой номер устройства назначен на подключенный USB-модем.	<code>rtt# show cellular status modem</code>	В поле "USB port" будет указан идентификатор подключенного устройства.
3	Создать профиль настроек для USB-модема и перейти в режим конфигурирования профиля.	<code>rtt(config)# cellular profile <ID></code>	<ID> – идентификатор профиля настроек для USB-модема в системе [1..10].
4	Задать описание профиля настроек (необязательно).	<code>rtt(config-cellular-profile)# description <DESCRIPTION></code>	<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.
5	Задать точку доступа мобильной сети.	<code>rtt(config-cellular-profile)# apn <NAME></code>	<NAME> – точка доступа мобильной сети, задаётся строкой до 31 символа.
6	Задать имя пользователя мобильной сети (если мобильный оператор требует аутентификации по логину/паролю).	<code>rtt(config-cellular-profile)# user <NAME></code>	<NAME> – имя пользователя, задаётся строкой до 31 символа.
7	Установить пароль для пользователя мобильной сети (если мобильный оператор требует аутентификации по логину/паролю).	<code>rtt(config-user)# password ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<CLEAR-TEXT> – пароль в открытой форме, задаётся строкой [1 .. 64] символов, может включать символы [0-9a-fA-F]; <ENCRYPTED-TEXT> – пароль в зашифрованной форме, задаётся строкой [2..128] символов.
8	Активировать пользователя (если мобильный оператор требует аутентификации по логину/паролю).	<code>rtt(config-user)# enable</code>	
9	Установить номер дозвона для подключения к мобильной сети.	<code>rtt(config-cellular-profile)# number <WORD></code>	<WORD> – номер дозвона для подключения к мобильной сети, задаётся строкой до 15 символов.
10	Задать метод аутентификации пользователя в мобильной сети (необязательно).	<code>rtt(config-cellular-profile)# allowed-auth <TYPE></code>	<TYPE> – метод аутентификации пользователя в мобильной сети [none, PAP, CHAP, MSCHAP, MSCHAPv2, EAP]. Значение по умолчанию: PAP.

Шаг	Описание	Команда	Ключи
11	Ограничить возможность использования семейств IP-адресов в мобильной сети.	<code>rtt(config-cellular-profile)# ip-version { ipv4 ipv6 }</code>	ipv4 – семейство IPv4; ipv6 – семейство IPv6.
12	Создать USB-модем в конфигурации маршрутизатора и перейти в режим конфигурирования модема.	<code>rtt(config)# cellular modem <ID></code>	<ID> – идентификатор USB-модема в системе [1..10].
13	Установить режим работы беспроводного модема.	<code>rtt(config)# mode <MODE></code>	<MODE> – режим работы USB-модема [stick, hilink].
14	Задать описание модема (необязательно).	<code>rtt(config-cellular-modem)# description <DESCRIPTION></code>	<DESCRIPTION> – описание модема, задаётся строкой до 255 символов.
15	Указать экземпляр VRF, в котором будет работать данный модем (необязательно).	<code>rtt(config-cellular-modem)# ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.
16	Задать идентификатор USB-модема, назначенного системой (определен в пункте 2).	<code>rtt(config-cellular-modem)# device <WORD></code>	<WORD> – идентификатор USB-порта подключенного модема, задаётся строкой до 12 символов.
17	Назначить ранее созданный профиль настроек для USB-модема.	<code>rtt(config-cellular-modem)# profile <ID></code>	<ID> – идентификатор профиля настроек для USB-модема в системе [1..10].
18	Задать код разблокировки SIM-карты (в случае необходимости).	<code>rtt(config-cellular-modem)# pin <WORD></code>	<WORD> – код разблокировки SIM-карты [4..8]. Возможно использование только цифр.
19	Разрешить использование того или иного режима работы сети USB-модема (необязательно).	<code>rtt(config-cellular-modem)# allowed-mode <MODE></code>	<MODE> – допустимый режим работы сети USB-модема [2g, 3g, 4g]. По умолчанию: разрешены все режимы, поддерживаемые модемом.
20	Задать размер максимального принимаемого пакета (необязательно).	<code>rtt(config-cellular-modem)# mru { <MRU> }</code>	<MRU> – значение MRU, принимает значения в диапазоне [128..16383]. Значение по умолчанию: 1500.
21	Изменить максимальный размер обрабатываемых пакетов MTU (MaximumTransmissionUnit). MTU более 1500 будет активно, только если применена команда "system jumbo-frames" (необязательно).	<code>rtt(config-cellular-modem)# mtu <MTU></code>	<MTU> – значение MTU в байтах. Значение по умолчанию: 1500.

Шаг	Описание	Команда	Ключи
22	Задать предпочтительный режим работы USB-модема в мобильной сети (необязательно).	<code>rtt(config-cellular-modem)# preferred-mode { <MODE> }</code>	<MODE> – предпочтительный режим работы USB-модема [2g, 3g, 4g].
23	Отключить на интерфейсе функции Firewall или включить интерфейс в зону безопасности (см. раздел Конфигурирование Firewall).	<code>rtt(config-if-sub)# ip firewall disable</code>	
		<code>rtt(config-if-sub)# security-zone <NAME></code>	<NAME> – имя зоны безопасности, задаётся строкой до 31 символа.
24	Активировать USB-модем.	<code>rtt(config-cellular-modem)# enable</code>	
<p>Также для модема сотовой сети возможно настроить:</p> <ul style="list-style-type: none"> • QoS в базовом или расширенном режимах (см. раздел Управление QoS); • проху (см. раздел Проксирование HTTP/HTTPS-трафика); • мониторинг трафика (см. разделы Настройка Netflow и Настройка sFlow); • функционал протоколов маршрутизации (см. разделы Настройка Policy-Based Routing и Настройка MultiWAN). 			
Список поддерживаемых устройств, предоставленный производителем интегрированного драйвера см. по ссылке.			



Для полноценного функционирования модема мобильной сети необходимо дополнительно настроить маршрутизацию и функционал NAT.

8.4.2. Пример настройки

Задача:

Настроить подключение к сети Интернет, используя USB-модем.

Решение:

Для примера разберём подключение к сотовому оператору МТС.

После подключения модема необходимо дождаться, когда система обнаружит устройство. Определим порт устройства, который был назначен на подключённый USB-модем:

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

Создадим профиль настроек для USB-модема:

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

Зададим APN, который требует провайдер, или иной необходимый адрес. Ниже показан пример подключения к APN МТС:

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

При необходимости задаём имя пользователя, пароль, номер дозвона и метод аутентификации:

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

Перейдём к конфигурированию USB-модема и зададим идентификатор, соответствующий порту устройства, который был определён в начале:

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

Назначим соответствующий профиль настроек и активируем модем:

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

8.5. Настройка PPP через E1

PPP (англ. Point-to-Point Protocol) — двухточечный протокол канального уровня, используется для установления прямой связи между двумя узлами сети. Может обеспечить аутентификацию соединения, шифрование и сжатие данных.

Для установления PPP-соединения через поток E1 необходимо наличие медиаконвертера TOPGATE-WAN-E1 в маршрутизаторе RTT.



На маршрутизаторах R800 не поддерживается работа модулей TOPGATE-WAN-E1 с аппаратной версией (hardware revision) 812.

8.5.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
Предварительная настройка:			

Шаг	Описание	Команда	Ключи
1	Необходимо включить поддержку Jumbo-фреймов. Для вступления изменений в силу требуется перезагрузка устройства.	<code>rtt(config)# system jumbo-frames</code>	
Настройка физического интерфейса:			
2	Необходимо выбрать интерфейс, в котором установлен TOPGATE-WAN-E1.	<code>rtt(config)# interface gigabitethernet 1/0/3</code>	
3	Перевести физический интерфейс в режим коммутации.	<code>rtt(config-if-gi)# mode switchport</code>	
4	Задать режим работы интерфейса E1.	<code>rtt(config-if-gi)# switchport mode e1</code>	
5	Задать источник синхронизации (необязательно).	<code>rtt(config-if-gi)# switchport e1 clock source <SOURCE></code>	<p><SOURCE> – источник синхронизации:</p> <ul style="list-style-type: none"> • internal (по умолчанию) – синхронизироваться с внутренним источником; • line – синхронизироваться с линейным сигналом.
6	Указать размер MTU (Maximum Transmission Unit) для физических интерфейсов.	<code>rtt(config-if-gi)# mtu <MTU></code>	<p><MTU> – значение MTU, для E1- и Multilink-интерфейсов необходимо указать значения в диапазоне [1510..9600].</p>
7	Задать хэш-алгоритм проверки кадра (необязательно).	<code>rtt(config-if-gi)# switchport e1 crc <FCS></code>	<p><FCS> – последовательность проверки кадра:</p> <ul style="list-style-type: none"> • 16 (по умолчанию) – FCS16; • 32 – FCS32.
8	Задать проверку на наличие ошибок при передаче (необязательно).	<code>rtt(config-if-gi)# switchport e1 framing <CRC></code>	<p><CRC> – проверка циклической избыточности:</p> <ul style="list-style-type: none"> • crc-4 – использовать алгоритм CRC-4; • no-crc4 (по умолчанию) – не использовать проверку.
9	Задать инвертирование передаваемых бит (необязательно).	<code>rtt(config-if-gi)# switchport e1 invert data</code>	

Шаг	Описание	Команда	Ключи
10	Задать тип линейного кодирования (необязательно).	<code>rtt(config-if-gi)# switchport e1 linecode <CODE></code>	<CODE> – тип линейного кодирования; <ul style="list-style-type: none">• ami – чередующейся полярностью импульсов;• hdb3 (по умолчанию) – двухполярный код высокой плотности порядка 3.
11	Задать количество тайм-слотов.	<code>rtt(config-if-gi)# switchport e1 timeslots <RANGE></code>	<RANGE> – количество тайм-слотов.
12	Использовать E1 как единую сущность, без тайм-слотов (необязательно).	<code>rtt(config-if-gi)# switchport e1 unframed</code>	
Настройка интерфейса E1:			
13	Необходимо выбрать интерфейс E1.	<code>rtt(config)# interface e1 1/<SLOT>/1</code>	<SLOT> – номер слота.
14	Указать IPv4 и маску подсети для конфигурируемого интерфейса.	<code>rtt(config-if-e1)# ip address <ADDR/LEN></code>	<ADDR/LEN> – IP-адрес и длина маски подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].
15	Отключить на интерфейсе функции Firewall или включить интерфейс в зону безопасности (см. раздел Конфигурирование Firewall).	<code>rtt(config-if-e1)# ip firewall disable</code>	
		<code>rtt(config-if-e1)# security-zone <NAME></code>	<NAME> – имя зоны безопасности, задаётся строкой до 31 символа.
Дополнительные настройки PPP для E1:			
16	Включить CHAP-аутентификацию для PPP (необязательно).	<code>rtt(config-if-e1)# ppp authentication chap</code>	
17	Задать имя маршрутизатора, которое отправляется удаленной стороне для прохождения CHAP-аутентификации (необязательно).	<code>rtt(config-if-e1)# ppp chap hostname <NAME></code>	<NAME> – имя маршрутизатора.
18	Задать пароль для аутентификации (необязательно).	<code>rtt(config-if-e1)# ppp chap password ascii-text <CLEAR-TEXT></code>	<CLEAR-TEXT> – пароль в открытой форме, задаётся строкой [1 .. 64] символов, может включать символы [0-9a-fA-F].
19	Включить игнорирование аутентификации (необязательно).	<code>rtt(config-if-e1)# ppp chap refuse</code>	

Шаг	Описание	Команда	Ключи
20	Задать имя пользователя для аутентификации (необязательно).	<code>rtt(config-if-e1)# ppp chap username <NAME></code>	<NAME> – имя пользователя.
21	Разрешается принимать от соседа любой ненулевой IP-адрес в качестве локального IP-адреса (необязательно).	<code>rtt(config-if-e1)# ppp ipcp accept-address</code>	
22	Задать IP-адрес, который отправляется удаленной стороне для последующего его присвоения (необязательно).	<code>rtt(config-if-e1)# ppp ipcp remote-address <ADDR></code>	<ADDR> – IP-адрес удаленного шлюза.
23	Задать количество попыток отправки Configure-Request пакетов, прежде чем удаленный пир будет признан неспособным ответить (необязательно).	<code>rtt(config-if-e1)# ppp max-configure <VALUE></code>	<VALUE> – количество попыток.
24	Задать количество попыток отправки Configure-NAK пакетов, прежде чем будут подтверждены все опции (необязательно).	<code>rtt(config-if-e1)# ppp max-failure <VALUE></code>	<VALUE> – количество попыток.
25	Задать количество попыток отправки Terminate-Request пакетов, прежде чем сессия будет прервана (необязательно).	<code>rtt(config-if-e1)# ppp max-terminate <VALUE></code>	<VALUE> – количество попыток.
26	Задать размер MRU (Maximum Receive Unit) для интерфейса (необязательно).	<code>rtt(config-if-e1)# ppp mru <MRU></code>	<MRU> – значение MRU.
27	Задается интервал времени в секундах, по истечении которого маршрутизатор отправляет keepalive-сообщение (необязательно).	<code>rtt(config-if-e1)# ppp timeout keepalive <TIME></code>	<TIME> – время в секундах.
28	Задается интервал, по истечении которого маршрутизатор повторяет запрос на установление сессии (необязательно).	<code>rtt(config-if-e1)# ppp timeout retry <TIME></code>	<TIME> – время в секундах.
Включение интерфейса E1 в Multilink PPP:			
29	Добавить в MLPPP-группу (необязательно).	<code>rtt(config-if-e1)# ppp multilink-group <GROUP-ID></code>	<GROUP-ID> – номер группы.
30	Включение режима MLPPP (необязательно).	<code>rtt(config-if-e1)# ppp multilink</code>	

8.5.2. Пример конфигурации

Задача:

Настроить PPP-соединение со встречной стороной с IP-адресом 192.0.2.2/24 через TOPGATE-WAN-E1, используя 1-8 канальные интервалы для передачи данных.



Решение:

Предварительно необходимо настроить system jumbo-frames, сохранить изменения в конфигурации и перезагрузить маршрутизатор:

```

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

Настроим физический интерфейс gigabitethernet 1/0/3, в котором установлен TOPGATE-WAN-E1:

- Укажем mtu не менее 1510.
- Переведем интерфейс в режим работы e1.
- Укажем канал e1 – 0.
- Укажем интервал каналов e1 – 1-8.

```

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

Настроим интерфейс e1:

```

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

Информацию о физическом состоянии e1 можно узнать с помощью следующей команды:

```

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

```

Информацию о состоянии e1-интерфейса можно узнать с помощью следующей команды:

```

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

```

8.6. Настройка MLPPP

Multilink PPP (MLPPP) предоставляет собой агрегированный канал, включающий в себя методы для распространения трафика через несколько физических каналов, имея одно логическое соединение. Этот вариант позволяет расширить пропускную способность и обеспечивает балансировку нагрузки.

8.6.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Настроить группу агрегации.	<code>rtt(config)# interface multilink <IF></code>	<IF> – наименование интерфейса.
2	Указать описание конфигурируемой группы агрегации (необязательно).	<code>rtt(config-if- multilink)# description <DESCRIPTION></code>	<DESCRIPTION> – описание группы агрегации, задаётся строкой до 255 символов.

Шаг	Описание	Команда	Ключи
3	Задать интервал времени, за который усредняется статистика о нагрузке на группе агрегации (необязательно).	<code>rtt(config-if-multilink)# load-average <TIME></code>	<TIME> – интервал в секундах, принимает значения [5..150]. Значение по умолчанию: 5.
4	Указать размер MTU (Maximum Transmission Unit) для группы агрегации (необязательно). MTU более 1500 будет активно только в случае применения команды "system jumbo-frames".	<code>rtt(config-if-multilink)# mtu <MTU></code>	<MTU> – значение MTU, принимает значения в диапазоне [1280..1500]. Значение по умолчанию: 1500.
5	Включить CHAP-аутентификацию.	<code>rtt(config-if-multilink)# ppp authentication chap</code>	
6	Включить игнорирование аутентификации (необязательно).	<code>rtt(config-if-multilink)# ppp chap refuse</code>	
7	Указать имя маршрутизатора, которое отправляется удаленной стороне для прохождения CHAP-аутентификации.	<code>rtt(config-if-multilink)# ppp chap hostname <NAME></code>	<NAME> – имя маршрутизатора, задаётся строкой до 31 символа
8	Указать пароль, который отправляется удаленной стороне вместе с именем маршрутизатора для прохождения CHAP-аутентификации.	<code>rtt(config-if-multilink)# ppp chap password ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<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.
9	Разрешить принимать от соседа любой ненулевой IP-адрес в качестве локального IP-адреса (необязательно).	<code>rtt(config-if-multilink)# ppp ipcp accept-address</code>	
10	Установить IP-адрес, который отправляется удаленной стороне для последующего его присвоения.	<code>rtt(config-if-multilink)# ppp iccp remote-address <ADDR></code>	<ADDR> – IP-адрес удаленного шлюза.
11	Указать пользователя для аутентификации удаленной стороны и перейти в режим конфигурирования указанного пользователя.	<code>rtt(config-if-multilink)# chap username <NAME></code>	<NAME> – имя пользователя, задаётся строкой до 31 символа.

Шаг	Описание	Команда	Ключи
12	Установить пароль в открытой или зашифрованной форме определенному пользователю для аутентификации удаленной стороны.	<code>rtt(config-ppp-user) # password ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<CLEAR-TEXT> – пароль в открытой форме, задается строкой [8 .. 64] символов, может включать символы [0-9a-fA-F]. <ENCRYPTED-TEXT> – пароль в зашифрованной форме, задается строкой [16..128] символов.
13	Установить количество попыток отправки Configure-Request пакетов, прежде чем удаленный пир будет признан неспособным ответить (необязательно).	<code>rtt(config-if-multilink) # ppp max-configure <VALUE></code>	<VALUE> – время в секундах, принимает значения [1..255]. Значение по умолчанию: 10.
14	Установить количество попыток выслать Configure-NAK пакеты, прежде чем будут подтверждены все опции (необязательно).	<code>rtt(config-if-multilink) # ppp max-failure <VALUE></code>	<VALUE> – время в секундах, принимает значения [1..255].
15	Установить количество попыток выслать Terminate-Request пакеты, прежде чем сессия будет прервана (необязательно).	<code>rtt(config-if-multilink) # ppp max-terminate <VALUE></code>	<VALUE> – время в секундах, принимает значения [1..255]. Значение по умолчанию: 2.
16	Указать размер MRU (Maximum Receive Unit) для интерфейса.	<code>rtt(config-if-multilink) # ppp mru <MRU></code>	<MRU> – значение MRU, принимает значения в диапазоне [128..1485]. Значение по умолчанию: 1500.
17	Указать интервал времени в секундах, по истечении которого маршрутизатор отправляет keepalive-сообщение (необязательно).	<code>rtt(config-if-multilink) # ppp timeout keepalive <TIME></code>	<TIME> – время в секундах, принимает значения [1..32767]. Значение по умолчанию: 10.
18	Установить интервал времени в секундах, по истечении которого маршрутизатор повторяет запрос на установление сессии (необязательно).	<code>rtt(config-if-multilink) # ppp timeout retry <TIME></code>	<TIME> – время в секундах, принимает значения [1..255]. Значение по умолчанию: 3.
19	Определить максимальный размер пакета для MLPP-интерфейса.	<code>rtt(config-if-multilink) # mrru <MRRU></code>	<MRRU> – максимальный размер принимаемого пакета для MLPP-интерфейса, принимает значение в диапазоне [1500..10000].
20	Привязать порт e1 к физическому интерфейсу.	<code>rtt(config-if-gi) # switchport e1 <SLOT></code>	<SLOT> – идентификатор слота, принимает значение в диапазоне [0..3].
21	Перевести физический порт в режим работы с SFPe1-модулем.	<code>rtt(config-if-gi) # switchport mode e1</code>	

Шаг	Описание	Команда	Ключи
22	Включить режим MLPPP на E1-интерфейсе.	<code>rtt(config-if-e1)# ppp multilink</code>	
23	Включить E1-интерфейс в группу агрегации.	<code>rtt(config-if-e1)# ppp multilink-group <GROUP-ID></code>	<GROUP-ID> – идентификатор группы, принимает значение [1..4].

8.6.2. Пример настройки

Задача:

Настроить MLPPP-соединение с встречной стороной с IP-адресом 192.0.2.2/24 через интерфейсы e1 1/0/1 и e1 1/1/1. Для построения агрегированного канала PPP используются интерфейсы gi 1/0/3 и gi 1/0/4, в которые вставлены TOPGATE-WAN-E1.



Решение:

Предварительно необходимо настроить system jumbo-frames, сохранить изменения в конфигурации и перезагрузить маршрутизатор:

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

Настроим физические интерфейсы gigabitethernet 1/0/3-4, в которых установлены TOPGATE-WAN-E1. При настройке физических интерфейсов укажем mtu не менее 1510, переведем интерфейс в режим работы e1, укажем канал e1:

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

```
rtt(config-if-gi)# mode switchport
rtt(config-if-gi)# mtu 1510
rtt(config-if-gi)# switchport mode e1
rtt(config-if-gi)# switchport e1 slot 1
rtt(config-if-gi)# switchport e1 timeslots 1-31
rtt(config-if-gi)# exit
```

Настроим интерфейс multilink:

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

Привяжем интерфейсы E1 к Multilink PPP. При настройке e1-интерфейса необходимо указать multilink-group и включить multilink:

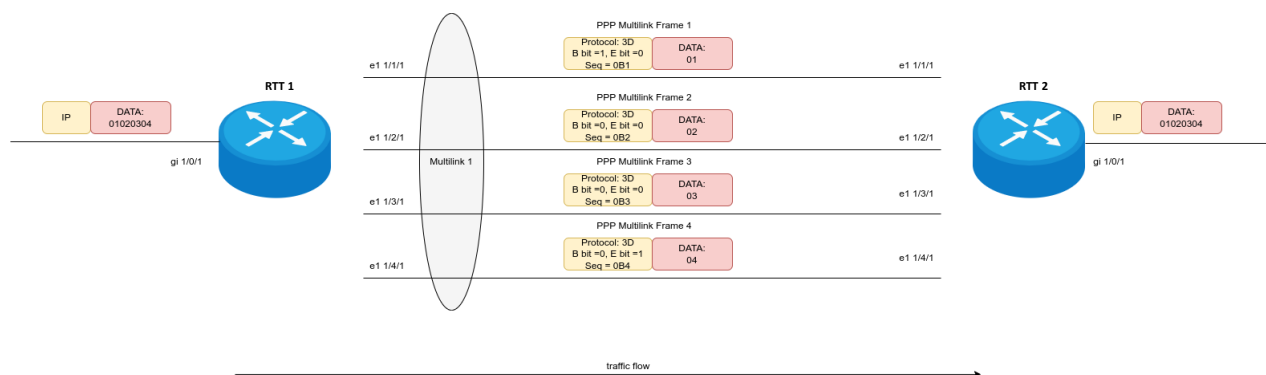
```
rtt(config)# interface e1 1/0/1
rtt(config-if-e1)# ppp multilink-group 3
rtt(config-if-e1)# ppp multilink
rtt(config-if-e1)# exit
rtt(config)# interface e1 1/1/1
rtt(config-if-e1)# ppp multilink-group 3
rtt(config-if-e1)# ppp multilink
rtt(config-if-e1)# exit
```

Информацию о состоянии multilink-интерфейса можно узнать с помощью следующей команды:

```
rtt# show interfaces status multilink 3
Interface 'mu3' status information:
  Description:      --
  Operational state: Up
  Administrative state: Up
  Track ID:         0
  Supports broadcast: No
  Supports multicast: Yes
  MTU:              1492
  MAC address:      none
  Last change:      1 hour, 4 minutes and 2 seconds
  Mode:             routerport
  Bandwidth:        3968 Kbps
  Member links:     2 active, 0 inactive
    * e1 1/0/1:     Up 23 minutes and 58 seconds ago
    e1 1/1/1:       Up 30 minutes and 36 seconds ago
```

8.6.3. Фрагментация трафика

По умолчанию каждый пакет, который будет отправлен через мультилинк, подлежит фрагментации. Пакет делится на равные части пропорционально количеству линков в мультилинке. Каждый фрагмент инкапсулируется в ML PPP. На противоположной стороне пакет собирается из фрагментов в свое первоначальное состояние.



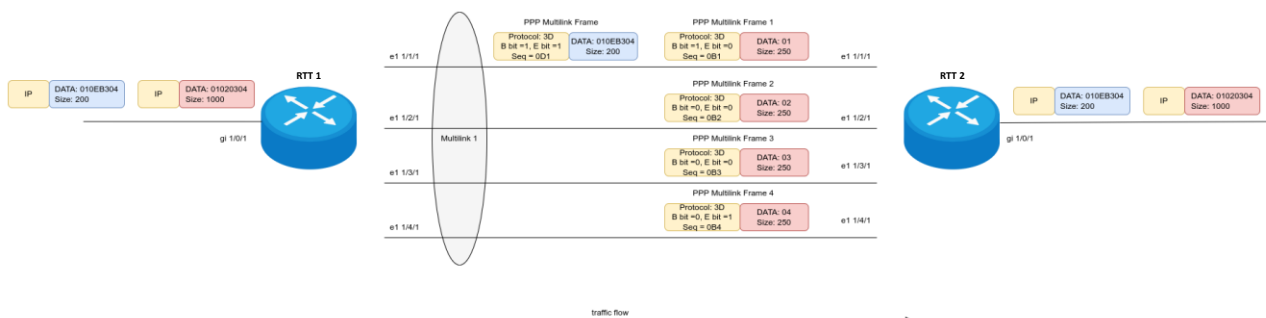
В случае передачи большого количество пакетов небольшого размера (например, голосовой трафик) такое поведение порождает избыточное количество служебного трафика, что негативно влияет на утилизацию канала, а также является одной из причин возникновения задержек при передаче.

Например, пакет размером 80 байт, проходящий через мультилинк, в котором 8 участников, будет разделен на 8 фрагментов по 10 байт. На каждый фрагмент будет добавлен ML PPP заголовок (4 байта).

Для оптимизации такого поведения можно указать минимальный размер, все пакеты меньше которого не будут подлежать фрагментации.

```
RTT# config
rtt(config)# interface multilink 1
rtt(config-if-multilink)# min-frag-size 200
rtt(config)# do commit
rtt(config)# do confirm
```

После включения данного функционала, пакеты, размер которых меньше 200 байт, не будут фрагментированы. Пакеты с большим размер будут подлежать фрагментации.



9. УПРАВЛЕНИЕ ТУННЕЛИРОВАНИЕМ

9.1. Настройка GRE-туннелей

GRE (англ. Generic Routing Encapsulation — общая инкапсуляция маршрутов) — протокол туннелирования сетевых пакетов. Его основное назначение — инкапсуляция пакетов сетевого уровня сетевой модели OSI в IP-пакеты. GRE может использоваться для организации VPN на 3 уровне модели OSI.

В маршрутизаторе RTT реализованы статические неуправляемые GRE-туннели, то есть туннели создаются вручную путем конфигурирования на локальном и удаленном узлах. Параметры туннеля для каждой из сторон должны быть взаимосогласованными или переносимые данные не будут декапсулироваться партнером.

9.1.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Сконфигурировать L3-интерфейс, от которого будет строиться GRE-туннель.		
2	Создать GRE-туннель и перейти в режим его конфигурирования.	<code>rtt(config)# tunnel gre <INDEX></code>	<p><INDEX> – идентификатор туннеля в диапазоне:</p> <ul style="list-style-type: none"> • для R100/200 – [1..250]; • для R800 – [1..500].
3	Указать экземпляр VRF, в котором будет работать данный GRE-туннель (необязательно).	<code>rtt(config-gre)# ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.
4	Указать имя VRF от IP-интерфейса которого будет строиться данный GRE-туннель (необязательно).	<code>rtt(config-gre)# tunnel-source vrf <VRF></code>	<p><VRF> – имя экземпляра VRF, задается строкой до 31 символа.</p> <p>Без указания ключа "vrf" и имени экземпляра VRF будет использоваться IP-интерфейс глобальной конфигурации.</p>
5	Указать описание конфигурируемого туннеля (необязательно).	<code>rtt(config-gre)# description <DESCRIPTION></code>	<DESCRIPTION> – описание туннеля, задается строкой до 255 символов.
6	Установить локальный IP-адрес для установки туннеля.	<code>rtt(config-gre)# local address <ADDR></code>	<ADDR> – IP-адрес локального шлюза, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Шаг	Описание	Команда	Ключи
		<code>rtt(config-gre) # local interface <IF></code>	<IF> – интерфейс, от IP-адреса которого устанавливается туннель.
7	Установить удаленный IP-адрес для установки туннеля.	<code>rtt(config-gre) # remote address <ADDR></code>	<ADDR> – IP-адрес локального шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
8	Указать режим инкапсуляции для GRE-туннеля.	<code>rtt(config-gre) # mode <MODE></code>	<p><MODE> – режим инкапсуляции для GRE-туннеля:</p> <ul style="list-style-type: none"> ip – инкапсуляция IP-пакетов в GRE; ethernet – инкапсуляция Ethernet-фреймов в GRE. <p>Значение по умолчанию: ip</p>
9	Установить IP-адрес локальной стороны туннеля (только в режиме ip).	<code>rtt(config-gre) # ip address <ADDR/LEN> [unit <ID>]</code> или <code>rtt(config-gre) # ip address <ADDR/LEN> secondary [unit <ID>]</code>	<p><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].</p> <p><ID> – номер юнита, принимает значения [1..4].</p> <p>Ключ secondary указывает, что настроенный адрес является дополнительным IP-адресом. Если это ключевое слово отсутствует, настроенный адрес является основным IP-адресом. Возможно указать до 7 дополнительных IP-адресов.</p> <p>Дополнительные функции IPv4-адресации см. в документе «Справочник команд CLI».</p>
10	Назначить широковещательный домен для инкапсуляции в GRE-пакеты данного туннеля (только в режиме ethernet).	<code>rtt(config-gre) # bridge-group <BRIDGE-ID></code>	<p><BRIDGE-ID> – идентификационный номер моста, принимает значения в диапазоне:</p> <ul style="list-style-type: none"> для R100/200 – [1..250]; для R800 – [1..500].
11	Включить GRE-туннель в зону безопасности и настроить правила	<code>rtt(config-gre) # security-zone<NAME></code>	<NAME> – имя зоны безопасности, задаётся строкой до 12 символов.

Шаг	Описание	Команда	Ключи
	взаимодействия между зонами или отключить firewall (см. раздел Конфигурирование Firewall).	<code>rtt(config-gre) # ip firewall disable</code>	
12	Указать размер MTU (Maximum Transmission Unit) для туннеля (необязательно). MTU более 1500 будет активно, только если применена команда "system jumbo-frames".	<code>rtt(config-gre) # mtu <MTU></code>	<MTU> – значение MTU, принимает значения в диапазоне: – [1280..10000]. Значение по умолчанию: 1500.
13	Указать значение времени жизни TTL для туннельных пакетов (необязательно).	<code>rtt(config-gre) # ttl <TTL></code>	<TTL> – значение TTL, принимает значения в диапазоне [1..255]. Значение по умолчанию: наследуется от инкапсулируемого пакета.
14	Указать DSCP для использования в IP-заголовке инкапсулирующего пакета (необязательно).	<code>rtt(config-gre) # dscp <DSCP></code>	<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63]. Значение по умолчанию: наследуется от инкапсулируемого пакета.
15	Разрешить передачу ключа (key) в туннельном заголовке GRE (в соответствии с RFC 2890) и установить значение ключа. Настраивается только с обеих сторон туннеля (необязательно).	<code>rtt(config-gre) # key <KEY></code>	<KEY> – значение KEY, принимает значения в диапазоне [1..2000000]. Значение по умолчанию: ключ не передаётся.
16	Включить вычисление контрольной суммы и занесение её в GRE-заголовки отправляемых пакетов. При этом на удаленной стороне необходимо включить проверку контрольной суммы (необязательно).	<code>rtt(config-gre) # local checksum</code>	
17	Включить проверку наличия и соответствия значений контрольной суммы в заголовках принимаемых GRE-пакетов. При этом на удаленной стороне необходимо включить вычисление контрольной суммы (необязательно).	<code>rtt(config-gre) # remote checksum</code>	
18	Включить проверку доступности удаленного шлюза туннеля (необязательно).	<code>rtt(config-gre) # keepalive enable</code>	

Шаг	Описание	Команда	Ключи
19	Изменить время ожидания keepalive пакетов от встречной стороны (необязательно).	<code>rtt(config-gre) # keepalive timeout <TIME></code>	<TIME> – время в секундах, принимает значения в диапазоне [1..32767]. Значение по умолчанию: 10.
20	Изменить количество попыток проверки доступности удаленного шлюза туннеля (необязательно).	<code>rtt(config-gre) # keepalive retries <VALUE></code>	<VALUE> – количество попыток, принимает значения в диапазоне [1..255]. Значение по умолчанию: 5.
21	Указать IP-адрес для работы механизма keepalive (обязательно в режиме ethernet).	<code>rtt(config-gre) # keepalive dst- address <ADDR></code>	<ADDR> – IP-адрес для проверки работоспособности GRE-туннеля.
22	Изменить интервал времени, за который усредняется статистика о нагрузке на туннеле (необязательно).	<code>rtt(config-gre) # load-average <TIME></code>	<TIME> – интервал в секундах, принимает значения [5..150]. Значение по умолчанию: 5.
23	Включить отправку snmp-trap о включении/отключении туннеля.	<code>rtt(config-gre) # snmp init-trap</code>	
24	Включить механизм перезапроса IP-адресов по протоколу DHCP на указанных интерфейсах при отключении GRE-туннеля по keepalive (необязательно).	<code>rtt(config-gre) # keepalive dhcp dependent-interface <IF></code>	<IF> – физический/логический интерфейс, на котором включено получение IP-адреса по DHCP.
25	Задать интервал времени между отключением GRE-туннеля и перезапросом IP-адреса на интерфейсе/интерфейсах, указанных командой keepalive dhcp dependent-interface (необязательно).	<code>rtt(config-gre) # keepalive dhcp link- timeout <SEC></code>	<SEC> – интервал между отключением GRE-туннеля и перезапросом IP-адреса по DHCP на интерфейсах.
26	Переопределить значение поля MSS (Maximum segment size) во входящих TCP-пакетах (необязательно).	<code>rtt(config-gre) # ip tcp adjust-mss <MSS></code>	<MSS> – значение MSS, принимает значения в диапазоне [500..1460]. Значение по умолчанию: 1460.
27	Включить запись статистики использования текущего туннеля (необязательно).	<code>rtt(config-gre) # history statistics</code>	
28	Активировать туннель.	<code>rtt(config-gre) # enable</code>	

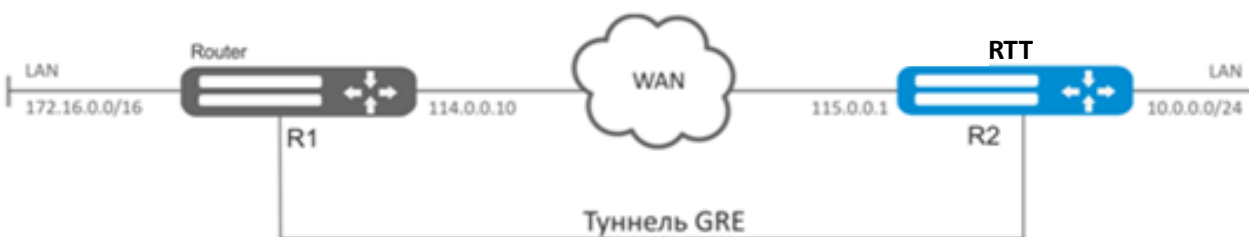
Шаг	Описание	Команда	Ключи
<p>Также для GRE-туннеля возможно настроить:</p> <ul style="list-style-type: none"> • QoS в базовом или расширенном режимах (см. раздел Управление QoS); • проху (см. раздел Проксирование HTTP/HTTPS-трафика); • мониторинг трафика (см. разделы Настройка Netflow и Настройка sFlow); • функционал протоколов маршрутизации (см. раздел Управление маршрутизацией). 			

9.1.2. Пример настройки IP-GRE-туннеля

Задача:

Организовать L3-VPN между офисами компании через IP-сеть, используя для туннелирования трафика протокол GRE.

- в качестве локального шлюза для туннеля используется IP-адрес 115.0.0.1;
- в качестве удаленного шлюза для туннеля используется IP-адрес 114.0.0.10;
- IP-адрес туннеля на локальной стороне 25.0.0.1/24.



Решение:

Предварительно на маршрутизаторах должны быть настроены интерфейсы для связи с сетью WAN разрешено получение пакетов протокола GRE из зоны безопасности, в которой работают интерфейсы, подключенные к сети WAN.

Создадим туннель GRE 10:

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

Укажем локальный и удаленный шлюз (IP-адреса интерфейсов, граничащих с WAN):

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

Укажем IP-адрес туннеля 25.0.0.1/24:

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

Также туннель должен принадлежать к зоне безопасности, для того чтобы можно было создать правила, разрешающие прохождение трафика в firewall. Принадлежность туннеля к зоне задается следующей командой:

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

Включим туннель:

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

На маршрутизаторе должен быть создан маршрут до локальной сети партнера. В качестве интерфейса назначения указываем ранее созданный туннель GRE:

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

После применения настроек трафик будет инкапсулироваться в туннель и отправляться партнеру, независимо от наличия GRE-туннеля и правильности настроек с его стороны.

Опционально для GRE-туннеля можно указать следующие параметры:

- Включить вычисление и включение в пакет контрольной суммы заголовка GRE и инкапсулированного пакета для исходящего трафика:

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

- Включить проверку наличия и корректности контрольной суммы GRE для входящего трафика:

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

- Указать уникальный идентификатор:

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

- Указать значение DSCP, MTU, TTL:

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

- Включить и настроить механизм keepalive:

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

Состояние туннеля можно посмотреть командой:

```
rtt# show tunnels status gre 10
```

Счетчики входящих и отправленных пакетов можно посмотреть командой:

```
rtt# show tunnels counters gre 10
```

Конфигурацию туннеля можно посмотреть командой:

```
rtt# show tunnels configuration gre 10
```

Настройка туннеля IPv4-over-IPv4 производится аналогичным образом.

При создании туннеля необходимо в firewall разрешить протокол GRE (47).

9.2. Настройка DMVPN

DMVPN (Dynamic Multipoint Virtual Private Network) — технология для создания виртуальных частных сетей, с возможностью динамического создания туннелей между узлами. Преимуществом данного решения является высокая масштабируемость и легкость настройки при подключении филиалов к головному офису. DMVPN используется в топологии Hub-and-Spoke, и позволяет строить прямые VPN-туннели Spoke-to-Spoke в дополнение к обычным Spoke-to-Hub туннелям. Это означает, что филиалы смогут общаться друг с другом напрямую, без необходимости прохождения трафика через Hub.

Чтобы установить такое соединение, клиенты (NHC) по зашифрованному IPsec-туннелю отправляют соответствие своего внутреннего (туннельного) адреса и внешнего (NBMA) адреса на NHRP-сервер (NHS). Когда клиент захочет соединиться с другим NHC, он посылает на сервер запрос, чтобы узнать его внешний адрес. Получив ответ от сервера, клиент теперь самостоятельно может устанавливать соединение с удалённым филиалом.

9.2.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Проверить доступность «внешних» IP-адресов, находящихся на физических интерфейсах.		
2	Подготовить IPsec-туннели для работы совместно с динамическими GRE-туннелями.		См. раздел Алгоритм настройки Policy-based IPsec VPN
3	Создать GRE-туннель и перейти в режим его конфигурирования.	<code>rtt(config)# tunnel gre <INDEX></code>	<INDEX> – идентификатор туннеля.
4	Перевести GRE-туннель в режим multipoint.	<code>rtt(config-gre)# multipoint</code>	
5	Указать экземпляр VRF, в котором будет работать данный GRE-туннель (необязательно).	<code>rtt(config-gre)# ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.

Шаг	Описание	Команда	Ключи
6	Указать имя VRF от IP-интерфейса которого будет строиться данный GRE-туннель (необязательно).	<code>rtt(config-gre)# tunnel-source vrf <VRF></code>	<p><VRF> – имя экземпляра VRF, задается строкой до 31 символа.</p> <p>Без указания ключа "vrf" и имени экземпляра VRF, будет использоваться IP-интерфейс глобальной конфигурации.</p>
7	Установить локальный IP-адрес для установки туннеля.	<code>rtt(config-gre)# local address <ADDR></code>	<ADDR> – IP-адрес локального шлюза, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		<code>rtt(config-gre)# local interface <IF></code>	<IF> – интерфейс, от IP-адреса которого устанавливается туннель.
8	Задать IP-адрес на туннеле. В качестве альтернативы можно настроить DHCP-клиент для получения IP-адреса от DHCP-сервера.	<code>rtt(config-gre)# ip address <ADDR/LEN></code> или <code>rtt(config-gre)# ip address <ADDR/LEN> secondary</code>	<p><ADDR/LEN> – IP-адрес и длина маски подсети, задается в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].</p> <p>Ключ secondary указывает, что настроенный адрес является дополнительным IP-адресом. Если это ключевое слово отсутствует, настроенный адрес является основным IP-адресом. Возможно указать до 7 дополнительных IP-адресов.</p>
		<code>rtt(config-gre)# ip address dhcp</code>	
9	Установить открытый пароль для NHRP-пакетов (необязательно).	<code>rtt(config-gre)# ip nhrp authentication <WORD></code>	<WORD> – пароль в открытой форме, задается строкой [1..8] символов, может включать символы [0-9a-fA-F].
10	Указать время, в течение которого на NHS будет существовать запись о данном клиенте (не обязательно).	<code>rtt(config-gre)# ip nhrp holding-time <TIME></code>	<p><TIME> – время в секундах, в течение которого на сервере будет существовать запись о данном клиенте, принимает значения [1..65535].</p> <p>Значение по умолчанию: 7200.</p>
11	Задать соответствие «внутреннего» туннельного адреса с «внешним» NBMA-адресом.	<code>rtt(config-gre)# ip nhrp map <ADDR> <ADDR></code>	<ADDR> – IP-адрес задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Шаг	Описание	Команда	Ключи
12	Задать «логический (туннельный)» адрес NHRP-сервера.	<code>rtt(config-gre)# ip nhrp nhs <ADDR></code>	<ADDR> – адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть AAA – DDD принимает значения [0..255].
13	Определить адресата мультикастного трафика.	<code>rtt(config-gre)# ip nhrp multicast { dynamic nhs <ADDR> }</code>	<ul style="list-style-type: none"> • dynamic – отправлять на все пиры, с которыми есть соединение; • nhs – отправлять на все статические сконфигурированные сервера; <p><ADDR> – отправлять на специфически сконфигурированный адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p>
14	Включить возможность отправки NHRP Traffic Indication пакетов. Выполняется на NHS (необязательно).	<code>rtt(config-gre)# ip nhrp redirect</code>	
15	Включить возможность создания кратчайших маршрутов. Выполняется на NHS (необязательно).	<code>rtt(config-gre)# ip nhrp shortcut</code>	
16	Привязать IPsec-VPN к mGRE-туннелю (необязательно).	<code>rtt(config-gre)# ip nhrp ipsec <WORD> { static dynamic }</code>	<p><WORD> – имя VPN, задаётся строкой до 31 символа;</p> <ul style="list-style-type: none"> • static – статическое соединение, применяется для связи с NHS; • dynamic – динамически устанавливаемое соединение, конфигурируется для связи между NHS.
17	Включить передачу имени NHRP-группы NHRP-соседям в процессе обмена NHRP-сообщениями (необязательно).	<code>rtt(config-gre)# ip nhrp attribute group <WORD></code>	<WORD> – имя группы NHRP, задаётся строкой [1..40] символов, не принимает символы [^#].

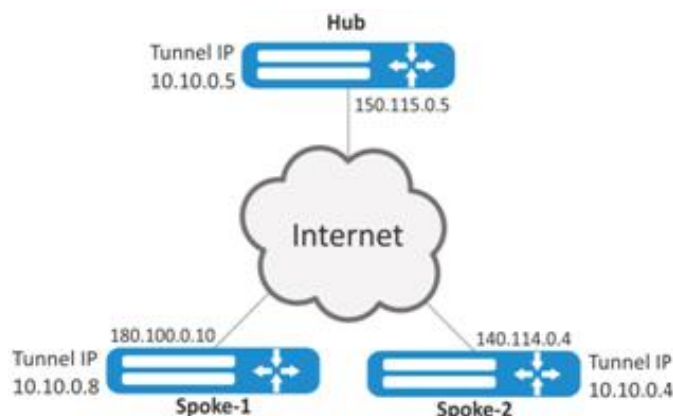
Шаг	Описание	Команда	Ключи
18	Задать соответствие группы NHRP, полученной от NHRP-соседа в процессе обмена NHRP-сообщениями, и политики QoS, которая будет применена к исходящему в сторону этого NHRP-соседа трафика (необязательно).	<code>rtt(config-gre)# ip nhrp map group <GROUP> service-policy output <POLICY></code>	<GROUP> – имя группы NHRP, задаётся строкой [1..40] символов, не принимает символы [^#]; <POLICY> – имя QoS-политики, задаётся строкой [1..31] символов.
19	Включить работу протокола NHRP.	<code>rtt(config-gre)# ip nhrp enable</code>	
20	Организовать IP-связность посредством протокола динамической маршрутизации.		

Остальные настройки аналогичны настройкам статичного GRE-туннеля (см. раздел **Настройка GRE-туннелей**).

9.2.2. Пример настройки 1

Задача:

Организовать DMVPN между офисами компании, используя mGRE-туннели, NHRP (Next Hop Resolution Protocol), протокол динамической маршрутизации (BGP), IPsec. В данном примере будет HUB-маршрутизатор и два филиала. HUB – это DMVPN-сервер (NHS), а филиалы – DMVPN-клиенты (NHC).



Hub внешний IP-адрес — 150.115.0.5;

Spoke-1 внешний IP-адрес — 180.100.0.10;

Spoke-2 внешний IP-адрес — 140.114.0.4.

Параметры IPsec VPN:

IKE:

- группа Диффи-Хэллмана: 2;
- алгоритм шифрования: AES128;
- алгоритм аутентификации: SHA1.

IPsec:

- алгоритм шифрования: AES128;
- алгоритм аутентификации: SHA1.

Предварительная настройка интерфейсов:

```
rtt-Hub# configure
rtt-Hub(config)# int gil/0/1
rtt-Hub(config-if-gi)# ip address 150.115.0.5/24
rtt-Hub(config-if-gi)# ip firewall disable
rtt-Hub(config-if-gi)# exit
rtt-Hub(config)# ip route 0.0.0.0/0 150.115.0.1
rtt-Hub(config)# do commit
rtt-Hub(config)# do confirm

rtt-Spoke-1# configure
rtt-Spoke-1(config)# int gil/0/1
rtt-Spoke-1(config-if-gi)# ip address 180.100.0.10/24
rtt-Spoke-1(config-if-gi)# ip firewall disable
rtt-Spoke-1(config-if-gi)# exit
rtt-Spoke-1(config)# ip route 0.0.0.0/0 180.100.0.1
rtt-Spoke-1(config)# do commit
rtt-Spoke-1(config)# do confirm

rtt-Spoke-2# configure
rtt-Spoke-2(config)# int gil/0/1
rtt-Spoke-2(config-if-gi)# ip address 140.114.0.4/24
rtt-Spoke-2(config-if-gi)# ip firewall disable
rtt-Spoke-2(config)# exit
rtt-Spoke-2(config)# ip route 0.0.0.0/0 140.114.0.1
rtt-Spoke-2(config)# do commit
rtt-Spoke-2(config)# do confirm
```

Решение:

1. Конфигурирование Hub.

Создадим туннель GRE:

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

Укажем IP-адрес интерфейса, граничащего с ISP:

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

Зададим значение MTU:


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

Установим значение ttl:

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

Отключим firewall:

```
rtt(config-gre)# ip firewall disable
```

Зададим IP-адрес GRE-туннеля:

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

Переведём GRE-туннель в multipoint режим для возможности соединения с несколькими точками:

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

Перейдём к настройке NHRP. Настроим отправку мультикастовых рассылок в динамически узнаваемые адреса:

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

Произведём настройку протокола динамической маршрутизации для Hub. В примере это будет BGP.

Поскольку в примере используется eBGP необходимо явно разрешить анонсирование подсетей:

```
rtt(config)# route-map PERMIT_ALL
rtt(config-route-map)# rule 1
rtt(config)# router bgp 65005
rtt(config-bgp)# neighbor 10.10.0.8
rtt(config-bgp-neighbor)# remote-as 65008
rtt(config-bgp-neighbor)# enable
rtt(config-bgp-neighbor)# address-family ipv4 unicast
rtt(config-bgp-neighbor-af)# route-map PERMIT_ALL out
rtt(config-bgp-neighbor-af)# enable
rtt(config-bgp-neighbor-af)# exit
rtt(config-bgp-neighbor)# exit
rtt(config-bgp)# neighbor 10.10.0.4
rtt(config-bgp-neighbor)# remote-as 65004
rtt(config-bgp-neighbor)# enable
rtt(config-bgp-neighbor)# address-family ipv4 unicast
rtt(config-bgp-neighbor-af)# route-map PERMIT_ALL out
rtt(config-bgp-neighbor-af)# enable
rtt(config-bgp-neighbor-af)# exit
rtt(config-bgp-neighbor)# exit
rtt(config-bgp)# address-family ipv4 unicast
rtt(config-bgp-af)# exit
rtt(config-bgp)# enable
rtt(config-bgp)# exit
```

Произведём настройку IPsec для Hub:

```
rtt(config)# security ike proposal IKEPROP
rtt(config-ike-proposal)# encryption algorithm aes128
rtt(config-ike-proposal)# dh-group 2
rtt(config-ike-proposal)# exit
rtt(config)# security ike policy IKEPOLICY
rtt(config-ike-policy)# pre-shared-key ascii-text encrypted 8CB5107EA7005AFF
rtt(config-ike-policy)# proposal IKEPROP
rtt(config-ike-policy)# exit
rtt(config)# security ike gateway IKEGW
rtt(config-ike-gw)# ike-policy IKEPOLICY
rtt(config-ike-gw)# local address 150.115.0.5
rtt(config-ike-gw)# local network 150.115.0.5/32 protocol gre
rtt(config-ike-gw)# remote address any
rtt(config-ike-gw)# remote network any
rtt(config-ike-gw)# mode policy-based
rtt(config-ike-gw)# exit
rtt(config)# security ipsec proposal IPSECPROP
rtt(config-ipsec-proposal)# encryption algorithm aes128
rtt(config-ipsec-proposal)# exit
rtt(config)# security ipsec policy IPSECPOLICY
rtt(config-ipsec-policy)# proposal IPSECPROP
rtt(config-ipsec-policy)# exit
rtt(config)# security ipsec vpn IPSECVPN
rtt(config-ipsec-vpn)# mode ike
rtt(config-ipsec-vpn)# type transport
rtt(config-ipsec-vpn)# ike establish-tunnel route
rtt(config-ipsec-vpn)# ike gateway IKEGW
rtt(config-ipsec-vpn)# ike ipsec-policy IPSECPOLICY
rtt(config-ipsec-vpn)# enable
```

Привяжем IPsec к GRE-туннелю, чтобы клиенты могли устанавливать шифрованное соединение:

```
rtt(config)# tunnel gre 5
rtt(config-gre)# ip nhrp ipsec IPSECVPN dynamic
```

Включим работу NHRP и сам туннель:

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

2. Конфигурирование Spoke

Проведём стандартную настройку DMVPN на туннеле:

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

Указываем, сколько времени будет храниться запись о клиенте на сервере:

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

Указываем туннельный адрес NHS:

```
rtt(config-gre)# ip nhrp nhs 10.10.0.5
```

Зададим соответствие туннельному адресу – реальный:

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

Настроим мультикастовую рассылку на NHRP-сервер:

```
rtt(config-gre)# ip nhrp multicast nhs
```

Произведём настройку BGP для spoke. Поскольку в примере используется eBGP необходимо явно разрешить анонсирование подсетей:

```
rtt(config)# route-map PERMIT_ALL
rtt(config-route-map)# rule 1
rtt(config)# router bgp 65008
rtt(config-bgp)# neighbor 10.10.0.5
rtt(config-bgp-neighbor)# remote-as 65005
rtt(config-bgp-neighbor)# enable
rtt(config-bgp-neighbor)# address-family ipv4 unicast
rtt(config-bgp-neighbor-af)# route-map PERMIT_ALL out
rtt(config-bgp-neighbor-af)# enable
rtt(config-bgp-neighbor-af)# exit
rtt(config-bgp-neighbor)# exit
rtt(config-bgp)# address-family ipv4 unicast
rtt(config-bgp-af)# exit
rtt(config-bgp)# enable
```

Произведём настройку IPsec. При создании шлюза протокола IKE для NHS, укажем конкретные адреса назначения. А при создании шлюза IKE для NHC – адрес назначения будет any:

```
rtt(config)# security ike proposal IKEPROP
rtt(config-ike-proposal)# encryption algorithm aes128
rtt(config-ike-proposal)# dh-group 2
rtt(config-ike-proposal)# exit
rtt(config)# security ike policy IKEPOLICY
rtt(config-ike-policy)# pre-shared-key ascii-text encrypted 8CB5107EA7005AFF
rtt(config-ike-policy)# proposal IKEPROP
rtt(config-ike-policy)# exit
rtt(config)# security ike gateway IKEGW_HUB
rtt(config-ike-gw)# ike-policy IKEPOLICY
rtt(config-ike-gw)# local address 180.100.0.10
rtt(config-ike-gw)# local network 180.100.0.10/32 protocol gre
rtt(config-ike-gw)# remote address 150.115.0.5
rtt(config-ike-gw)# remote network 150.115.0.5/32 protocol gre
rtt(config-ike-gw)# mode policy-based
rtt(config-ike-gw)# exit
rtt(config)# security ike gateway IKEGW_SPOKE
rtt(config-ike-gw)# ike-policy IKEPOLICY
rtt(config-ike-gw)# local address 180.100.0.10
```

```
rtt(config-ike-gw)# local network 180.100.0.10/32 protocol gre
rtt(config-ike-gw)# remote address any
rtt(config-ike-gw)# remote network any
rtt(config-ike-gw)# mode policy-based
rtt(config-ike-gw)# exit
rtt(config)# security ipsec proposal IPSECPROP
rtt(config-ipsec-proposal)# encryption algorithm aes128
rtt(config-ipsec-proposal)# exit
rtt(config)# security ipsec policy IPSECPOLICY
rtt(config-ipsec-policy)# proposal IPSECPROP
rtt(config-ipsec-policy)# exit
rtt(config)# security ipsec vpn IPSECVPN_HUB
rtt(config-ipsec-vpn)# mode ike
rtt(config-ipsec-vpn)# type transport
rtt(config-ipsec-vpn)# ike establish-tunnel route
rtt(config-ipsec-vpn)# ike gateway IKEGW_HUB
rtt(config-ipsec-vpn)# ike ipsec-policy IPSECPOLICY
rtt(config-ipsec-vpn)# enable
rtt(config-ipsec-vpn)# exit
rtt(config)# security ipsec vpn IPSECVPN_SPOKE
rtt(config-ipsec-vpn)# mode ike
rtt(config-ipsec-vpn)# type transport
rtt(config-ipsec-vpn)# ike establish-tunnel route
rtt(config-ipsec-vpn)# ike gateway IKEGW_SPOKE
rtt(config-ipsec-vpn)# ike ipsec-policy IPSECPOLICY
rtt(config-ipsec-vpn)# enable
rtt(config-ipsec-vpn)# exit
```

Привяжем IPsec к GRE-туннелю для возможности установления зашифрованного соединения с сервером и с другими клиентами сети:

```
rtt(config)# tunnel gre 8
rtt(config-gre)# ip nhrp ipsec IPSECVPN_HUB static
rtt(config-gre)# ip nhrp ipsec IPSECVPN_SPOKE dynamic
```

Включим работу NHRP и сам туннель:

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

Сохраним конфигурацию:

```
rtt# commit
rtt# confirm
```



Для использования firewall необходимо произвести его настройку. В данном примере firewall был отключён.

Состояние NHRP-записей можно посмотреть командой:

```
rtt# show ip nhrp peers
```

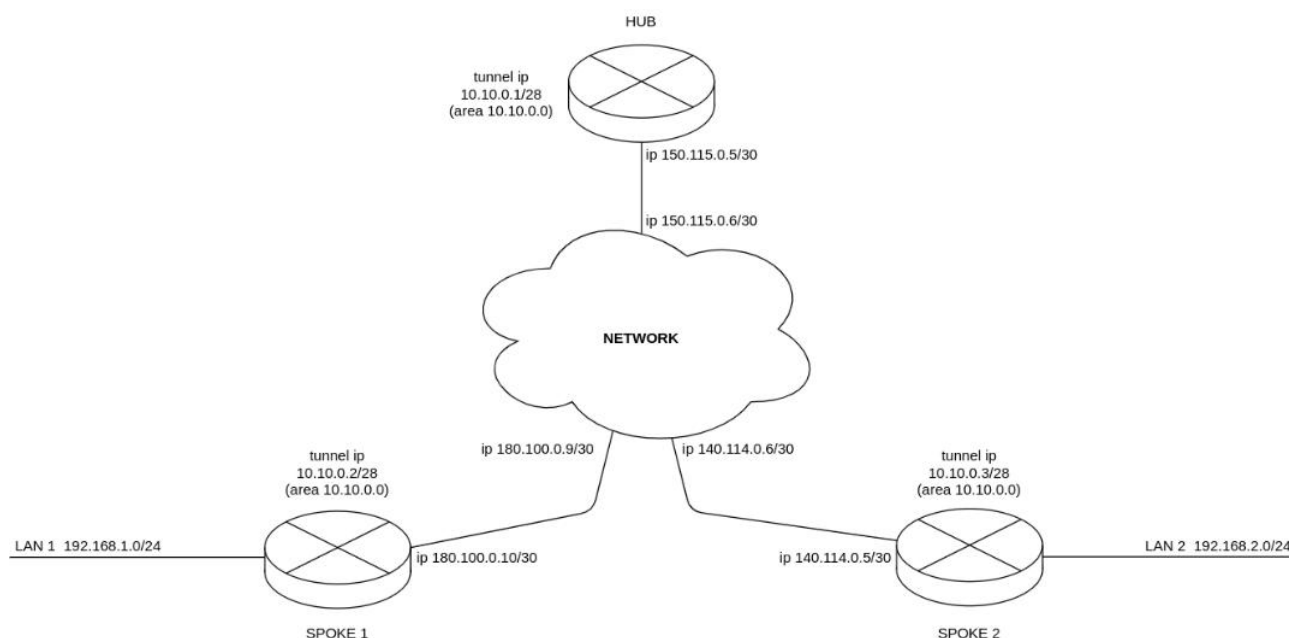
Очистить NHRP-записи можно командой:

```
rtt# clear ip nhrp peers
```

9.2.3. Пример настройки 2

Задача:

Организовать DMVPN между офисами компании с соответствующими подсетями LAN1 и LAN2, используя mGRE-туннели, NHRP (Next Hop Resolution Protocol), протокол динамической маршрутизации (OSPF), IPsec. В нашем примере у нас будет HUB-маршрутизатор и два филиала. HUB – это DMVPN-сервер (NHS), а филиалы – DMVPN-клиенты (NHS).



При использовании схемы DMVPN необходимо, чтобы HUB являлся DR-маршрутизатором. Таким образом, маршруты локальных подсетей spoke 1 и spoke 2 будут ретранслироваться через hub.

HUB внешний IP-адрес — 150.115.0.5;

SPOKE-1 внешний IP-адрес — 180.100.0.10;

SPOKE-2 внешний IP-адрес — 140.114.0.4.

Параметры IPsec VPN:

IKE:

- группа Диффи-Хэллмана: 2;
- алгоритм шифрования: AES128;
- алгоритм аутентификации: MD5.

IPsec:

- группа Диффи-Хэллмана: 2;
- алгоритм шифрования: AES128;
- алгоритм аутентификации: MD5.

Решение:

1. Конфигурирование HUB:

Предварительно настроим протокол OSPF:

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

Настроим интерфейс и определим принадлежность к зоне безопасности:

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

Настроим GRE-туннель, определим принадлежность к зоне безопасности, настроим OSPF на GRE-туннеле, настроим NHRP и включим туннель и NHRP командой enable. Чтобы HUB стал DR, необходимо выставить максимальный приоритет:

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

Создадим статические маршруты для подсетей интерфейсов spoke 180.100.0.8/30 и 140.114.0.4/30:

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

Произведём настройку IPsec для HUB:

```
rtt(config)# security ike proposal ike_prop1
rtt(config-ike-proposal)# authentication algorithm md5
rtt(config-ike-proposal)# encryption algorithm aes128
rtt(config-ike-proposal)# dh-group 2
rtt(config-ike-proposal)# exit
rtt(config)# security ike policy ike_poll
rtt(config-ike-policy)# pre-shared-key ascii-text password
rtt(config-ike-policy)# proposal ike_prop1
rtt(config-ike-policy)# exit
rtt(config)# security ike gateway ike_spoke
rtt(config-ike-gw)# ike-policy ike_poll
rtt(config-ike-gw)# local address 150.115.0.5
rtt(config-ike-gw)# local network 150.115.0.5/32 protocol gre
rtt(config-ike-gw)# remote address any
rtt(config-ike-gw)# remote network any
rtt(config-ike-gw)# mode policy-based
rtt(config-ike-gw)# exit
rtt(config)# security ipsec proposal ipsec_prop1
rtt(config-ipsec-proposal)# authentication algorithm md5
rtt(config-ipsec-proposal)# encryption algorithm aes128
rtt(config-ipsec-proposal)# pfs dh-group 2
rtt(config-ipsec-proposal)# exit
rtt(config)# security ipsec policy ipsec_poll
rtt(config-ipsec-policy)# proposal ipsec_prop1
rtt(config-ipsec-policy)# exit
rtt(config)# security ipsec vpn ipsec_spoke
rtt(config-ipsec-vpn)# mode ike
rtt(config-ipsec-vpn)# type transport
rtt(config-ipsec-vpn)# ike establish-tunnel route
rtt(config-ipsec-vpn)# ike gateway ike_spoke
rtt(config-ipsec-vpn)# ike ipsec-policy ipsec_poll
rtt(config-ipsec-vpn)# enable
rtt(config-ipsec-vpn)# exit
```

Привяжем IPsec к GRE-туннелю, чтобы клиенты могли устанавливать шифрованное соединение:

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

2. Конфигурирование SPOKE:

Предварительно настроим протокол OSPF с анонсированием подсети LAN1:

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

Настроим интерфейс и определим принадлежность к зоне безопасности:

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

Настроим GRE-туннель, определим принадлежность к зоне безопасности, настроим OSPF на GRE-туннеле, настроим NHRP и включим туннель и NHRP командой enable. Чтобы HUB стал DR, необходимо выставить минимальный приоритет на spoke:

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

Создадим статические маршруты для подсетей интерфейсов spoke 180.100.0.8/30 и 140.114.0.4/30:

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

Произведём настройку IPsec для SPOKE:

```
rtt(config)# security ike proposal ike_prop1
rtt(config-ike-proposal)# authentication algorithm md5
rtt(config-ike-proposal)# encryption algorithm aes128
rtt(config-ike-proposal)# dh-group 2
rtt(config-ike-proposal)# exit
rtt(config)# security ike policy ike_poll
rtt(config-ike-policy)# pre-shared-key ascii-text password
rtt(config-ike-policy)# proposal ike_prop1
rtt(config-ike-policy)# exit
rtt(config)# security ike gateway ike_spoke
rtt(config-ike-gw)# ike-policy ike_poll
rtt(config-ike-gw)# local address 180.100.0.10
rtt(config-ike-gw)# local network 180.100.0.10/32 protocol gre
rtt(config-ike-gw)# remote address any
rtt(config-ike-gw)# remote network any
rtt(config-ike-gw)# mode policy-based
```



```
rtt(config-ike-gw)# exit
rtt(config)# security ike gateway ike_hub
rtt(config-ike-gw)# ike-policy ike_poll
rtt(config-ike-gw)# local address 180.100.0.10
rtt(config-ike-gw)# local network 180.100.0.10/32 protocol gre
rtt(config-ike-gw)# remote address 150.115.0.5
rtt(config-ike-gw)# remote network 150.115.0.5/32 protocol gre
rtt(config-ike-gw)# mode policy-based
rtt(config-ike-gw)# exit
rtt(config)# security ipsec proposal ipsec_prop1
rtt(config-ipsec-proposal)# authentication algorithm md5
rtt(config-ipsec-proposal)# encryption algorithm aes128
rtt(config-ipsec-proposal)# pfs dh-group 2
rtt(config-ipsec-proposal)# exit
rtt(config)# security ipsec policy ipsec_poll
rtt(config-ipsec-policy)# proposal ipsec_prop1
rtt(config-ipsec-policy)# exit
rtt(config)# security ipsec vpn ipsec_spoke
rtt(config-ipsec-vpn)# mode ike
rtt(config-ipsec-vpn)# type transport
rtt(config-ipsec-vpn)# ike establish-tunnel route
rtt(config-ipsec-vpn)# ike gateway ike_spoke
rtt(config-ipsec-vpn)# ike ipsec-policy ipsec_poll
rtt(config-ipsec-vpn)# enable
rtt(config-ipsec-vpn)# exit
rtt(config)# security ipsec vpn ipsec_hub
rtt(config-ipsec-vpn)# mode ike
rtt(config-ipsec-vpn)# type transport
rtt(config-ipsec-vpn)# ike establish-tunnel route
rtt(config-ipsec-vpn)# ike gateway ike_hub
rtt(config-ipsec-vpn)# ike ipsec-policy ipsec_poll
rtt(config-ipsec-vpn)# enable
rtt(config-ipsec-vpn)# exit
```

Привяжем IPsec к GRE-туннелю, для возможности установления зашифрованного соединения с сервером и с другими клиентами сети:

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

Состояние NHRP-записей можно посмотреть командой:

```
rtt# show ip nhrp
```

Дополнительно в security zone-pair untrusted self необходимо разрешить протоколы для GRE over IPsec-туннеля, а также для протокола OSPF:

```
rtt(config)# object-group service ISAKMP_PORT
rtt(config-object-group-service)# port-range 500
rtt(config-object-group-service)# port-range 4500
rtt(config-object-group-service)# exit
rtt(config)# security zone-pair untrusted self
rtt(config-zone-pair)# rule 1
rtt(config-zone-pair-rule)# action permit
```

```

rtt(config-zone-pair-rule)# match protocol udp
rtt(config-zone-pair-rule)# match destination-port object-group ISAKMP_PORT
rtt(config-zone-pair-rule)# enable
rtt(config-zone-pair-rule)# exit
rtt(config-zone-pair)# rule 2
rtt(config-zone-pair-rule)# action permit
rtt(config-zone-pair-rule)# match protocol gre
rtt(config-zone-pair-rule)# enable
rtt(config-zone-pair-rule)# exit
rtt(config-zone-pair)# rule 3
rtt(config-zone-pair-rule)# action permit
rtt(config-zone-pair-rule)# match protocol esp
rtt(config-zone-pair-rule)# enable
rtt(config-zone-pair-rule)# exit
rtt(config-zone-pair)# rule 4
rtt(config-zone-pair-rule)# action permit
rtt(config-zone-pair-rule)# match protocol ospf
rtt(config-zone-pair-rule)# enable
rtt(config-zone-pair-rule)# exit
rtt(config-zone-pair)# exit

```

9.3. Настройка L2TPv3-туннелей

L2TPv3 (Layer 2 Tunneling Protocol Version 3) — протокол для туннелирования пакетов 2 уровня модели OSI между двумя IP-узлами. В качестве инкапсулирующего протокола используется IP или UDP. L2TPv3 может использоваться как альтернатива MPLS P2P L2VPN (VLL) для организации VPN уровня L2. В маршрутизаторе RTT реализованы статические неуправляемые L2TPv3-туннели, то есть туннели создаются вручную путем конфигурирования на локальном и удаленном узлах. Параметры туннеля на каждой из сторон должны быть взаимосогласованными или переносимые данные не будут деинкапсулироваться партнером.

9.3.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Сконфигурировать L3-интерфейс, от которого будет строиться L2TPv3-туннель.		
2	Создать L2TPv3-туннель и перейти в режим его конфигурирования.	<code>rtt(config)# tunnel l2tpv3 <INDEX></code>	<p><INDEX> – идентификатор туннеля в диапазоне:</p> <ul style="list-style-type: none"> • для R100/200 – [1..250]; • для R800 – [1..500].
3	Указать описание конфигурируемого туннеля (необязательно).	<code>rtt(config-l2tpv3)# description <DESCRIPTION></code>	<DESCRIPTION> – описание туннеля, задаётся строкой до 255 символов.
4	Установить локальный IP-адрес для установки туннеля.	<code>rtt(config-l2tpv3)# local address <ADDR></code>	<ADDR> – IP-адрес локального шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Шаг	Описание	Команда	Ключи
5	Установить удаленный IP-адрес для установки туннеля.	<code>rtt(config-l2tpv3) # remote address <ADDR></code>	<ADDR> – IP-адрес локального шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
6	Выбрать метод инкапсуляции для туннеля L2TPv3.	<code>rtt(config-l2tpv3) # protocol <TYPE></code>	<TYPE> – тип инкапсуляции, возможные значения: <ul style="list-style-type: none"> • ip – инкапсуляция в IP-пакет; • udp – инкапсуляция в UDP-дейтаграммы.
7	Установить локальный идентификатор сессии.	<code>rtt(config-l2tpv3) # local session-id <SESSION-ID></code>	<SESSION-ID> – идентификатор сессии, принимает значения [1..200000].
8	Установить удаленный идентификатор сессии.	<code>rtt(config-l2tpv3) # remote session-id <SESSION-ID></code>	<SESSION-ID> – идентификатор сессии, принимает значения [1..200000].
9	Определить локальный UDP-порт (если в качестве метода инкапсуляции был выбран UDP протокол).	<code>rtt(config-l2tpv3) # local port <UDP></code>	<UDP> – номер UDP-порта в диапазоне [1..65535].
10	Определить удаленный UDP-порт (если в качестве метода инкапсуляции был выбран UDP-протокол).	<code>rtt(config-l2tpv3) # remote port <UDP></code>	<UDP> – номер UDP-порта в диапазоне [1..65535].
11	Назначить широковещательный домен для инкапсуляции в L2TPV3-пакеты данного туннеля.	<code>rtt(config-l2tpv3) # bridge-group <BRIDGE-ID></code>	<BRIDGE-ID> – идентификационный номер моста, принимает значения в диапазоне: <ul style="list-style-type: none"> • для R100/200 – [1..250]; • для R800 – [1..500].
12	Активировать туннель.	<code>rtt(config-l2tpv3) # enable</code>	
13	Указать размер MTU (MaximumTransmissionUnit) для туннелей (необязательно). MTU более 1500 будет активно только в случае применения команды "system jumbo-frames".	<code>rtt(config-l2tpv3) # mtu <MTU></code>	<MTU> – значение MTU, принимает значения в диапазоне: [1280..10000]. Значение по умолчанию: 1500.
14	Определить локальное значение cookie для дополнительной проверки соответствия между передаваемыми данными и сессией (необязательно).	<code>rtt(config-l2tpv3) # local cookie <COOKIE></code>	<COOKIE> – значение COOKIE, параметр принимает значения длиной восемь или шестнадцать символов в шестнадцатеричном виде.

Шаг	Описание	Команда	Ключи
15	Определить удаленное значение cookie для дополнительной проверки соответствия между передаваемыми данными и сессией (необязательно).	<code>rtt(config-l2tpv3) # remote cookie <COOKIE></code>	<COOKIE> – значение COOKIE, параметр принимает значения длиной восемь или шестнадцать символов в шестнадцатеричном виде.
16	Задать интервал времени, за который усредняется статистика о нагрузке на туннеле (не обязательно).	<code>rtt(config-l2tpv3) # load-average <TIME></code>	<TIME> – интервал в секундах, принимает значения [5..150]. Значение по умолчанию: 5.
17	Включить запись статистики использования текущего туннеля (необязательно).	<code>rtt(config-if-sub) # history statistics</code>	

Также для L2TPv3-туннеля возможно настроить:

- QoS в базовом или расширенном режимах (см. раздел **Управление QoS**);
- функционал BRAS (см. раздел [Управление BRAS \(Broadband Remote Access Server\)](#)).

9.3.2. Пример настройки L2TPv3-туннеля

Задача:

Организовать L2 VPN между офисами компании через IP-сеть, используя для туннелирования трафика протокол L2TPv3.

- в качестве инкапсулирующего протокола используется UDP, номер порта на локальной стороне и номер порта на стороне партнера 519;
- в качестве локального шлюза для туннеля используется IP-адрес 21.0.0.1;
- в качестве удаленного шлюза для туннеля используется IP-адрес 183.0.0.10;
- идентификатор туннеля на локальной стороне равен 2, на стороне партнера 3;
- идентификатор сессии внутри туннеля равен 100, на стороне партнера 200;
- в туннель направим трафик из bridge с идентификатором 333.



Решение:



Предварительно необходимо в firewall разрешить входящий трафик по протоколу UDP с портом отправителя 519 и портом назначения 519.

Создадим туннель L2TPv3 333:

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

Укажем локальный и удаленный шлюз (IP-адреса интерфейсов, граничащих с WAN):

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

Укажем тип инкапсулирующего протокола и номера UDP-портов:

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

Укажем идентификаторы сессии внутри туннеля для локальной и удаленной сторон:

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

Установим принадлежность L2TPv3-туннеля к мосту, который должен быть связан с сетью удаленного офиса (настройка моста рассматривается в пункте **Пример настройки bridge для VLAN и L2TPv3-туннеля**):

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

Включим ранее созданный туннель и выйдем:

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

Создадим суб-интерфейс для коммутации трафика, поступающего из туннеля, в локальную сеть с тегом VLAN id 333:

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

Установим принадлежность суб-интерфейса к мосту, который должен быть связан с локальной сетью (настройка моста рассматривается в пункте **Настройка PPP через E1**):

```
rtt(config-if-sub)# bridge-group 333
rtt(config-if-sub)# exit
```

После применения настроек трафик будет инкапсулироваться в туннель и отправляться партнеру, независимо от наличия L2TPv3-туннеля и правильности настроек с его стороны.

Настройки туннеля в удаленном офисе должны быть зеркальными локальным. В качестве локального шлюза должен использоваться IP-адрес 183.0.0.10. В качестве удаленного шлюза должен использоваться IP-адрес 21.0.0.1. Номер порта инкапсулирующего протокола на локальной стороне и стороне партнера 519. Идентификатор сессии внутри туннеля должен быть равным 200, на стороне партнера 100. Также туннель должен принадлежать мосту, который необходимо соединить с сетью партнера.

Состояние туннеля можно посмотреть командой:

```
rtt# show tunnels status l2tpv3 333
```

Счетчики входящих и отправленных пакетов можно посмотреть командой:

```
rtt# show tunnels counters l2tpv3 333
```

Конфигурацию туннеля можно посмотреть командой:

```
rtt# show tunnels configuration l2tpv3 333
```

9.4. Настройка IPsec VPN

IPsec — это набор протоколов, которые обеспечивают защиту передаваемых с помощью IP-протокола данных. Данный набор протоколов позволяет осуществлять подтверждение подлинности (аутентификацию), проверку целостности и шифрование IP-пакетов, а также включает в себя протоколы для защищённого обмена ключами в сети Интернет.

9.4.1. Алгоритм настройки Route-based IPsec VPN

Шаг	Описание	Команда	Ключи
1	Создать VTI-туннель и перейти в режим его конфигурирования.	<code>rtt(config)# tunnel vti <TUN></code>	<TUN> – имя туннеля устройства.
2	Указать локальный IP-адрес VTI-туннеля.	<code>rtt(config-vti)#local address <ADDR></code>	<ADDR> – IP-адрес локального шлюза.
3	Указать удаленный IP-адрес VTI-туннеля.	<code>rtt(config-vti)#remote address <ADDR></code>	<ADDR> – IP-адрес удаленного шлюза.

Шаг	Описание	Команда	Ключи
4	Установить IP-адрес локальной стороны VTI-туннеля.	<pre>rtt(config-vti)# ip address <ADDR/LEN> [unit <ID>]</pre> <p>или</p> <pre>rtt(config-vti)# ip address <ADDR/LEN> secondary [unit <ID>]</pre>	<p><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];</p> <p><ID> – номер юнита, принимает значения [1..4].</p> <p>Ключ secondary указывает, что настроенный адрес является дополнительным IP-адресом. Если это ключевое слово отсутствует, настроенный адрес является основным IP-адресом. Возможно указать до 7 дополнительных IP-адресов.</p>
5	Включить VTI-туннель в зону безопасности и настроить правила взаимодействия между зонами или отключить firewall для VTI-туннеля.	<pre>rtt(config-vti)# security-zone<NAME></pre> <pre>rtt(config-vti)# ip firewall disable</pre>	<p><NAME> – имя зоны безопасности, задаётся строкой до 12 символов.</p>
6	Включить туннель.	<pre>rtt(config-vti)#enable</pre>	
7	Создать IKE-профиль и перейти в режим его конфигурирования.	<pre>rtt(config)# security ike proposal <NAME></pre>	<NAME> – имя профиля протокола IKE, задаётся строкой до 31 символа.
8	Указать описание конфигурируемого IKE-профиля (необязательно).	<pre>rtt(config-ike-proposal)# description<DESCRIPTION></pre>	<DESCRIPTION> – описание туннеля, задаётся строкой до 255 символов.
9	Определить алгоритм аутентификации для IKE (необязательно).	<pre>rtt(config-ike-proposal)# authentication algorithm <ALGORITHM></pre>	<p><ALGORITHM> – алгоритм аутентификации, принимает значения: md5, sha1, sha2-256, sha2-384, sha2-512.</p> <p>Значение по умолчанию: sha1.</p>
10	Определить алгоритм шифрования для IKE (необязательно).	<pre>rtt(config-ike-proposal)# encryption algorithm <ALGORITHM></pre>	<p><ALGORITHM> – протокол шифрования, принимает значения: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.</p> <p>Значение по умолчанию: 3des.</p>

Шаг	Описание	Команда	Ключи
11	Определить номер группы Диффи-Хэллмана (необязательно).	<code>rtt(config-ike-proposal)# dh-group <DH-GROUP></code>	<DH-GROUP> – номер группы Диффи-Хэллмана, принимает значения [1, 2, 5, 14, 15, 16, 17, 18]. Значение по умолчанию: 1.
12	Создать IKE-политику и перейти в режим её конфигурирования.	<code>rtt(config)# security ike policy <NAME></code>	<NAME> – имя политики IKE, задаётся строкой до 31 символа.

Шаг	Описание	Команда	Ключи
13	Определить режим аутентификации IKE (необязательно).	<code>rtt(config-ike-policy) # authentication method <METHOD></code>	<p><METHOD> – метод аутентификации IKE-сессии. Может принимать значения:</p> <ul style="list-style-type: none"> • pre-shared-key – метод аутентификации, использующий предварительно согласованные ключи, которые должны совпадать у обоих участников IKE-сессии; • keyring – метод аутентификации, использующий набор предварительно согласованных ключей; • public-key – метод аутентификации, использующий приватные ключи и сертификаты X.509. Файлы сертификатов и ключей должны быть загружены в локальное хранилище маршрутизатора; • trustpoint – метод аутентификации, использующий приватные ключи и сертификаты X.509. Файлы сертификатов и ключей предоставляются PKI-клиентом, который автоматически выписывает актуальные сертификаты у удостоверяющего центра; • xauth-psk-key – метод расширенной аутентификации, использующий в качестве первого фактора аутентификации предварительно согласованные ключи и пару логин-пароль пользователя в качестве второго фактора аутентификации; • eap – метод расширенной аутентификации, использующий приватные ключи и сертификаты X.509 для аутентификации ответчика в IKE-сессии и пару логин-пароль пользователя для аутентификации инициатора IKE-сессии.

Шаг	Описание	Команда	Ключи
14	Задать время жизни соединения протокола IKE (необязательно).	<code>rtt(config-ike-policy) # lifetime seconds <SEC></code>	<p><SEC> – период времени, принимает значения [4 ..86400] секунд.</p> <p>Значение по умолчанию: 10800.</p>
15	Привязать IKE-профиль к IKE-политике.	<code>rtt(config-ike-policy) # proposal <NAME></code>	<NAME> – имя профиля протокола IKE, задаётся строкой до 31 символа.
16	Указать ключ аутентификации (обязательно, если в качестве режима аутентификации выбран pre-shared-key).	<code>rtt(config-ike-policy) # pre-shared-key ascii-text<TEXT></code>	<TEXT> – строка [1..64] ASCII-символов.
17	Создать IKE-шлюз и перейти в режим его конфигурирования.	<code>rtt(config) # security ike gateway <NAME></code>	<NAME> – имя шлюза протокола IKE, задаётся строкой до 31 символа.
18	Привязать IKE-политику к IKE-шлюзу.	<code>rtt(config-ike-gw) # ike-policy <NAME></code>	<NAME> – имя политики протокола IKE, задаётся строкой до 31 символа.
19	Указать версию IKE (необязательно).	<code>rtt(config-ike-gw) # version <VERSION></code>	<p><version> – версия IKE-протокола: v1-only или v2-only.</p> <p>Значение по умолчанию: v1-only.</p>
20	Установить режим перенаправления трафика в туннель – route-based.	<code>rtt(config-ike-gw) # mode route-based</code>	
21	Указать действие для DPD (необязательно).	<code>rtt(config-ike-gw) # dead-peer-detection action <MODE></code>	<p><MODE> – режим работы DPD:</p> <ul style="list-style-type: none"> • restart – соединение переустанавливается; • clear – соединение останавливается; • hold – соединение поддерживается; • none – механизм выключен, никаких действий не предпринимается. <p>Значение по умолчанию: none.</p>
22	Указать интервал между отправкой сообщений механизмом DPD (необязательно).	<code>rtt(config-ike-gw) # dead-peer-detection interval <SEC></code>	<p><SEC> – интервал между отправкой сообщений механизмом DPD, принимает значения [1..180] секунд.</p> <p>Значение по умолчанию: 2.</p>

Шаг	Описание	Команда	Ключи
23	Указать период времени ожидания ответа на сообщения механизма DPD (необязательно).	<code>rtt(config-ike-gw)# dead-peer-detection timeout <SEC></code>	<p><SEC> – период времени ожидания ответа на сообщения механизма DPD принимает значения [1..180] секунд.</p> <p>Значение по умолчанию: 30 секунд.</p>
24	Указать базовый период времени ожидания ответа на сообщения (необязательно).	<code>rtt(config-ike-gw)# retransmit timeout <SEC></code>	<p><SEC> – базовый период времени ожидания ответа на сообщения принимает значения [1..30] секунд.</p> <p>Значение по умолчанию: 4 секунды.</p>
25	Указать количество попыток повторной отправки сообщений после наступления таймаута ожидания ответа (необязательно).	<code>rtt(config-ike-gw)# retransmit tries <TRIES></code>	<p><TRIES> – количество попыток повторной отправки сообщений механизма DPD в случае наступления таймаута ожидания ответа принимает значения от 1 до 10. Для первого отправленного сообщения период времени ожидания ответа будет равен базовому периоду, указанному в команде <code>retransmit timeout</code>, а для последующих попыток интервал ожидания будет рассчитан по формуле:</p> <p>"retransmit timeout" * 1.8 ^ (N-1), где N - номер попытки.</p> <p>Значение по умолчанию: 5 попыток.</p>
26	Указать уровень случайного разброса периода ожидания ответа на сообщения (необязательно).	<code>rtt(config-ike-gw)# retransmit jitter <VALUE></code>	<p><VALUE> – максимальный процент разброса значений, принимает значения [0..100].</p> <p>Значение по умолчанию: 0 %</p>
27	Указать ограничение максимального периода времени ожидания ответа на сообщения (необязательно).	<code>rtt(config-ike-gw)# retransmit limit <SEC></code>	<p><SEC> – максимальный период времени ожидания ответа на сообщения принимает значения [15..300] секунд.</p> <p>Значение по умолчанию: 0 секунд, у периода нет верхнего предела.</p>
28	Данная команда отключает расширение MOBIKE IKEv2, которое позволяет инициатору ike-сессии изменять local address в соответствии с RFC 4555.	<code>rtt(config-ike-gw)# mobike disable</code>	
29	Привязать VTI-туннель к IKE-шлюзу.	<code>rtt(config-ike-gw)# bind-interface vti <VTI></code>	<VTI> – идентификационный номер интерфейса VTI.
30	Создать в IPsec-профиль.	<code>rtt(config)# security ipsec proposal <NAME></code>	<NAME> – имя профиля протокола IPsec, задаётся строкой до 31 символа.

Шаг	Описание	Команда	Ключи
31	Определить алгоритм аутентификации для IPsec (необязательно).	<code>rtt(config-ipsec-proposal)# authentication algorithm <ALGORITHM></code>	<p><ALGORITHM> – алгоритм аутентификации, принимает значения: md5, sha1, sha2-256, sha2-384, sha2-512.</p> <p>Значение по умолчанию: sha1.</p>
32	Определить алгоритм шифрования для IPsec (необязательно).	<code>rtt(config-ipsec-proposal)# encryption algorithm <ALGORITHM></code>	<p><ALGORITHM> – протокол шифрования, принимает значения: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.</p> <p>Значение по умолчанию: 3des.</p>
33	Указать протокол инкапсуляции для IPsec (необязательно).	<code>rtt(config-ipsec-proposal)# protocol <PROTOCOL></code>	<p><PROTOCOL> – инкапсулирующий протокол, принимает значения:</p> <ul style="list-style-type: none"> • ah – данный протокол осуществляет только аутентификацию трафика, шифрование данных не выполняется; • esp – данный протокол осуществляет аутентификацию и шифрование трафика. <p>Значение по умолчанию: esp.</p>
34	Создать IPsec-политику и перейти в режим её конфигурирования.	<code>rtt(config)# security ipsec policy <NAME></code>	<NAME> – имя политики IPsec, задаётся строкой до 31 символа.
35	Привязать IPsec-профиль к IPsec-политике.	<code>rtt(config-ipsec-policy)# proposal <NAME></code>	<NAME> – имя профиля протокола IPsec, задаётся строкой до 31 символа.
36	Задать время жизни IPsec-туннеля (необязательно).	<code>rtt(config-ipsec-policy)# lifetime { seconds <SEC> packets <PACKETS> kilobytes <KB> }</code>	<p><SEC> – период времени жизни IPsec-туннеля, по истечении происходит пересогласование. Принимает значения [1140..86400] секунд.</p> <p><PACKETS> – количество пакетов, после передачи которого происходит пересогласование IPsec-туннеля. Принимает значения [4..86400].</p> <p><KB> – объем трафика, после передачи которого происходит пересогласование IPsec-туннеля. Принимает значения [4..4608000] секунд.</p> <p>Значение по умолчанию: 28800 секунд.</p>

Шаг	Описание	Команда	Ключи
37	Отключить реаутентификацию IKE-сессии (необязательно).	<code>rtt(config-ipsec-policy)# reauthentication disable</code>	
38	Создать IPsec VPN и перейти в режим конфигурирования.	<code>rtt(config)# security ipsec vpn <NAME></code>	<NAME> – имя VPN, задаётся строкой до 31 символа.
39	Определить режим согласования данных, необходимых для активации VPN.	<code>rtt(config-ipsec-vpn)# mode <MODE></code>	<MODE> – режим работы VPN.
40	Привязать IPsec-политику к IPsec-VPN.	<code>rtt(config-ipsec-vpn)# ike ipsec-policy <NAME></code>	<NAME> – имя IPsec-политики, задаётся строка до 31 символа.
41	Задать значение DSCP для использования в IP-заголовке исходящих пакетов IKE-протокола (необязательно).	<code>rtt(config-ipsec-vpn)# ike dscp <DSCP></code>	DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63]. Значение по умолчанию: 63.
42	Установить режим активации VPN.	<code>rtt(config-ipsec-vpn)# ike establish-tunnel <MODE></code>	<MODE> – режим активации VPN: <ul style="list-style-type: none"> • by - request – соединение активируется встречной стороной; • route – соединение активируется при появлении трафика, маршрутизируемого в туннель; • immediate – туннель активируется автоматически после применения конфигурации.
43	Осуществить привязку IKE-шлюза к IPsec-VPN.	<code>rtt(config-ipsec-vpn)# ike gateway <NAME></code>	<NAME> – имя IKE-шлюза, задаётся строкой до 31 символа.
44	Установить значение временного интервала в секундах, по истечению которого соединение закрывается, если не было принято или передано ни одного пакета через SA (необязательно).	<code>rtt(config-ipsec-vpn)# ike idle-time <TIME></code>	<TIME> – интервал в секундах, принимает значения [4..86400].

Шаг	Описание	Команда	Ключи
45	Отключить пересогласование ключей до разрыва IKE-соединения по истечению времени, количеству переданных пакетов или байт (необязательно).	<code>rtt(config-ipsec-vpn) # ike rekey disable</code>	
46	Настроить начало пересогласования ключей IKE-соединения до истечения времени жизни (необязательно).	<code>rtt(config-ipsec-vpn) # ike rekey margin { seconds <SEC> packets <PACKETS> kilobytes <KB> }</code>	<p><SEC> – интервал времени в секундах, оставшийся до закрытия соединения (задается командой <code>lifetimeseconds</code>, см. 22.2.13). Принимает значения [4..86400].</p> <p><PACKETS> – количество пакетов, оставшихся до закрытия соединения (задается командой <code>lifetimepackets</code>). Принимает значения [4..86400]</p> <p><KB> – объем трафика в килобайтах, оставшийся до закрытия соединения (задается командой <code>lifetimekilobytes</code>). Принимает значения [4..86400]</p> <p>Значение по умолчанию:</p> <ul style="list-style-type: none"> • Пересогласование ключей до истечения времени – за 540 секунд. • Пересогласование ключей до истечения объема трафика и количества пакетов – отключено.
47	Установить уровень случайного разброса значений параметров <code>margin seconds</code> , <code>margin packets</code> , <code>margin kilobytes</code> (необязательно).	<code>rtt(config-ipsec-vpn) # ike rekey randomization <VALUE></code>	<p><VALUE> – максимальный процент разброса значений, принимает значения [1..100].</p> <p>Значение по умолчанию: 100%</p>
48	Указать описание для IPsec-VPN (необязательно).	<code>rtt(config-ipsec-vpn) # description <DESCRIPTION></code>	<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.
49	Активировать IPsec VPN.	<code>rtt(config-ipsec-vpn) # enable</code>	

9.4.2. Пример настройки Route-based IPsec VPN



Задача:

Настроить IPsec-туннель между R1 и R2.

- R1 IP-адрес – 120.11.5.1;
- R2 IP-адрес – 180.100.0.1.

IKE:

- группа Диффи-Хэллмана: 2;
- алгоритм шифрования: AES 128 bit;
- алгоритм аутентификации: MD5.

IP sec:

- алгоритм шифрования: AES 128 bit;
- алгоритм аутентификации: MD5.

Решение:



Предварительно в firewall необходимо разрешить протокол ESP и ISAKMP (UDP-порт 500, 4500).

1. Конфигурирование R1

Настроим внешний сетевой интерфейс и определим принадлежность к зоне безопасности:

```
rtt# configure
rtt(config)# interface gi 1/0/1
rtt(config-if-gi)# ip address 180.100.0.1/24
rtt(config-if-gi)# security-zone untrusted
rtt(config-if-gi)# exit
```

Создадим туннель VTI. Трафик будет перенаправляться через VTI в IPsec-туннель. В качестве локального и удаленного шлюза указываются IP-адреса интерфейсов, граничащих с WAN:

```
rtt(config)# tunnel vti 1
rtt(config-vti)# local address 180.100.0.1
rtt(config-vti)# remote address 120.11.5.1
rtt(config-vti)# ip address 10.10.10.1/30
```

```
rtt(config-vti)# enable
rtt(config-vti)# exit
```

Для настройки правил зон безопасности потребуется создать профиль порта протокола ISAKMP:

```
rtt(config)# object-group service ISAKMP
rtt(config-object-group-service)# port-range 500,4500
rtt(config-object-group-service)# exit
```

Создадим статический маршрут до удаленной LAN-сети. Для каждой подсети, которая находится за IPsec-туннелем, нужно указать маршрут через VTI-туннель:

```
rtt(config)# ip route 192.0.2.0/24 tunnel vti 1
```

Создадим профиль протокола IKE. В профиле укажем группу Диффи-Хэллмана 2, алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IKE-соединения:

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

Создадим политику протокола IKE. В политике указывается список профилей протокола IKE, по которым могут согласовываться узлы и ключ аутентификации:

```
rtt(config)# security ike policy ike_poll
rtt(config-ike-policy)# pre-shared-key hexadecimal 123FFF
rtt(config-ike-policy)# proposal ike_prop1
rtt(config-ike-policy)# exit
```

Создадим шлюз протокола IKE. В данном профиле указывается VTI-туннель, политика, версия протокола и режим перенаправления трафика в туннель. Поддержка MOBIKE отключается для route-based IPsec в обязательном порядке:

```
rtt(config)# security ike gateway ike_gw1
rtt(config-ike-gw)# ike-policy ike_poll
rtt(config-ike-gw)# mode route-based
rtt(config-ike-gw)# mobike disable
rtt(config-ike-gw)# bind-interface vti 1
rtt(config-ike-gw)# version v2-only
rtt(config-ike-gw)# exit
```

Создадим профиль параметров безопасности для IPsec-туннеля. В профиле укажем алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IPsec-туннеля:

```
rtt(config)# security ipsec proposal ipsec_prop1
rtt(config-ipsec-proposal)# authentication algorithm md5
rtt(config-ipsec-proposal)# encryption algorithm aes128
rtt(config-ipsec-proposal)# exit
```


Создадим политику для IPsec-туннеля. В политике указывается список профилей IPsec-туннеля, по которым могут согласовываться узлы:

```
rtt(config)# security ipsec policy ipsec_poll
rtt(config-ipsec-policy)# proposal ipsec_prop1
rtt(config-ipsec-policy)# exit
```

Создадим IPsec VPN. В VPN указывается шлюз IKE-протокола, политика IP sec-туннеля, режим обмена ключами и способ установления соединения. После ввода всех параметров включим туннель командой **enable**:

```
rtt(config)# security ipsec vpn ipsec1
rtt(config-ipsec-vpn)# mode ike
rtt(config-ipsec-vpn)# ike establish-tunnel route
rtt(config-ipsec-vpn)# ike gateway ike_gw1
rtt(config-ipsec-vpn)# ike ipsec-policy ipsec_poll
rtt(config-ipsec-vpn)# enable
rtt(config-ipsec-vpn)# exit
rtt(config)# exit
```

2. Конфигурирование R2

Настроим внешний сетевой интерфейс и определим принадлежность к зоне безопасности:

```
rtt# configure
rtt(config)# interface gi 1/0/1
rtt(config-if)# ip address 120.11.5.1/24
rtt(config-if)# security-zone untrusted
rtt(config-if)# exit
```

Создадим туннель VTI. Трафик будет перенаправляться через VTI в IPsec-туннель. В качестве локального и удаленного шлюза указываются IP-адреса интерфейсов, граничащих с WAN:

```
rtt(config)# tunnel vti 1
rtt(config-vti)# remote address 180.100.0.1
rtt(config-vti)# local address 120.11.5.1
rtt(config-vti)# ip address 10.10.10.2/30
rtt(config-vti)# enable
rtt(config-vti)# exit
```

Для настройки правил зон безопасности потребуется создать профиль порта протокола ISAKMP:

```
rtt(config)# object-group service ISAKMP
rtt(config-object-group-service)# port-range 500,4500
rtt(config-object-group-service)# exit
```

Создадим статический маршрут до удаленной LAN-сети. Для каждой подсети, которая находится за IPsec-туннелем, нужно указать маршрут через VTI-туннель:

```
rtt(config)# ip route 10.0.0.0/16 tunnel vti 1
```

Создадим профиль протокола IKE. В профиле укажем группу Диффи-Хэллмана 2, алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IKE-соединения:

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

Создадим политику протокола IKE. В политике указывается список профилей протокола IKE, по которым могут согласовываться узлы и ключ аутентификации:

```
rtt(config)# security ike policy ike_poll
rtt(config-ike-policy)# pre-shared-key hexadecimal 123FFF
rtt(config-ike-policy)# proposal ike_prop1
rtt(config-ike-policy)# exit
```

Создадим шлюз протокола IKE. В данном профиле указывается VTI-туннель, политика, версия протокола и режим перенаправления трафика в туннель:

```
rtt(config)# security ike gateway ike_gw1
rtt(config-ike-gw)# ike-policy ike_poll
rtt(config-ike-gw)# mode route-based
rtt(config-ike-gw)# bind-interface vti 1
rtt(config-ike-gw)# version v2-only
rtt(config-ike-gw)# exit
```

Создадим профиль параметров безопасности для IPsec-туннеля. В профиле укажем алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IPsec-туннеля:

```
rtt(config)# security ipsec proposal ipsec_prop1
rtt(config-ipsec-proposal)# authentication algorithm md5
rtt(config-ipsec-proposal)# encryption algorithm aes128
rtt(config-ipsec-proposal)# exit
```

Создадим политику для IPsec-туннеля. В политике указывается список профилей IPsec-туннеля, по которым могут согласовываться узлы.

```
rtt(config)# security ipsec policy ipsec_poll
rtt(config-ipsec-policy)# proposal ipsec_prop1
rtt(config-ipsec-policy)# exit
```

Создадим IPsec VPN. В VPN указывается шлюз IKE-протокола, политика IP sec-туннеля, режим обмена ключами и способ установления соединения. После ввода всех параметров включим туннель командой **enable**:

```
rtt(config)# security ipsec vpn ipsec1
rtt(config-ipsec-vpn)# mode ike
rtt(config-ipsec-vpn)# ike establish-tunnel route
rtt(config-ipsec-vpn)# ike gateway ike_gw1
```

```

rtt(config-ipsec-vpn)# ike ipsec-policy ipsec_pol1
rtt(config-ipsec-vpn)# enable
rtt(config-ipsec-vpn)# exit
rtt(config)# exit

```

Состояние туннеля можно посмотреть командой:

```

rtt# show security ipsec vpn status ipsec1

```

Конфигурацию туннеля можно посмотреть командой:

```

rtt# show security ipsec vpn configuration ipsec1

```

9.4.3. Алгоритм настройки Policy-based IPsec VPN

Шаг	Описание	Команда	Ключи
1	Создать IKE-экземпляр и перейти в режим его конфигурирования.	<code>rtt(config)# security ike proposal <NAME></code>	<NAME> – имя профиля протокола IKE, задаётся строкой до 31 символа.
2	Указать описание конфигурируемого туннеля (необязательно).	<code>rtt(config-ike-proposal)# description<DESCRIPTION></code>	<DESCRIPTION> – описание туннеля, задаётся строкой до 255 символов.
3	Определить алгоритм аутентификации для IKE.	<code>rtt(config-ike-proposal)# authentication algorithm <ALGORITHM></code>	<ALGORITHM> – алгоритм аутентификации, принимает значения: md5, sha1, sha2-256, sha2-384, sha2-512.
4	Определить алгоритм шифрования для IKE.	<code>rtt(config-ike-proposal)# encryption algorithm <ALGORITHM></code>	<ALGORITHM> – протокол шифрования, принимает значения: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.
5	Определить номер группы Диффи-Хэллмана.	<code>rtt(config-ike-proposal)# dh-group <DH-GROUP></code>	<DH-GROUP> – номер группы Диффи-Хэллмана, принимает значения [1, 2, 5, 14, 15, 16, 17, 18].
6	Создать политику для профиля IKE и перейти в режим её конфигурирования.	<code>rtt(config)# security ike policy <NAME></code>	<NAME> – имя политики IKE, задаётся строкой до 31 символа.

Шаг	Описание	Команда	Ключи
7	Определить режим аутентификации.	<code>rtt(config-ike-policy) # authentication method <METHOD></code>	<p><METHOD> – метод аутентификации IKE-сессии. Может принимать значения:</p> <ul style="list-style-type: none"> • pre-shared-key – метод аутентификации, использующий предварительно согласованные ключи, которые должны совпадать у обоих участников IKE-сессии; • keyring – метод аутентификации, использующий набор предварительно согласованных ключей; • public-key – метод аутентификации, использующий приватные ключи и сертификаты X.509. Файлы сертификатов и ключей должны быть загружены в локальное хранилище маршрутизатора; • trustpoint – метод аутентификации, использующий приватные ключи и сертификаты X.509. Файлы сертификатов и ключей предоставляются PKI-клиентом, который автоматически выписывает актуальные сертификаты у удостоверяющего центра; • xauth-psk-key – метод расширенной аутентификации, использующий в качестве первого фактора аутентификации предварительно согласованные ключи и пару логин-пароль пользователя в качестве второго фактора аутентификации; • eap – метод расширенной аутентификации, использующий приватные ключи и сертификаты X.509 для аутентификации ответчика в IKE-сессии и пару логин-пароль пользователя для аутентификации инициатора IKE-сессии.

Шаг	Описание	Команда	Ключи
8	Задать время жизни соединения протокола IKE (необязательно).	<code>rtt(config-ike-policy) # lifetime seconds <SEC></code>	<SEC> – период времени, принимает значения [4 ..86400] секунд. Значение по умолчанию: 10800.
9	Привязать политику к профилю.	<code>rtt(config-ike-policy) # proposal <NAME></code>	<NAME> – имя профиля протокола IKE, задаётся строкой до 31 символа.
10	Указать ключ аутентификации.	<code>rtt(config-ike-policy) # pre-shared-key ascii-text<TEXT></code>	<TEXT> – строка [1..64] ASCII-символов.
11	Создать шлюз для IKE и перейти в режим его конфигурирования.	<code>rtt(config) # security ike gateway <NAME></code>	<NAME> – имя шлюза протокола IKE, задаётся строкой до 31 символа.
12	Привязать политику IKE.	<code>rtt(config-ike-gw) # ike-policy <NAME></code>	<NAME> – имя политики протокола IKE, задаётся строкой до 31 символа.
13	Указать версию IKE (необязательно).	<code>rtt(config-ike-gw) # version <VERSION></code>	<version> – версия IKE-протокола: v1-only или v2-only .
14	Установить режим перенаправления трафика в туннель.	<code>rtt(config-ike-gw) # mode<MODE></code>	<MODE> – режим перенаправления трафика в туннель, принимает значения: <ul style="list-style-type: none"> • policy - based – трафик перенаправляется на основе принадлежности к указанным в политиках подсетям; • route - based – трафик перенаправляется на основе маршрутов, шлюзом у которых является туннельный интерфейс.
15	Указать действие для DPD (необязательно).	<code>rtt(config-ike-gw) # dead-peer-detection action <MODE></code>	<MODE> – режим работы DPD: <ul style="list-style-type: none"> • restart – соединение переустанавливается; • clear – соединение останавливается; • hold – соединение поддерживается; • none – механизм выключен, никаких действий не предпринимается.
16	Указать интервал между отправкой сообщений механизмом DPD (необязательно).	<code>rtt(config-ike-gw) # dead-peer-detection interval <SEC></code>	<SEC> – интервал между отправкой сообщений механизмом DPD, принимает значения [1..180] секунд.

Шаг	Описание	Команда	Ключи
17	Указать период времени ожидания ответа на сообщения механизма DPD (необязательно).	<code>rtt(config-ike-gw)# dead-peer-detection timeout <SEC></code>	<p><SEC> – период времени ожидания ответа на сообщения механизма DPD принимает значения [1..180] секунд.</p> <p>Значение по умолчанию: 30 секунд.</p>
18	Указать базовый период времени ожидания ответа на сообщения (необязательно).	<code>rtt(config-ike-gw)# retransmit timeout <SEC></code>	<p><SEC> – базовый период времени ожидания ответа на сообщения принимает значения [1..30] секунд.</p> <p>Значение по умолчанию: 4 секунды.</p>
19	Указать количество попыток повторной отправки сообщений после наступления таймаута ожидания ответа (необязательно).	<code>rtt(config-ike-gw)# retransmit tries <TRIES></code>	<p><TRIES> – количество попыток повторной отправки сообщений в случае наступления таймаута ожидания ответа принимает значения от 1 до 10.</p> <p>Для первого отправленного сообщения период времени ожидания ответа будет равен базовому периоду, указанному в команде <code>retransmit timeout</code>, а для последующих попыток интервал ожидания будет рассчитан по формуле:</p> <p>"retransmit timeout" * 1.8 ^ (N-1), где N - номер попытки.</p> <p>Значение по умолчанию: 5 попыток.</p>
20	Указать уровень случайного разброса периода ожидания ответа на сообщения (необязательно).	<code>rtt(config-ike-gw)# retransmit jitter <VALUE></code>	<p><VALUE> – максимальный процент разброса значений, принимает значения [0..100].</p> <p>Значение по умолчанию: 0 %</p>
21	Указать ограничение максимального периода времени ожидания ответа на сообщения (необязательно).	<code>rtt(config-ike-gw)# retransmit limit <SEC></code>	<p><SEC> – максимальный период времени ожидания ответа на сообщения принимает значения [15..300] секунд.</p> <p>Значение по умолчанию: 0 секунд, у периода нет верхнего предела.</p>
22	Установить IP-адрес удаленного шлюза IPsec-туннеля.	<code>rtt(config-ike-gw)# remote address <ADDR></code>	<ADDR> – IP-адрес удаленного шлюза.

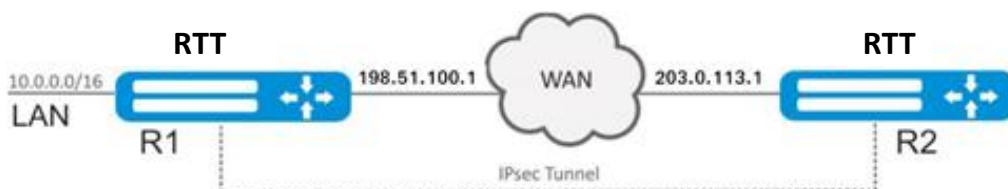
Шаг	Описание	Команда	Ключи
23	Установить IP-адрес подсети получателя, а также IP-протокол и порт.	<pre>rtt(config-ike-gw)# remote network <ADDR/LEN> [protocol { <TYPE> <ID> } [port <PORT>]]</pre>	<p><ADDR/LEN> – IP-адрес и маска подсети отправителя. Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];</p> <p><TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;</p> <p><ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF];</p> <p><PORT> – TCP/UDP-порт, принимает значения [1..65535].</p>
24	Создать в профиль IPsec.	<pre>rtt(config)# security ipsec proposal <NAME></pre>	<NAME> – имя профиля протокола IPsec, задаётся строкой до 31 символа.
25	Определить алгоритм аутентификации для IPsec.	<pre>rtt(config-ipsec- proposal)# authentication algorithm <ALGORITHM></pre>	<ALGORITHM> – алгоритм аутентификации, принимает значения: md5, sha1, sha2-256, sha2-384, sha2-512.
26	Определить алгоритм шифрования для IPsec.	<pre>rtt(config-ipsec- proposal)# encryption algorithm <ALGORITHM></pre>	<ALGORITHM> – протокол шифрования, принимает значения: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.
27	Указать протокол (необязательно).	<pre>rtt(config-ipsec- proposal)#protocol <PROTOCOL></pre>	<p><PROTOCOL> – инкапсулирующий протокол, принимает значения</p> <ul style="list-style-type: none"> • ah – данный протокол осуществляет только аутентификацию трафика, шифрование данных не выполняется; • esp – данный протокол осуществляет аутентификацию и шифрование трафика. <p>Значение по умолчанию: esp.</p>
28	Создать политику для профиля IPsec и перейти в режим её конфигурирования.	<pre>rtt(config)# security ipsec policy <NAME></pre>	<NAME> – имя политики IPsec, задаётся строкой до 31 символа.
29	Привязать политику к профилю.	<pre>rtt(config-ipsec-policy)# proposal <NAME></pre>	<NAME> – имя профиля протокола IPsec, задаётся строкой до 31 символа.

Шаг	Описание	Команда	Ключи
30	Задать время жизни IPsec-туннеля (необязательно).	<code>rtt(config-ipsec-policy)# lifetime { seconds <SEC> packets <PACKETS> kilobytes <KB> }</code>	<p><SEC> – период времени жизни IPsec-туннеля, по истечении которого происходит пересогласование. Принимает значения [1140..86400] секунд.</p> <p><PACKETS> – количество пакетов, после передачи которых происходит пересогласование IPsec-туннеля. Принимает значения [4..86400].</p> <p><KB> – объем трафика, после передачи которого происходит пересогласование IPsec-туннеля. Принимает значения [4..4608000] секунд.</p>
31	Отключить реаутентификацию IKE-сессии (необязательно).	<code>rtt(config-ipsec-policy)# reauthentication disable</code>	
32	Создать IPsec VPN и перейти в режим конфигурирования.	<code>rtt(config)# security ipsecvpn <NAME></code>	<NAME> – имя VPN, задаётся строкой до 31 символа.
33	Определить режим согласования данных, необходимых для активации VPN.	<code>rtt(config-ipsec-vpn)# mode <MODE></code>	<MODE> – режим работы VPN.
34	Привязать IPsec политику к VPN.	<code>rtt(config-ipsec-vpn)#ike ipsec-policy <NAME></code>	<NAME> – имя IPsec-политики, задаётся строка до 31 символа.
35	Задать значение DSCP для использования в IP-заголовке исходящих пакетов IKE-протокола (необязательно).	<code>rtt(config-ipsec-vpn)#ike dscp <DSCP></code>	DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].
36	Установить режим активации VPN.	<code>rtt(config-ipsec-vpn)#ike establish-tunnel <MODE></code>	<p><MODE> – режим активации VPN:</p> <ul style="list-style-type: none"> • by - request – соединение активируется встречной стороной; • route – соединение активируется при появлении трафика, маршрутизируемого в туннель; • immediate – туннель активируется автоматически после применения конфигурации.

Шаг	Описание	Команда	Ключи
37	Осуществить привязка IKE-шлюза к VPN.	<code>rtt(config-ipsec-vpn) # ike gateway <NAME></code>	<NAME> – имя IKE-шлюза, задаётся строкой до 31 символа.
38	Установить значение временного интервала в секундах, по истечению которого соединение закрывается, если не было принято или передано ни одного пакета через SA (необязательно).	<code>rtt(config-ipsec-vpn) # ike idle-time <TIME></code>	<TIME> – интервал в секундах, принимает значения [4..86400].
39	Отключить пересогласование ключей до разрыва IKE-соединения по истечению времени, количеству переданных пакетов или байт (необязательно).	<code>rtt(config-ipsec-vpn) #ike rekey disable</code>	
40	Настроить начало пересогласования ключей IKE-соединения до истечения времени жизни (необязательно).	<code>rtt(config-ipsec-vpn) # Ike rekey margin { seconds <SEC> packets <PACKETS> kilobytes <KB> }</code>	<p><SEC> – интервал времени в секундах, оставшийся до закрытия соединения (задается командой <code>lifetimeseconds</code>) . Принимает значения [4..86400].</p> <p><PACKETS> – количество пакетов, оставшихся до закрытия соединения (задается командой <code>lifetimepackets</code>). Принимает значения [4..86400].</p> <p><KB> – объем трафика в килобайтах, оставшийся до закрытия соединения (задается командой <code>lifetimekilobytes</code>). Принимает значения [4..86400]</p>
41	Установить уровень случайного разброса значений параметров <code>marginseconds</code> , <code>marginpackets</code> , <code>marginkilobytes</code> (необязательно).	<code>rtt(config-ipsec-vpn) # ike rekey randomization <VALUE></code>	<VALUE> – максимальный процент разброса значений, принимает значения [1..100].
42	Описать VPN (необязательно).	<code>rtt(config-ipsec-vpn) # description <DESCRIPTION></code>	<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.
43	Активировать IPsec VPN.	<code>rtt(config-ipsec-vpn) # enable</code>	

9.4.4. Пример настройки Policy-based IPsec VPN с аутентификацией по общему

Задача:



Настроить IPsec-туннель между R1 и R2.

R1 IP-адрес – 198.51.100.1;

R2 IP-адрес – 203.0.113.1;

IKE:

- группа Диффи-Хэллмана: 2;
- алгоритм шифрования: AES 128 bit;
- алгоритм аутентификации: MD5;
- аутентификация по общему известному ключу.

IPsec:

- алгоритм шифрования: AES 128 bit;
- алгоритм аутентификации: MD5.

Решение:



В firewall необходимо разрешить протокол ESP, UDP-порт 500 (для протокола ISAKMP), UDP-порт 4500 (для IPsec трафика при наличии NAT между IPsec соседями).

1. Конфигурирование R1

Настроим IP-адрес на внешнем сетевом интерфейсе:

```
rtt# configure
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# ip address 198.51.100.1/24
rtt(config-if-gi)# exit
```

Создадим набор алгоритмов для протокола IKE. В наборе укажем группу Диффи-Хэллмана 2, алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IKE-соединения:

```
rtt(config)# security ike proposal ike_prop1
rtt(config-ike-proposal)# dh-group 2
```

```
rtt(config-ike-proposal)# authentication algorithm md5
rtt(config-ike-proposal)# encryption algorithm aes128
rtt(config-ike-proposal)# exit
```

Создадим политику протокола IKE. В политике укажем ранее созданный набор алгоритмов, а также укажем общий известный ключ, который будет использован при аутентификации IKE-сессии:

```
rtt(config)# security ike policy ike_poll
rtt(config-ike-policy)# pre-shared-key hexadecimal 123FFF
rtt(config-ike-policy)# proposal ike_prop1
rtt(config-ike-policy)# exit
```

Создадим шлюз протокола IKE. В данном разделе привяжем ранее созданную политику протокола IKE, укажем IP-адреса для построения IPsec туннеля и набор локальных и удаленных сетей, трафик между которыми необходимо будет шифровать. Также укажем версию протокола IKE и "policy-based" в качестве режима перенаправления трафика в туннель:

```
rtt(config)# security ike gateway ike_gw1
rtt(config-ike-gw)# ike-policy ike_poll
rtt(config-ike-gw)# local address 198.51.100.1
rtt(config-ike-gw)# local network 10.0.0.0/16
rtt(config-ike-gw)# remote address 203.0.113.1
rtt(config-ike-gw)# remote network 192.0.2.0/24
rtt(config-ike-gw)# mode policy-based
rtt(config-ike-gw)# exit
```

Создадим набор алгоритмов для IPsec-туннеля. В нем укажем алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IPsec-туннеля:

```
rtt(config)# security ipsec proposal ipsec_prop1
rtt(config-ipsec-proposal)# authentication algorithm md5
rtt(config-ipsec-proposal)# encryption algorithm aes128
rtt(config-ipsec-proposal)# exit
```

Создадим политику для IPsec-туннеля. В политике указывается ранее описанный набор алгоритмов для IPsec-туннеля.

```
rtt(config)# security ipsec policy ipsec_poll
rtt(config-ipsec-policy)# proposal ipsec_prop1
rtt(config-ipsec-policy)# exit
```

Создадим IPsec VPN. В VPN указывается шлюз IKE-протокола, политика IPsec-туннеля, режим обмена ключами и способ установления соединения. После ввода всех параметров включим туннель командой **enable**:

```
rtt(config)# security ipsec vpn ipsec1
rtt(config-ipsec-vpn)# mode ike
rtt(config-ipsec-vpn)# ike establish-tunnel route
rtt(config-ipsec-vpn)# ike gateway ike_gw1
rtt(config-ipsec-vpn)# ike ipsec-policy ipsec_poll
rtt(config-ipsec-vpn)# enable
rtt(config-ipsec-vpn)# exit
```

```
rtt(config)# exit
```

2. Конфигурирование R2

Настроим IP-адрес на внешнем сетевом интерфейсе:

```
rtt# configure
rtt(config)# interface gi 1/0/1
rtt(config-if)# ip address 203.0.113.1/24
rtt(config-if)# exit
```

Создадим набор алгоритмов для протокола IKE. В наборе укажем группу Диффи-Хэллмана 2, алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IKE-соединения:

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

Создадим политику протокола IKE. В политике укажем ранее созданный набор алгоритмов, а также укажем общий известный ключ, который будет использован при аутентификации IKE-сессии:

```
rtt(config)# security ike policy ike_poll
rtt(config-ike-policy)# pre-shared-key hexadecimal 123FFF
rtt(config-ike-policy)# proposal ike_prop1
rtt(config-ike-policy)# exit
```

Создадим шлюз протокола IKE. В данном разделе привяжем ранее созданную политику протокола IKE, укажем IP-адреса для построения IPsec туннеля и набор локальных и удаленных сетей, трафик между которыми необходимо будет шифровать. Также укажем версию протокола IKE и «policy-based» в качестве режима перенаправления трафика в туннель:

```
rtt(config)# security ike gateway ike_gw1
rtt(config-ike-gw)# ike-policy ike_poll
rtt(config-ike-gw)# remote address 198.51.100.1
rtt(config-ike-gw)# remote network 10.0.0.0/16
rtt(config-ike-gw)# local address 203.0.113.1
rtt(config-ike-gw)# local network 192.0.2.0/24
rtt(config-ike-gw)# mode policy-based
rtt(config-ike-gw)# exit
```

Создадим набор алгоритмов для IPsec-туннеля. В нем укажем алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IPsec-туннеля:

```
rtt(config)# security ipsec proposal ipsec_prop1
rtt(config-ipsec-proposal)# authentication algorithm md5
rtt(config-ipsec-proposal)# encryption algorithm aes128
rtt(config-ipsec-proposal)# exit
```

Создадим политику для IPsec-туннеля. В политике указывается ранее описанный набор алгоритмов для IPsec-туннеля.

```
rtt(config)# security ipsec policy ipsec_poll
rtt(config-ipsec-policy)# proposal ipsec_prop1
rtt(config-ipsec-policy)# exit
```

Создадим IPsec VPN. В VPN указывается шлюз IKE-протокола, политика IPsec-туннеля, режим обмена ключами и способ установления соединения. После ввода всех параметров включим туннель командой **enable**:

```
rtt(config)# security ipsec vpn ipsec1
rtt(config-ipsec-vpn)# mode ike
rtt(config-ipsec-vpn)# ike establish-tunnel route
rtt(config-ipsec-vpn)# ike gateway ike_gw1
rtt(config-ipsec-vpn)# ike ipsec-policy ipsec_poll
rtt(config-ipsec-vpn)# enable
rtt(config-ipsec-vpn)# exit
rtt(config)# exit
```

Состояние туннеля можно посмотреть командой:

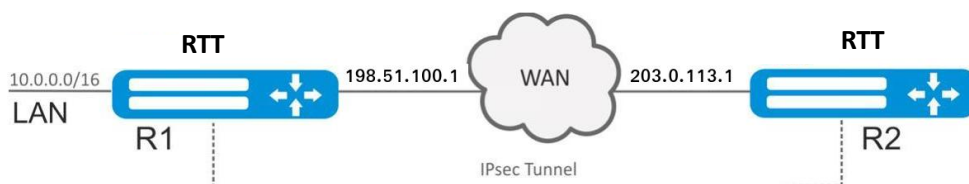
```
rtt# show security ipsec vpn status ipsec1
```

Конфигурацию туннеля можно посмотреть командой:

```
rtt# show security ipsec vpn configuration ipsec1
```

9.4.5. Пример настройки Policy-based IPsec VPN с аутентификацией сертификатам X.509, выдаваемых PKI-клиентом

Задача:



Настроить IPsec-туннель между R1 и R2.

R1 IP-адрес – 198.51.100.1;

R2 IP-адрес – 203.0.113.1;

PKI:

- R1 выступает в роли PKI-сервера – удостоверяющего центра с самоподписанным сертификатом;

- На R1 и R2 настраиваются PKI-клиенты, запрашивающие выпуск сертификатов для IPsec VPN на R1.

IKE:

- группа Диффи-Хэллмана: 2;
- алгоритм шифрования: AES 128 bit;
- алгоритм аутентификации: MD5;
- аутентификация по сертификатам X.509.

IPsec:

- алгоритм шифрования: AES 128 bit;
- алгоритм аутентификации: MD5.

Решение:



В firewall необходимо разрешить протокол ESP, UDP-порт 500 (для протокола ISAKMP), UDP-порт 4500 (для IPsec трафика при наличии NAT между IPsec соседями). Также на стороне R1 необходимо разрешить TCP-порт 80 для доступа PKI-клиентов к PKI-серверу.

1. Конфигурирование R1

Настроим IP-адрес на внешнем сетевом интерфейсе:

```
rtt# configure
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# ip address 198.51.100.1/24
rtt(config-if-gi)# exit
```

Настроим NTP-клиента на получение точного времени от шлюза Интернет-провайдера:

```
rtt(config)# ntp enable
rtt(config)# ntp server 198.51.100.15
rtt(config-ntp-server)# exit
rtt(config)#
```

Настроим PKI-сервер. В нем заполним отличительное имя, challenge-password, время жизни выписываемых клиентских сертификатов и привяжем PKI-сервер к внешнему сетевому интерфейсу:

```
rtt(config)# crypto pki server
rtt(config-pki-server)# subject-name
rtt(config-pki-server-subject-name)# country RU
rtt(config-pki-server-subject-name)# state Moscow
rtt(config-pki-server-subject-name)# locality Moscow
rtt(config-pki-server-subject-name)# organization Company
rtt(config-pki-server-subject-name)# common-name ca.company.loc
rtt(config-pki-server-subject-name)# exit
rtt(config-pki-server)# source-interface gi 1/0/1
rtt(config-pki-server)# challenge-password password
rtt(config-pki-server)# lifetime 7
```

```
rtt(config-pki-server)# enable
rtt(config-pki-server)# exit
rtt(config)#
```

На этом этапе необходимо применить конфигурацию на маршрутизаторе, чтобы получить цифровой отпечаток сертификата PKI-сервера из вывода команды **show crypto pki server**, он понадобится в дальнейшей настройке PKI-клиентов:

```
rtt# show crypto pki server
Status:                Enabled
Lifetime days:         14
Certificate fingerprint: 79:D2:B6:7E:DF:77:2D:C5:27:68:99:10:BA:EC:D2:47
Source:                gigabitethernet 1/0/1
Last issued serial number: --
Challenge password:    Active
RTT.CA#
```

Продолжим дальнейшую настройку PKI-клиента. Необходимо заполнить отличительное имя, URL для подключения к PKI-серверу (в случае маршрутизатора R1 – его собственный IP-адрес, назначенный на внешний интерфейс), ранее полученный цифровой отпечаток сертификата PKI-сервера и доменное имя маршрутизатора в качестве альтернативного имени клиентского сертификата:

```
rtt(config)# crypto pki trustpoint TP_R1
rtt(config-trustpoint)# subject-name
rtt(config-trustpoint-subject-name)# country RU
rtt(config-trustpoint-subject-name)# state Moscow
rtt(config-trustpoint-subject-name)# locality Moscow
rtt(config-trustpoint-subject-name)# organization Company
rtt(config-trustpoint-subject-name)# common-name r1.company.loc
rtt(config-trustpoint-subject-name)# exit
rtt(config-trustpoint)# subject-alt-name
rtt(config-trustpoint-san)# dns r1.company.loc
rtt(config-trustpoint-san)# exit
rtt(config-trustpoint)# url http://198.51.100.1/
rtt(config-trustpoint)# fingerprint
79:D2:B6:7E:DF:77:2D:C5:27:68:99:10:BA:EC:D2:47
rtt(config-trustpoint)# challenge-password password
rtt(config-trustpoint)# enable
rtt(config-trustpoint)# exit
rtt(config)#
```

Перейдем к настройке VPN.

Создадим набор алгоритмов для протокола IKE. В наборе укажем группу Диффи-Хэллмана 2, алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IKE-соединения:

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

Создадим политику протокола IKE. В политике указываем ранее созданный набор алгоритмов, в качестве метода аутентификации выбираем "trustpoint" и в команде **crypto trustpoint** указываем имя ранее созданного PKI-клиента, выписываемые им сертификаты будут использованы при аутентификации IKE-сессии:

```
rtt(config)# security ike policy ike_poll
rtt(config-ike-policy)# proposal ike_prop1
rtt(config-ike-policy)# authentication method trustpoint
rtt(config-ike-policy)# crypto trustpoint TP_R1
rtt(config-ike-policy)# exit
```

Создадим шлюз протокола IKE. В данном разделе привяжем ранее созданную политику протокола IKE, укажем IP-адреса для построения IPsec туннеля и набор локальных и удаленных сетей, трафик между которыми необходимо будет шифровать. Также укажем версию протокола IKE и «policy-based» в качестве режима перенаправления трафика в туннель:



Важным моментом при настройке аутентификации по сертификатам X.509 является указание корректных local id и remote id, присутствующих в сертификатах в качестве альтернативных имен сертификата. Поскольку ранее в настройках PKI-клиента в качестве альтернативного имени было указано полное доменное имя маршрутизатора – укажем его и в командах local id и remote id.

```
rtt(config)# security ike gateway ike_gw1
rtt(config-ike-gw)# ike-policy ike_poll
rtt(config-ike-gw)# local address 198.51.100.1
rtt(config-ike-gw)# local network 10.0.0.0/16
rtt(config-ike-gw)# remote address 203.0.113.1
rtt(config-ike-gw)# remote network 192.0.2.0/24
rtt(config-ike-gw)# local id dns r1.company.loc
rtt(config-ike-gw)# remote id dns r2.company.loc
rtt(config-ike-gw)# mode policy-based
rtt(config-ike-gw)# exit
```

Создадим набор алгоритмов для IPsec-туннеля. В нем укажем алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IPsec-туннеля:

```
rtt(config)# security ipsec proposal ipsec_prop1
rtt(config-ipsec-proposal)# authentication algorithm md5
rtt(config-ipsec-proposal)# encryption algorithm aes128
rtt(config-ipsec-proposal)# exit
```

Создадим политику для IPsec-туннеля. В политике указывается ранее описанный набор алгоритмов для IPsec-туннеля.

```
rtt(config)# security ipsec policy ipsec_poll
rtt(config-ipsec-policy)# proposal ipsec_prop1
rtt(config-ipsec-policy)# exit
```

Создадим IPsec VPN. В VPN указываются шлюз IKE-протокола, политика IPsec-туннеля, режим обмена ключами и способ установления соединения. После ввода всех параметров включим туннель командой **enable**:


```
rtt(config)# security ipsec vpn ipsec1
rtt(config-ipsec-vpn)# mode ike
rtt(config-ipsec-vpn)# ike establish-tunnel route
rtt(config-ipsec-vpn)# ike gateway ike_gw1
rtt(config-ipsec-vpn)# ike ipsec-policy ipsec_poll
rtt(config-ipsec-vpn)# enable
rtt(config-ipsec-vpn)# exit
rtt(config)# exit
```

2. Конфигурирование R2

Настроим IP-адрес на внешнем сетевом интерфейсе:

```
rtt# configure
rtt(config)# interface gi 1/0/1
rtt(config-if)# ip address 203.0.113.1/24
rtt(config-if)# exit
```

Произведем настройку настройки PKI-клиента. Необходимо заполнить отличительное имя, URL для подключения к PKI-серверу (в случае маршрутизатора R2 – внешний IP-адрес маршрутизатора R1), ранее полученный цифровой отпечаток сертификата PKI-сервера и доменное имя маршрутизатора в качестве альтернативного имени клиентского сертификата:

```
rtt(config)# crypto pki trustpoint TP_R2
rtt(config-trustpoint)# subject-name
rtt(config-trustpoint-subject-name)# country RU
rtt(config-trustpoint-subject-name)# state Moscow
rtt(config-trustpoint-subject-name)# locality Moscow
rtt(config-trustpoint-subject-name)# organization Company
rtt(config-trustpoint-subject-name)# common-name r2.company.loc
rtt(config-trustpoint-subject-name)# exit
rtt(config-trustpoint)# subject-alt-name
rtt(config-trustpoint-san)# dns r2.company.loc
rtt(config-trustpoint-san)# exit
rtt(config-trustpoint)# url http://198.51.100.1/
rtt(config-trustpoint)# fingerprint
79:D2:B6:7E:DF:77:2D:C5:27:68:99:10:BA:EC:D2:47
rtt(config-trustpoint)# challenge-password password
rtt(config-trustpoint)# enable
rtt(config-trustpoint)# exit
rtt(config)#
```

Перейдем к настройке VPN.

Создадим набор алгоритмов для протокола IKE. В наборе укажем группу Диффи-Хэллмана 2, алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IKE-соединения:

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

Создадим политику протокола IKE. В политике указываем ранее созданный набор алгоритмов, в качестве метода аутентификации выбираем «trustpoint» и в команде **crypto trustpoint** указываем имя ранее созданного PKI-клиента, выписываемые им сертификаты будут использованы при аутентификации IKE-сессии:

```
rtt(config)# security ike policy ike_poll
rtt(config-ike-policy)# proposal ike_prop1
rtt(config-ike-policy)# authentication method trustpoint
rtt(config-ike-policy)# crypto trustpoint TP_R2
rtt(config-ike-policy)# exit
```

Создадим шлюз протокола IKE. В данном разделе привяжем ранее созданную политику протокола IKE, укажем IP-адреса для построения IPsec туннеля и набор локальных и удаленных сетей, трафик между которыми необходимо будет шифровать. Также укажем версию протокола IKE и «policy-based» в качестве режима перенаправления трафика в туннель:



Важным моментом при настройке аутентификации по сертификатам X.509 является указание корректных local id и remote id, присутствующих в сертификатах в качестве альтернативных имен сертификата. Поскольку ранее в настройках PKI-клиента в качестве альтернативного имени было указано полное доменное имя маршрутизатора – укажем его и в командах local id и remote id.

```
rtt(config)# security ike gateway ike_gw1
rtt(config-ike-gw)# ike-policy ike_poll
rtt(config-ike-gw)# remote address 198.51.100.1
rtt(config-ike-gw)# remote network 10.0.0.0/16
rtt(config-ike-gw)# local address 203.0.113.1
rtt(config-ike-gw)# local network 192.0.2.0/24
rtt(config-ike-gw)# local id dns r2.company.loc
rtt(config-ike-gw)# remote id dns r1.company.loc
rtt(config-ike-gw)# mode policy-based
rtt(config-ike-gw)# exit
```

Создадим набор алгоритмов для IPsec-туннеля. В нем укажем алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IPsec-туннеля:

```
rtt(config)# security ipsec proposal ipsec_prop1
rtt(config-ipsec-proposal)# authentication algorithm md5
rtt(config-ipsec-proposal)# encryption algorithm aes128
rtt(config-ipsec-proposal)# exit
```

Создадим политику для IPsec-туннеля. В политике указывается ранее описанный набор алгоритмов для IPsec-туннеля.

```
rtt(config)# security ipsec policy ipsec_poll
rtt(config-ipsec-policy)# proposal ipsec_prop1
rtt(config-ipsec-policy)# exit
```

Создадим IPsec VPN. В VPN указываются шлюз IKE-протокола, политика IPsec-туннеля, режим обмена ключами и способ установления соединения. После ввода всех параметров включим туннель командой **enable**:

```

rtt(config)# security ipsec vpn ipsec1
rtt(config-ipsec-vpn)# mode ike
rtt(config-ipsec-vpn)# ike establish-tunnel route
rtt(config-ipsec-vpn)# ike gateway ike_gw1
rtt(config-ipsec-vpn)# ike ipsec-policy ipsec_poll
rtt(config-ipsec-vpn)# enable
rtt(config-ipsec-vpn)# exit
rtt(config)# exit

```

Состояние туннеля можно посмотреть командой:

```
rtt# show security ipsec vpn status ipsec1
```

Конфигурацию туннеля можно посмотреть командой:

```
rtt# show security ipsec vpn configuration ipsec1
```

Состояние PKI-клиента и время следующей процедуры автоматического перевыпуска сертификата можно посмотреть командой:

```
rtt# show crypto pki trustpoint TP_R1
```

9.4.6. Алгоритм настройки Remote Access IPsec VPN

Remote Access IPsec VPN – сценарий организации временных VPN-подключений, в котором сервер IPsec VPN находится в режиме ожидания входящих подключений, а клиенты осуществляют временные подключения к серверу для получения доступа к сетевым ресурсам.

Дополнительной особенностью RA IPsec VPN является возможность использования двухфакторной аутентификации IPsec, где первым фактором аутентификации является Extended Authentication (XAUTH) или Extensible Authentication Protocol (EAP), а вторым фактором аутентификации является пара логин-пароль для клиента IPsec VPN.

Шаг	Описание	Команда	Ключи
1	Создать IKE-экземпляр и перейти в режим его конфигурирования.	<code>rtt(config)# security ike proposal <NAME></code>	<NAME> – имя профиля протокола IKE, задаётся строкой до 31 символа.
2	Указать описание конфигурируемого туннеля (необязательно).	<code>rtt(config-ike-proposal)# description <DESCRIPTION></code>	<DESCRIPTION> – описание туннеля, задаётся строкой до 255 символов.
3	Определить алгоритм аутентификации для IKE (необязательно).	<code>rtt(config-ike-proposal)# authentication algorithm <ALGORITHM></code>	<ALGORITHM> – алгоритм аутентификации, принимает значения: md5, sha1, sha2-256, sha2-384, sha2-512. Значение по умолчанию: sha1.

Шаг	Описание	Команда	Ключи
4	Установить IP-адрес локальной стороны VTI-туннеля (необязательно).	<pre>rtt(config-vti)# ip address <ADDR/LEN> [unit <ID>]</pre> <p>или</p> <pre>rtt(config-vti)# ip address <ADDR/LEN> secondary [unit <ID>]</pre>	<p><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..31];</p> <p><ID> – номер юнита, принимает значения [1..4].</p> <p>Ключ secondary указывает, что настроенный адрес является дополнительным IP-адресом. Если это ключевое слово отсутствует, настроенный адрес является основным IP-адресом. Возможно указать до 7 дополнительных IP-адресов.</p>
5	Определить номер группы Диффи-Хэллмана (необязательно).	<pre>rtt(config-ike- proposal)# dh-group <DH-GROUP></pre>	<p><DH-GROUP> – номер группы Диффи-Хэллмана, принимает значения [1, 2, 5, 14, 15, 16, 17, 18].</p> <p>Значение по умолчанию: 1.</p>
6	Создать политику для профиля IKE и перейти в режим её конфигурирования.	<pre>rtt(config)# security ike policy <NAME></pre>	<p><NAME> – имя политики IKE, задаётся строкой до 31 символа.</p>
7	Определить режим аутентификации.	<pre>rtt(config-ike- policy)# authentication method <METHOD></pre>	<p><METHOD> – метод аутентификации ключа. Может принимать значения:</p> <ul style="list-style-type: none"> • xauth - psk - key – метод двухфакторной аутентификации, использующий пару логин-пароль и предварительно полученные ключи шифрования; • eap - метод двухфакторной аутентификации, использующий пару логин-пароль и предварительно полученные сертификаты.
8	Задать режим клиента (только для клиента).	<pre>rtt(config-ike- policy)# authentication mode client</pre>	
9	Задать время жизни соединения протокола IKE (необязательно).	<pre>rtt(config-ike- policy)# lifetime seconds <SEC></pre>	<p><SEC> – период времени, принимает значения [4 ..86400] секунд.</p> <p>Значение по умолчанию: 3600.</p>
10	Отключить реаутентификацию IKE сессии (необязательно).	<pre>rtt(config-ipsec- policy)# reauthentication disable</pre>	

Шаг	Описание	Команда	Ключи
11	Привязать политику к профилю.	<code>rtt(config-ike-policy)# proposal <NAME></code>	<NAME> – имя профиля протокола IKE, задаётся строкой до 31 символа.
12	Указать ключ аутентификации.	<code>rtt(config-ike-policy)# pre-shared-key ascii-text <TEXT></code>	<TEXT> – строка [1..64] ASCII символов.
13	Указать сертификаты и ключи (только для метода EAP)	<code>rtt(config-ike-policy)# crypto <CERTIFICATE-TYPE> <NAME></code>	<p><CERTIFICATE-TYPE> – тип сертификата или ключа, может принимать следующие значения:</p> <ul style="list-style-type: none"> • ca – сертификат удостоверяющего сервера; • crl – список отозванных сертификатов; • local-crt – публичный сертификат сервера удалённого доступа; • local-crt-key – приватный ключ сервера удалённого доступа. <p><NAME> – имя сертификата или ключа, задаётся строкой до 31 символа.</p> <p>На стороне сервера необходимо добавить набор сертификатов (ca, local-crt, local-key).</p> <p>На стороне клиента необходимо указать только корневой сертификат (ca).</p>
14	Создать профиль доступа.	<code>rtt(config)# access profile <NAME></code>	<NAME> – имя профиля доступа, задаётся строкой до 31 символа.
15	Создать имя пользователя.	<code>rtt(config-access-profile)# user <LOGIN></code>	<LOGIN> – логин клиента, задаётся строкой до 31 символа.
16	Задать пароль пользователя.	<code>rtt(config-profile)# password ascii-text <TEXT></code>	<TEXT> – строка [8..32] ASCII символов.
17	Создать пул адресов назначения (только для сервера).	<code>rtt(config)# address-assignment pool <NAME></code>	<NAME> – имя пула адресов назначения, задаётся строкой до 31 символа.
18	Задать подсеть, из которой будут выдаваться IP клиентам (только для сервера).	<code>rtt(config-pool)# ip prefix <ADDR/LEN></code>	<ADDR/LEN> – адрес подсети и префикс.
19	Создать шлюз для IKE и перейти в режим его конфигурирования.	<code>rtt(config)# security ike gateway <NAME></code>	<NAME> – имя шлюза протокола IKE, задаётся строкой до 31 символа.
20	Привязать политику IKE.	<code>rtt(config-ike-gw)# ike-policy <NAME></code>	<NAME> – имя политики протокола IKE, задаётся строкой до 31 символа.

Шаг	Описание	Команда	Ключи
21	Установить режим перенаправления трафика в туннель.	<code>rtt(config-ike-gw)# mode <MODE></code>	<p><MODE> – режим перенаправления трафика в туннель, принимает значения:</p> <ul style="list-style-type: none"> • policy - based – трафик перенаправляется на основе принадлежности к указанным в политиках подсетям.
22	Указать действие для DPD (необязательно).	<code>rtt(config-ike-gw)# dead-peer-detection action <MODE></code>	<p><MODE> – режим работы DPD:</p> <ul style="list-style-type: none"> • restart – соединение переустанавливается; • clear – соединение останавливается; • hold – соединение поддерживается; • none – механизм выключен, никаких действий не предпринимается. <p>Значение по умолчанию: none.</p>
23	Указать интервал между отправкой сообщений механизмом DPD (необязательно).	<code>rtt(config-ike-gw)#dead-peer- detection interval <SEC></code>	<p><SEC> – интервал между отправкой сообщений механизмом DPD, принимает значения [1..180] секунд.</p> <p>Значение по умолчанию: 2.</p>
24	Указать период времени ожидания ответа на сообщения механизма DPD (необязательно).	<code>rtt(config-ike-gw)# dead-peer-detection timeout <SEC></code>	<p><SEC> – период времени ожидания ответа на сообщения механизма DPD принимает значения [1..180] секунд.</p> <p>Значение по умолчанию: 30 секунд.</p>
25	Указать базовый период времени ожидания ответа на сообщения (необязательно).	<code>rtt(config-ike-gw)# retransmit timeout <SEC></code>	<p><SEC> – базовый период времени ожидания ответа на сообщения принимает значения [1..30] секунд.</p> <p>Значение по умолчанию: 4 секунды.</p>
26	Указать количество попыток повторной отправки сообщений после наступления таймаута ожидания ответа (необязательно).	<code>rtt(config-ike-gw)# retransmit tries <TRIES></code>	<p><TRIES> – количество попыток повторной отправки сообщений в случае наступления таймаута ожидания ответа принимает значения от 1 до 10.</p> <p>Для первого отправленного сообщения период времени ожидания ответа будет равен базовому периоду, указанному в команде retransmit timeout, а для последующих попыток интервал ожидания будет рассчитан по формуле:</p> <p>"retransmit timeout" * 1.8 ^ (N-1), где N - номер попытки.</p> <p>Значение по умолчанию: 5 попыток.</p>

Шаг	Описание	Команда	Ключи
27	Указать уровень случайного разброса периода ожидания ответа на сообщения (необязательно).	<code>rtt(config-ike-gw)# retransmit jitter <VALUE></code>	<VALUE> – максимальный процент разброса значений, принимает значения [0..100]. Значение по умолчанию: 0 %
28	Указать ограничение максимального периода времени ожидания ответа на сообщения (необязательно).	<code>rtt(config-ike-gw)# retransmit limit <SEC></code>	<SEC> – максимальный период времени ожидания ответа на сообщения принимает значения [15..300] секунд. Значение по умолчанию: 0 секунд, у периода нет верхнего предела.
29	Задать пул динамического выделения IP-адресов клиентам (только для сервера).	<code>rtt(config-ike-gw)# remote network dynamic pool <NAME></code>	<NAME> – имя пула адресов назначения, задаётся строкой до 31 символа.
30	Задать режим динамического установления удаленной подсети (только для клиента).	<code>rtt(config-ike-gw)# remote network dynamic client</code>	
31	Задать профиль доступа (только для сервера).	<code>rtt(config-ike-gw)# access-profile <NAME></code>	<NAME> – имя профиля доступа, задаётся строкой до 31 символа.
32	Задать профиль доступа и логин (только для клиента).	<code>rtt(config-ike-gw)# access-profile <NAME> client <LOGIN></code>	<NAME> – имя профиля доступа, задаётся строкой до 31 символа; <LOGIN> – логин клиента, задаётся строкой до 31 символа.
33	Задать интерфейс терминирования выделенного IP для построения IPsec VPN (только для клиента).	<code>rtt(config-ike-gw)# assign-interface loopback <INDEX></code>	<INDEX> – индекс интерфейса, принимает значения [1..65535].
34	Создать профиль IPsec.	<code>rtt(config)# security ipsec proposal <NAME></code>	<NAME> – имя профиля протокола IPsec, задаётся строкой до 31 символа.
35	Определить алгоритм аутентификации для IPsec (необязательно).	<code>rtt(config-ipsec- proposal)# authentication algorithm <ALGORITHM></code>	<ALGORITHM> – алгоритм аутентификации, принимает значения: md5, sha1, sha2-256, sha2-384, sha2-512. Значение по умолчанию: sha1.

Шаг	Описание	Команда	Ключи
36	Определить алгоритм шифрования для IPsec (необязательно).	<code>rtt(config-ipsec-proposal)# encryption algorithm <ALGORITHM></code>	<p><ALGORITHM> – протокол шифрования, принимает значения: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.</p> <p>Значение по умолчанию: 3des.</p>
37	Указать протокол (необязательно).	<code>rtt(config-ipsec-proposal)# protocol <PROTOCOL></code>	<p><PROTOCOL> – инкапсулирующий протокол, принимает значения:</p> <ul style="list-style-type: none"> ah – данный протокол осуществляет только аутентификацию трафика, шифрование данных не выполняется; esp – данный протокол осуществляет аутентификацию и шифрование трафика. <p>Значение по умолчанию: esp.</p>
38	Задать config-ipsec-proposal конфигурирования.	<code>rtt(config)# security ipsec policy <NAME></code>	<NAME> – имя политики IPsec, задаётся строкой до 31 символа.
39	Привязать политику к профилю.	<code>rtt(config-ipsec-policy)# proposal <NAME></code>	<NAME> – имя профиля протокола IPsec, задаётся строкой до 31 символа.
40	Задать время жизни IPsec-туннеля (необязательно).	<code>rtt(config-ipsec-policy)# lifetime { seconds <SEC> packets <PACKETS> kilobytes <KB> }</code>	<p><SEC> – период времени жизни IPsec-туннеля, по истечении которого происходит пересогласование.</p> <p>Принимает значения [1140..86400] секунд.</p> <p>Значение по умолчанию: 540.</p> <p><PACKETS> – количество пакетов, после передачи которых происходит пересогласование IPsec-туннеля.</p> <p>Принимает значения [4..86400].</p> <p>Значение по умолчанию: отключено.</p> <p><KB> – объем трафика, после передачи которого происходит пересогласование IPsec-туннеля. Принимает значения [4..4608000] секунд.</p> <p>Значение по умолчанию: отключено.</p>

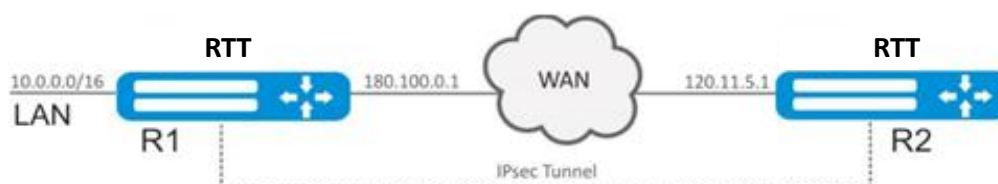
Шаг	Описание	Команда	Ключи
41	Создать IPsec VPN и перейти в режим конфигурирования.	<code>rtt(config)# security ipsec vpn <NAME></code>	<NAME> – имя VPN, задаётся строкой до 31 символа.
42	Определить режим согласования данных, необходимых для активации VPN.	<code>rtt(config-ipsec-vpn)# mode <MODE></code>	<MODE> – режим работы VPN, принимает значения: <code>ike</code> , <code>manual</code> .
43	Привязать IPsec политику к VPN.	<code>rtt(config-ipsec-vpn)#ike ipsec-policy <NAME></code>	<NAME> – имя IPsec-политики, задаётся строка до 31 символа.
44	Задать значение DSCP для использования в IP-заголовке исходящих пакетов IKE-протокола (необязательно).	<code>rtt(config-ipsec-vpn)#ike dscp <DSCP></code>	DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63]. Значение по умолчанию: 63.
45	Устанавливается режим активации VPN.	<code>rtt(config-ipsec-vpn)#ike establish-tunnel <MODE></code>	<MODE> – режим активации VPN: <ul style="list-style-type: none"> • by - request – соединение активируется встречной стороной, доступно для сервера; • route – соединение активируется при появлении трафика, маршрутизируемого в туннель, доступно для сервера; • immediate – туннель активируется автоматически после применения конфигурации, доступно для клиента.
46	Осуществить привязку IKE-шлюза к VPN.	<code>rtt(config-ipsec-vpn)# ike gateway <NAME></code>	<NAME> – имя IKE-шлюза, задаётся строкой до 31 символа.
47	Установить значение временного интервала в секундах, по истечению которого соединение закрывается, если не было принято или передано ни одного пакета через SA (необязательно).	<code>rtt(config-ipsec-vpn)# ike idle-time <TIME></code>	<TIME> – интервал в секундах, принимает значения [4..86400]. Значение по умолчанию: 0.
48	Отключить пересогласование ключей до разрыва IKE-соединения по истечению времени, количеству переданных пакетов или байт (необязательно).	<code>rtt(config-ipsec-vpn)#ike rekey disable</code>	Значение по умолчанию: включено.

Шаг	Описание	Команда	Ключи
49	Настроить начало пересогласования ключей IKE-соединения до истечения времени жизни (необязательно).	<code>rtt(config-ipsec-vpn) # ike rekey margin { seconds <SEC> packets <PACKETS> kilobytes <KB> }</code>	<p><SEC> – интервал времени в секундах, оставшийся до закрытия соединения (задается командой <code>lifetimeseconds</code>) . Принимает значения [4..86400]. Значение по умолчанию: 540</p> <p><PACKETS> – количество пакетов, оставшихся до закрытия соединения (задается командой <code>lifetimepackets</code>). Принимает значения [4..86400]. Значение по умолчанию: отключено.</p> <p><KB> – объем трафика в килобайтах, оставшийся до закрытия соединения (задается командой <code>lifetimekilobytes</code>). Принимает значения [4..86400] Значение по умолчанию: отключено.</p>
50	Установить уровень случайного разброса значений параметров <code>marginseconds</code> , <code>marginpackets</code> , <code>marginkilobytes</code> (необязательно).	<code>rtt(config-ipsec-vpn) # ike rekey randomization <VALUE></code>	<p><VALUE> – максимальный процент разброса значений, принимает значения [1..100]. Значение по умолчанию: 100.</p>
51	Описать VPN (необязательно).	<code>rtt(config-ipsec-vpn) # description <DESCRIPTION></code>	<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.
52	Активировать IPsec VPN.	<code>rtt(config-ipsec-vpn) # enable</code>	

Шаг	Описание	Команда	Ключи
53	Включить режим переподключения клиентов XAUTH с одним логином/паролем (только для сервера) (необязательно).	<pre>rtt(config-ipsec- vpn)# security ike session uniqueids <MODE></pre>	<p><MODE> – режим переподключения, принимает следующие значения:</p> <ul style="list-style-type: none"> no – установленное подключение XAUTH будет удалено, если для нового подключения XAUTH инициатором соединения будет отправлено уведомление "INITIAL_CONTACT", будет назначен ранее использованный IP-адрес. В противном случае, установленное соединение XAUTH будет удержано. Для нового подключения XAUTH будет назначен новый IP-адрес. never – установленное подключение XAUTH будет удержано. Для нового подключения XAUTH будет назначен новый IP-адрес. Уведомление "INITIAL_CONTACT" будет в любом случае проигнорировано. replace – установленное подключение XAUTH будет удалено. Для нового подключения XAUTH будет использован ранее использованный IP-адрес. keep – установленное подключение XAUTH будет удержано. Новое подключение XAUTH будет отклонено.

9.4.7. Пример настройки Remote Access IPsec VPN

Задача:



Настроить Remote Access IPsec VPN между R1 и R2 с использованием второго фактора аутентификации IPsec - XAUTH. В качестве сервера IPsec VPN настроить маршрутизатор R1, а маршрутизатор R2 в качестве клиента IPsec VPN.

R2 IP-адрес – 120.11.5.1;

R1 IP-адрес – 180.100.0.1;

Клиентам IPsec VPN:

- выдавать адреса из пула подсети 192.0.2.0/24
- предоставлять доступ до LAN подсети 10.0.0.0/16

IKE:

- группа Диффи-Хэллмана: 2;
- алгоритм шифрования: 3DES;
- алгоритм аутентификации: SHA1.

IPsec:

- алгоритм шифрования: 3DES;
- алгоритм аутентификации: SHA1.

XAUTH:

- логин: client1;
- пароль: password123.

Решение:



Предварительно в firewall необходимо разрешить протокол ESP и ISAKMP (UDP-порт 500, 4500).

1. Конфигурирование R1

Настроим внешний сетевой интерфейс и определим принадлежность к зоне безопасности:

```
rtt# configure
rtt(config)# security zone untrusted
rtt(config-zone)# exit
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# security-zone untrusted
rtt(config-if-gi)# ip address 180.100.0.1/24
rtt(config-if-gi)# exit
```

Для настройки правил зон безопасности потребуется создать профиль порта протокола ISAKMP:

```
rtt(config)# object-group service ISAKMP
rtt(config-object-group-service)# port-range 500,4500
rtt(config-object-group-service)# exit
```

Создадим профиль протокола IKE. В профиле укажем группу Диффи-Хэллмана 2, алгоритм шифрования 3 DES, алгоритм аутентификации SHA1. Данные параметры безопасности используются для защиты IKE-соединения:

```
rtt(config)# security ike proposal IKEPROP
rtt(config-ike-proposal)# dh-group 2
```

```
rtt(config-ike-proposal)# authentication algorithm sha1
rtt(config-ike-proposal)# encryption algorithm 3des
rtt(config-ike-proposal)# exit
```

Создадим политику протокола IKE. В политике указывается список профилей протокола IKE, по которым могут согласовываться узлы, ключ аутентификации и метод аутентификации XAUTH по ключу:

```
rtt(config)# security ike policy IKEPOLICY
rtt(config-ike-policy)# pre-shared-key hexadecimal 123FFF
rtt(config-ike-policy)# authentication method xauth-psk-key
rtt(config-ike-policy)# proposal IKEPROP
rtt(config-ike-policy)# exit
```

Создадим профиль доступа и заведем в нем пару логин и пароль для клиента IPsec VPN:

```
rtt(config)# access profile XAUTH
rtt(config-access-profile)# user client1
rtt(config-profile)# password ascii-text password123
rtt(config-profile)# exit
rtt(config-access-profile)# exit
```

Создадим пул адресов назначения, из которого будут выдаваться IP клиентам IPsec VPN:

```
rtt(config)# address-assignment pool CLIENT_POOL
rtt(config-pool)# ip prefix 192.0.2.0/24
rtt(config-pool)# exit
```

Создадим шлюз протокола IKE. В данном профиле необходимо указать политику протокола IKE, указать локальную подсеть, в качестве удаленной подсети указать пул адресов назначения, задать режим перенаправления трафика в туннель по политике и использование второго фактора аутентификации XAUTH:

```
rtt(config)# security ike gateway IKEGW
rtt(config-ike-gw)# ike-policy IKEPOLICY
rtt(config-ike-gw)# local address 180.100.0.1
rtt(config-ike-gw)# local network 10.0.0.0/16
rtt(config-ike-gw)# remote address any
rtt(config-ike-gw)# remote network dynamic pool CLIENT_POOL
rtt(config-ike-gw)# dead-peer-detection action clear
rtt(config-ike-gw)# mode policy-based
rtt(config-ike-gw)# access-profile XAUTH
rtt(config-ike-gw)# exit
```

Создадим профиль параметров безопасности для IPsec-туннеля. В профиле укажем алгоритм шифрования 3DES, алгоритм аутентификации SHA1. Данные параметры безопасности используются для защиты IPsec-туннеля:

```
rtt(config)# security ipsec proposal IPSECPROP
rtt(config-ipsec-proposal)# authentication algorithm sha1
rtt(config-ipsec-proposal)# encryption algorithm 3des
rtt(config-ipsec-proposal)# exit
```

Создадим политику для IPsec-туннеля. В политике указывается список профилей IPsec-туннеля, по которым могут согласовываться узлы.

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

Создадим IPsec VPN. В VPN указывается шлюз IKE-протокола, политика IPsec-туннеля, режим обмена ключами и режим ожидания входящего соединения IPsec – by-request. После ввода всех параметров включим туннель командой **enable**:

```
rtt(config)# security ipsec IPSECVPN
rtt(config-ipsec-vpn)# mode ike
rtt(config-ipsec-vpn)# ike establish-tunnel by-request
rtt(config-ipsec-vpn)# ike gateway IKEGW
rtt(config-ipsec-vpn)# ike ipsec-policy IPSECPOLICY
rtt(config-ipsec-vpn)# enable
rtt(config-ipsec-vpn)# exit
```

Разрешим протокол esp и udp порты 500, 4500 в конфигурации firewall для установления IPsec VPN:

```
rtt(config)# security zone-pair untrusted self
rtt(config-zone-pair)# rule 1
rtt(config-zone-pair-rule)# action permit
rtt(config-zone-pair-rule)# match protocol udp
rtt(config-zone-pair-rule)# match destination-port object-group ISAKMP
rtt(config-zone-pair-rule)# enable
rtt(config-zone-pair-rule)# exit
rtt(config-zone-pair)# rule 2
rtt(config-zone-pair-rule)# action permit
rtt(config-zone-pair-rule)# match protocol esp
rtt(config-zone-pair-rule)# enable
rtt(config-zone-pair-rule)# exit
rtt(config-zone-pair)# end
```

2. Конфигурирование R2

Настроим внешний сетевой интерфейс и определим принадлежность к зоне безопасности:

```
rtt# configure
rtt(config)# interface gi 1/0/1
rtt(config-if)# ip address 120.11.5.1/24
rtt(config-if)# security-zone untrusted
rtt(config-if)# exit
```

Для настройки правил зон безопасности потребуется создать профиль порта протокола ISAKMP:

```
rtt(config)# object-group service ISAKMP
rtt(config-addr-set)# port-range 500,4500
rtt(config-addr-set)# exit
```

Создадим профиль протокола IKE. В профиле укажем группу Диффи-Хэллмана 2, алгоритм шифрования 3 DES, алгоритм аутентификации SHA1. Данные параметры безопасности используются для защиты IKE-соединения:

```
rtt(config)# security ike proposal IKEPROP
rtt(config-ike-proposal)# dh-group 2
rtt(config-ike-proposal)# authentication algorithm sha1
rtt(config-ike-proposal)# encryption algorithm 3des
rtt(config-ike-proposal)# exit
```

Создадим политику протокола IKE. В политике указывается список профилей протокола IKE, по которым могут согласовываться узлы, ключ аутентификации, метод аутентификации XAUTH по ключу и режим аутентификации – клиент:

```
rtt(config)# security ike policy IKEPOLICY
rtt(config-ike-policy)# pre-shared-key hexadecimal 123FFF
rtt(config-ike-policy)# authentication method xauth-psk-key
rtt(config-ike-policy)# authentication mode client
rtt(config-ike-policy)# proposal IKEPROP
rtt(config-ike-policy)# exit
```

Создадим профиль доступа и заведем в нем пару логин и пароль:

```
rtt(config)# access profile XAUTH
rtt(config-access-profile)# user client1
rtt(config-profile)# password ascii-text password123
rtt(config-profile)# exit
rtt(config-access-profile)# exit
```

Создадим интерфейс loopback для терминции IP-адреса, полученного от IPsec VPN сервера:

```
rtt(config)# interface loopback 8
rtt(config-loopback)# exit
```

Создадим шлюз протокола IKE. В данном профиле указывается политика, интерфейс терминции, режим динамического установления удаленной подсети, выбор профиля доступа для XAUTH и режим перенаправления трафика в туннель по политике:

```
rtt(config)# security ike gateway IKEGW
rtt(config-ike-gw)# ike-policy IKEPOLICY
rtt(config-ike-gw)# assign-interface loopback 8
rtt(config-ike-gw)# local address 120.11.5.1
rtt(config-ike-gw)# remote address 180.100.0.1
rtt(config-ike-gw)# remote network dynamic client
rtt(config-ike-gw)# mode policy-based
rtt(config-ike-gw)# access-profile xauth client client1
rtt(config-ike-gw)# exit
```

Создадим профиль параметров безопасности для IPsec-туннеля. В профиле укажем алгоритм шифрования 3DES, алгоритм аутентификации SHA1. Данные параметры безопасности используются для защиты IPsec-туннеля:

```
rtt(config)# security ipsec proposal IPSECPROP
```

```
rtt(config-ipsec-proposal)# authentication algorithm md5
rtt(config-ipsec-proposal)# encryption algorithm aes128
rtt(config-ipsec-proposal)# exit
```

Создадим политику для IPsec-туннеля. В политике указывается список профилей IPsec-туннеля, по которым могут согласовываться узлы:

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

Создадим IPsec VPN. В VPN указывается шлюз IKE-протокола, политика IP sec-туннеля, режим обмена ключами и способ установления соединения. После ввода всех параметров включим туннель командой **enable**:

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

Разрешим протокол esp и udp порты 500,4500 в конфигурации firewall для установления IPsec VPN:

```
rtt(config)# security zone-pair untrusted self
rtt(config-zone-pair)# rule 1
rtt(config-zone-pair-rule)# action permit
rtt(config-zone-pair-rule)# match protocol udp
rtt(config-zone-pair-rule)# match destination-port object-group ISAKMP
rtt(config-zone-pair-rule)# enable
rtt(config-zone-pair-rule)# exit
rtt(config-zone-pair)# rule 2
rtt(config-zone-pair-rule)# action permit
rtt(config-zone-pair-rule)# match protocol esp
rtt(config-zone-pair-rule)# enable
rtt(config-zone-pair-rule)# exit
rtt(config-zone-pair)# end
```

Состояние туннеля можно посмотреть командой:

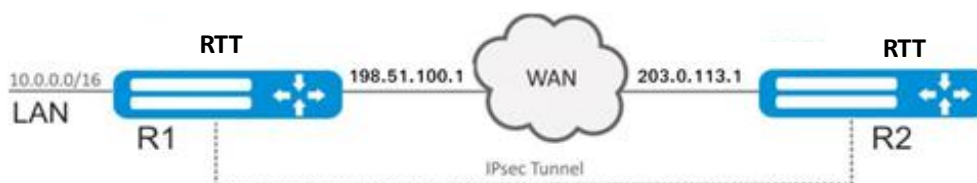
```
rtt# show security ipsec vpn status IPSECVPN
```

Конфигурацию туннеля можно посмотреть командой:

```
rtt# show security ipsec vpn configuration IPSECVPN
```

9.4.8. Пример настройки DPD (Dead Peer Detection)

Задача:



Настроить Dead Peer Detection на R1 для Policy-based IPsec VPN между R1 и R2.

Исходную конфигурацию можно взять из **Пример настройки Policy-based IPsec VPN с аутентификацией по общему известному ключу**.

Решение:

На R1 в шлюзе протокола IKE укажем: режим работы DPD – restart, интервал опроса – 1 секунду, таймаут – 4 секунды:

```

rtt# configure
rtt(config)# security ike gateway ike_gw1
rtt(config-ike-gw)# dead-peer-detection action restart
rtt(config-ike-gw)# dead-peer-detection interval 1
rtt(config-ike-gw)# dead-peer-detection timeout 4
rtt(config-ike-gw)# exit
  
```

Состояние туннеля можно посмотреть командой:

```
rtt# show security ipsec vpn status ipsec1
```

Конфигурацию туннеля можно посмотреть командой:

```
rtt# show security ipsec vpn configuration ipsec1
```

После разрыва соединения между R1 и R2 на R1 IPsec-туннель начнет перестраиваться спустя 4 секунды после разрыва.

```

rtt# show security ipsec vpn status
Name                               Local host      Remote host      Initiator
spi                               Responder spi   State
-----
ipsec1                            198.51.100.1    203.0.113.1
0x7a77a25a55853255 0xb62fd04f2db43d08 Established
2037-10-30T07:52:53+00:00 %CLI-I-CMD: user admin from console input: show
security ipsec vpn status
rtt# show security ipsec vpn status
Name                               Local host      Remote host      Initiator
spi                               Responder spi   State
-----
ipsec1                            198.51.100.1    203.0.113.1
0x77706e37b4e68cce 0x0000000000000000 Connecting
2037-10-30T07:52:57+00:00 %CLI-I-CMD: user admin from console input: show
security ipsec vpn status
  
```

9.5. Настройка LT-туннелей

LT (англ. Logical Tunnel – логический туннель) – тип туннелей, предназначенный для передачи маршрутной информации и трафика между различными виртуальными маршрутизаторами (VRF), сконфигурированными на одном аппаратном маршрутизаторе. LT-туннель может использоваться для организации взаимодействия между двумя или более VRF с применением ограничений firewall.

9.5.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать LT-туннели для каждого из существующих VRF.	<code>rtt(config)# tunnel lt <ID></code>	<ID> – идентификатор туннеля в диапазоне [1..128].
2	Указать описание конфигурируемых туннелей (необязательно).	<code>rtt(config-lt)# description <DESCRIPTION></code>	<DESCRIPTION> – описание туннеля, задаётся строкой до 255 символов.
3	Включить каждый LT-туннель в соответствующий VFR.	<code>rtt(config-lt)# ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.
4	Включить каждый LT-туннель в зону безопасности и настроить правила взаимодействия между зонами или отключить firewall для LT-туннеля.	<code>rtt(config-lt)# security-zone<NAME></code>	<NAME> – имя зоны безопасности, задаётся строкой до 12 символов.
		<code>rtt(config-lt)# ip firewall disable</code>	
5	Для каждого LT-туннеля задать номер противоположный LT туннель (в другом VRF).	<code>rtt(config-lt)# peer lt <ID></code>	<ID> – идентификатор туннеля в диапазоне [1..128].
6	Для каждого LT-туннеля указать IP-адрес для маршрутизации пакетов. Для взаимодействующих LT-туннелей, IP-адреса должны быть из одной IP-подсети.	<code>rtt(config-lt)# ip address <ADDR/LEN> [unit <ID>]</code> или <code>rtt(config-lt)# ip address <ADDR/LEN> secondary [unit <ID>]</code>	<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]; <ID> – номер юнита, принимает значения [1..4]. Ключ secondary указывает, что настроенный адрес является дополнительным IP-адресом. Если это ключевое слово отсутствует, настроенный адрес является основным IP-адресом. Возможно указать до 7 дополнительных IP-адресов.

Шаг	Описание	Команда	Ключи
7	Включить туннели.	<code>rtt(config-lt) # enable</code>	
8	Для каждого VRF настроить необходимые протоколы маршрутизации через LT-туннель.		
9	Задать интервал времени, за который усредняется статистика о нагрузке на туннеле (необязательно).	<code>rtt(config-lt) # load-average <TIME></code>	<TIME> – интервал в секундах, принимает значения [5..150]. Значение по умолчанию: 5.
10	Указать размер MTU (Maximum Transmission Unit) пакетов, которые может пропускать данный bridge (необязательно; возможно, если в bridge включен только VLAN). MTU более 1500 будет активно только в случае применения команды "system jumbo-frames".	<code>rtt(config-lt) # mtu <MTU></code>	<MTU> – значение MTU, принимает значения в диапазоне: [1280..10000]. Значение по умолчанию: 1500.

9.5.2. Пример настройки

Задача:

Организовать взаимодействие между хостами, терминированными в двух VRF vrf_1 и vrf_2.

Исходная конфигурация:

```
hostname rtt
ip vrf vrf_1
exit
ip vrf vrf_2
exit
interface gigabitethernet 1/0/1
  ip vrf forwarding vrf_1
  ip firewall disable
  ip address 10.0.0.1/24
exit
interface gigabitethernet 1/0/2
  ip vrf forwarding vrf_2
  ip firewall disable
  ip address 10.0.1.1/24
```

Решение:

Создадим LT-туннели для каждого VRF с указанием IP-адресов из одной подсети:

```
rtt(config)# tunnel lt 1
rtt(config-lt)# ip vrf forwarding vrf_1
rtt(config-lt)# ip firewall disable
rtt(config-lt)# ip address 192.168.0.1/30
rtt(config-lt)# exit
rtt(config)# tunnel lt 2
rtt(config-lt)# ip vrf forwarding vrf_2
rtt(config-lt)# ip firewall disable
rtt(config-lt)# ip address 192.168.0.2/30
rtt(config-lt)# exit
```

Укажем для каждого LT-туннеля LT-туннель из VRF, с которым необходимо установить связь, и активируем их:

```
rtt(config)# tunnel lt 1
rtt(config-lt)# peer lt 2
rtt(config-lt)# enable
rtt(config-lt)# exit
rtt(config)# tunnel lt 2
rtt(config-lt)# peer lt 1
rtt(config-lt)# enable
rtt(config-lt)# exit
```



Если в VRF не сконфигурирован ни один из протоколов динамической маршрутизации, то необходимо указать статические маршруты для каждого VRF:

```
rtt(config)# ip route vrf vrf_1 0.0.0.0/0 192.168.0.2
rtt(config)# ip route vrf vrf_2 0.0.0.0/0 192.168.0.1
```

10.УПРАВЛЕНИЕ ФУНКЦИЯМИ ВТОРОГО УРОВНЯ (L2)

10.1. Настройка физического интерфейса

10.1.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Переход в режим конфигурирования функционала.	<pre> rtt(config)# interface gigabitethernet rtt(config)# interface tengigabitethernet rtt(config)# interface fourtygigabitethernet rtt(config)# interface twentyfivegigabitethernet rtt(config)# interface port-channel { <ID> <UNIT>/<ID> } </pre>	<p><UNIT> – номер устройства в группе устройств [1..4].</p> <p><ID> – порядковый номер группы агрегации каналов, принимает значения [1..12].</p>
2	Выключить интерфейс.	<pre> rtt(config-if-gi) # shutdown </pre>	
3	Задать описание (необязательно).	<pre> rtt(config-if-gi) # description <text> </pre>	<text> – до 255 символов.
4	Задать MTU (необязательно).	<pre> rtt(config-if-gi) # mtu <count> </pre>	<p><count> – 552–10000.</p> <p>Значение по умолчанию: 1500.</p>

Шаг	Описание	Команда	Ключи
5	Задать скорость и режим работы приемопередатчика (необязательно).	<code>rtt(config-if-gi)# speed <SPEED> <DUPLEX></code>	<p><SPEED> – значение скорости:</p> <ul style="list-style-type: none"> • 10M – значение скорости 10 Мбит/с; • 100M – значение скорости 100 Мбит/с; • 1000M – значение скорости 1000 Мбит/с; • 10G – значение скорости 10 Гбит/с; • auto – автоматический выбор режима (недоступно для 10G-интерфейсов). <p>Значение по умолчанию: auto.</p> <p><DUPLEX> – режим работы приемопередатчика, принимает значения:</p> <ul style="list-style-type: none"> • full-duplex – дуплекс; • half-duplex – полудуплекс.
6	Задать MAC-адрес (необязательно).	<code>rtt(config-if-gi)# mac-address <ADDR></code>	<p><ADDR> – MAC-адрес сетевого моста, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].</p>

На данном этапе настройки нет необходимости в задании настроек firewall.

10.1.2. Пример настройки режима L2

Задача:

Настроить интерфейс gigabitethernet 1/0/1 на прохождение трафика следующим образом:

- Задать MAC-address 68:13:e2:7e:e4:9a;
- Перевести интерфейс в режим Switchport;
- Установить значение MTU=1400;
- Перевести интерфейс в режим работы Full-duplex на скорости 10M.



Решение:

Перейдём в режим конфигурирования интерфейса gigabitethernet 1/0/1 и зададим на нём MAC-address 68:13:e2:7e:e4:9a:

```
rtt# configure
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# mac-address 68:13:e2:7e:e4:9a
```

Переведём интерфейс в режим switchport:

```
rtt(config-if-gi)# mode switchport
```

Установим значение MTU на интерфейсе, равное 1400:

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

Установим на интерфейсе скорость 10М и согласуем режим работы приемопередатчика в полном дуплексе. Выйдем из режима конфигурирования, применим и сохраним настройки:

```
rtt(config-if-gi)# speed 10m full-duplex
rtt(config-if-gi)# end
rtt# commit
rtt# confirm
```

Проверим настроенные параметры на интерфейсе:

```
rtt# show interfaces switch-port status gigabitethernet 1/0/1
Interface          gigabitethernet 1/0/1
  Status:           Up
  Media:             copper
  Speed:             10M
  Duplex:            full
  Flow Control:      Disabled
  MDI Mode:          MDI
```

10.2. Настройка VLAN

VLAN (англ. Virtual Local Area Network) — логическая («виртуальная») локальная сеть, представляет собой группу устройств, которые взаимодействуют между собой на канальном уровне независимо от их физического местонахождения. Работа VLAN основана на использовании дополнительных полей Ethernet-заголовка согласно стандарту 802.1q. По сути, VLAN изолирует широковещательный домен путем ограничения коммутации Ethernet-фреймов только с одинаковым VLAN-ID в Ethernet-заголовке.

10.2.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать VLAN.	<code>rtt(config)# vlan <VID></code>	<p><VID> – идентификатор VLAN, задаётся в диапазоне [2..4094].</p> <p>Также есть возможность создания нескольких vlan (через запятую), диапазона vlan (через дефис или комбинированную запись, содержащую запятые и дефисы).</p>
2	Задать имя vlan (необязательно).	<code>rtt(config-vlan)# name <vlan-name></code>	<vlan-name> – до 255 символов.
3	Отключить отслеживание состояния интерфейсов, на которых разрешена обработка Ethernet-фреймов данного VLAN (необязательно).	<code>rtt(config-vlan)# force-up</code>	
4	Установить режим работы физического интерфейса.	<code>rtt(config-if-gi)# mode switchport</code>	Допустимо для всех моделей.
		<code>rtt(config-if-gi)# mode hybrid</code>	Допустимо только для R800
5	Отключить обработку входящих нетегированных Ethernet-фреймов на основе таблицы коммутации VLAN-а по умолчанию (VLAN-ID – 1) (необязательно).	<code>rtt(config-if-gi)# switchport forbidden default-vlan</code>	
6	Задать режим работы L2-интерфейса.	<code>rtt(config-if-gi)# switchport mode access</code>	<p>Только для R100/200</p> <p>Данный режим является режимом по умолчанию и не отображается в конфигурации.</p>
		<code>rtt(config-if-gi)# switchport mode trunk</code>	Только для R100/200
		<code>rtt(config-if-gi)# switchport mode general</code>	<p>Только для R800</p> <p>Данный режим является режимом по умолчанию и не отображается в конфигурации.</p>

Шаг	Описание	Команда	Ключи
7	Настроить список VLAN на интерфейсе.	<pre>rtt(config-if-gi) # switchport trunk allowed vlan <ACT> <VID></pre>	<p>Для R100/200</p> <p><ACT> – назначаемое действие:</p> <p>add – включение интерфейса во VLAN;</p> <p>remove – исключение интерфейса из VLAN.</p> <p><VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094]. Можно задать диапазоном через «-» или перечислением через «,».</p>
		<pre>rtt(config-if-gi) # switchport general allowed vlan <ACT> <VID> [<TYPE>]</pre>	<p>Для R800</p> <p><ACT> – назначаемое действие:</p> <p>add – включение интерфейса во VLAN;</p> <p>remove – исключение интерфейса из VLAN.</p> <p><VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094]. Можно задать диапазоном</p> <p>через «-» или перечислением через «,»;</p> <p><TYPE> – тип пакета:</p> <p>tagged – интерфейс будет передавать и принимать пакеты в указанных VLAN тегированными;</p> <p>untagged – интерфейс будет передавать пакеты в указанных VLAN нетегированными. VLAN, в которую будут направлены входящие нетегированные пакеты, настраивается командой switchport general pvid.</p>
8	Настроить VLAN в качестве Default VLAN на данном интерфейсе для нетегированного трафика, поступающего на данный порт.	<pre>rtt(config-if-gi) # switchport trunk native- vlan <VID></pre>	<p>Для R100/200</p> <p><VID> – идентификатор VLAN, задаётся в диапазоне [2..4094].</p>

Шаг	Описание	Команда	Ключи
9	Установить идентификатор VLAN-порта (PVID) для входящего нетегированного трафика (необязательно).	<code>rtt(config-if-gi) # switchport general pvid <VID></code>	Для R800 <VID> – идентификационный номер VLAN, задаётся в диапазоне [1...4094].
10	Разрешить на интерфейсе обработку Ethernet-фреймов всех созданных на маршрутизаторе VLAN (необязательно).	<code>rtt(config-if-gi) # switchport trunk allowed vlan auto-all</code>	Только для R100/200
		<code>rtt(config-if-gi) # switchport general allowed vlan auto-all</code>	Только для R800
11	Перевести интерфейс в режим изоляции по группам (необязательно).	<code>rtt(config-if-gi) # switchport protected-port</code>	
12	Добавить интерфейс в группу изоляции (необязательно). Данная команда актуальна, только если порт находится в режиме изоляции по группам.	<code>rtt(config-if-gi) # switchport community</code>	<ID> – идентификатор группы, принимает значение в диапазоне [1...30].
13	Включить функцию Private VLAN на интерфейсе (необязательно).	<code>rtt(config-if-gi) # switchport protected</code>	<IF> – интерфейс, задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора .

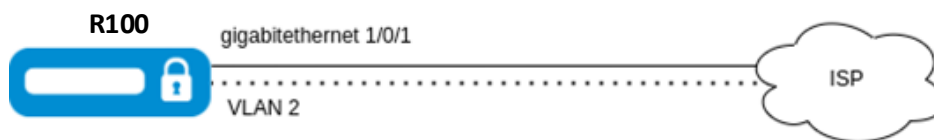
10.2.2. Манипуляции с VLAN на интерфейсе

10.2.3. Пример настройки 1

Данный пример предназначен для использования при конфигурировании VLAN на устройствах R100/200.

Задача:

На основе заводской конфигурации для R100 удалить из VLAN 2 порт gigabitethernet 1/0/1 и назначить его на интерфейс gigabitethernet 1/0/2 в нетегированном режиме.



Решение:

Попадём на интерфейс gigabitethernet 1/0/1 и удалим его из VLAN 2:

```
rtt# configure
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# no switchport access vlan
rtt(config-if-gi)# exit
```

Перейдём на интерфейс gigabitethernet 1/0/2 и назначим его в VLAN 2:

```
rtt(config)# interface gigabitethernet 1/0/2
rtt(config-if-gi)# switchport access vlan 2
```

Применим и сохраним изменения конфигурации:

```
rtt(config-if-gi)# end
rtt# commit
rtt# confirm
```

Проверим, что внесенные изменения вступили в силу, и теперь интерфейс gigabitethernet 1/0/2 принадлежит к VLAN 2 в нетегированном режиме:

```
rtt# show vlans 2
```

VID	Name	Tagged	Untagged
2	--		gi1/0/2, te1/0/1-2

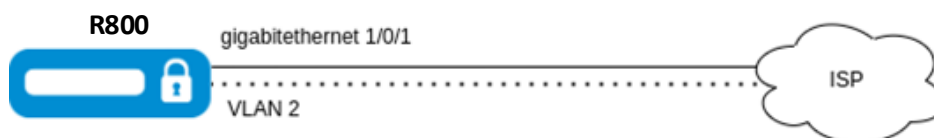
10.2.4. Пример настройки 2



Данный пример предназначен для использования при конфигурировании VLAN на устройствах R800.

Задача:

На основе заводской конфигурации для R800 удалить из VLAN 2 порт gigabitethernet 1/0/1 и назначить его на интерфейс gigabitethernet 1/0/2 в нетегированном режиме.



Решение:

Попадём на интерфейс `gigabitethernet 1/0/1` и удалим его из VLAN 2:

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

Перейдём на интерфейс `gigabitethernet 1/0/2` и назначим его в VLAN 2:

```
rtt(config)# interface gigabitethernet 1/0/2
rtt(config-if-gi)# switchport general allowed vlan add 2 untagged
rtt(config-if-gi)# switchport general pvid 2
rtt(config-if-gi)# end
```

Применим и сохраним изменения конфигурации:

```
rtt(config-if-gi)# end
rtt# commit
rtt# confirm
```

Проверим, что внесенные изменения вступили в силу, и теперь интерфейс `gigabitethernet 1/0/2` принадлежит к VLAN 2 в нетегированном режиме:

```
rtt# show vlans 2
```

VID	Name	Tagged	Untagged
2	--		gil/0/2, tel/0/1-2

10.2.5. Разрешение обработки VLAN в тегированном и нетегированном режимах

10.2.6. Пример настройки 1



Данный пример предназначен для использования при конфигурировании VLAN на устройствах R100/200.

Задача:

На R100 настроить интерфейс `gigabitethernet 1/0/1` на режим `trunk` для передачи и приема фреймов в VLAN 2, VLAN 64, VLAN 2000 в тегированном режиме в сторону устройства Switch. Интерфейс `gigabitethernet 1/0/2` настроить на режим `access` для определения нетегированного трафика от PC в VLAN 2.



Решение:

Перейдём в глобальный режим конфигурирования устройства и создадим VLAN 2, VLAN 64, VLAN 2000:

```
rtt(config)# vlan 2,64,2000
rtt(config-vlan)# exit
```

Перейдём в режим настройки интерфейса gigabitethernet 1/0/1, запретим использование default-vlan (VLAN 1) и настроим его принадлежность к VLAN 2, VLAN 64, VLAN 2000 в тегированном режиме:

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

Перейдём в режим настройки интерфейса gigabitethernet 1/0/2 и определим его к VLAN 2 в нетегированном режиме:

```
rtt(config)# interface gigabitethernet 1/0/2
rtt(config-if-gi)# mode switchport
rtt(config-if-gi)# switchport access vlan 2
```

Применим и сохраним изменения конфигурации:

```
rtt(config-if-gi)# end
rtt# commit
rtt# confirm
```

Проверим, что внесенные изменения вступили в силу, и теперь интерфейс gigabitethernet 1/0/1 принадлежит к VLAN 2, 64, 2000, настроенным в тегированном режиме, а интерфейс gigabitethernet 1/0/2 принадлежит к VLAN 2 в нетегированном режиме:

```
rtt# show vlans
```

VID	Name	Tagged	Untagged
2	--	gil/0/1	gil/0/2
64	--	gil/0/1	
2000	--	gil/0/1	

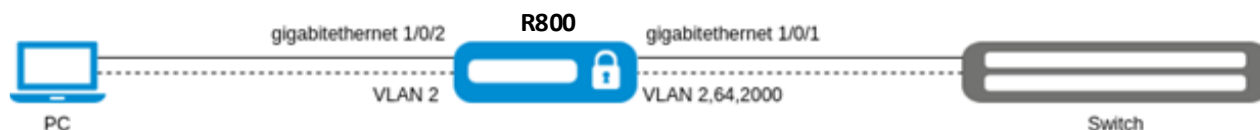
10.2.7. Пример настройки 2



Данный пример предназначен для использования при конфигурировании VLAN на устройствах R800.

Задача:

На R800 настроить интерфейс gigabitethernet 1/0/1 на режим trunk для передачи и приема фреймов в VLAN 2, VLAN 64, VLAN 2000 в тегированном режиме в сторону устройства Switch. Интерфейс gigabitethernet 1/0/2 настроить на режим access для определения нетегированного трафика от PC в VLAN 2.



Решение:

Перейдём в глобальный режим конфигурирования устройства и создадим VLAN 2, VLAN 64, VLAN 2000:

```

rtt(config)# vlan 2,64,2000
rtt(config-vlan)# exit

```

Перейдём в режим настройки интерфейса gigabitethernet 1/0/1, запретим использование default-vlan (VLAN 1) и настроим его принадлежность к VLAN 2, VLAN 64, VLAN 2000 в тегированном режиме:

```

rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# mode switchport
rtt(config-if-gi)# switchport forbidden default-vlan
rtt(config-if-gi)# switchport mode general
rtt(config-if-gi)# switchport general allowed vlan add 2,64,2000 tagged
rtt(config-if-gi)# exit

```

Перейдём в режим настройки интерфейса gigabitethernet 1/0/2, запретим использование default-vlan (VLAN 1), укажем Port-VLAN ID (PVID) 2 и определим его к VLAN 2 в нетегированном режиме:

```

rtt(config)# interface gigabitethernet 1/0/2
rtt(config-if-gi)# mode switchport
rtt(config-if-gi)# switchport forbidden default-vlan
rtt(config-if-gi)# switchport mode general
rtt(config-if-gi)# switchport general allowed vlan add 2 untagged
rtt(config-if-gi)# switchport general pvid 2

```

Применим и сохраним изменения конфигурации:

```

rtt(config-if-gi)# end
rtt# commit
rtt# confirm

```

Проверим, что внесенные изменения вступили в силу, и теперь интерфейс gigabitethernet 1/0/1 принадлежит к VLAN 2, 64, 2000, настроенным в тегированном режиме, а интерфейс gigabitethernet 1/0/2 принадлежит к VLAN 2 в нетегированном режиме:

```

rtt# show vlans

```

VID	Name	Tagged	Untagged
2	--	gil/0/1	gil/0/2
64	--	gil/0/1	

10.2.8. Пример настройки Private Vlan

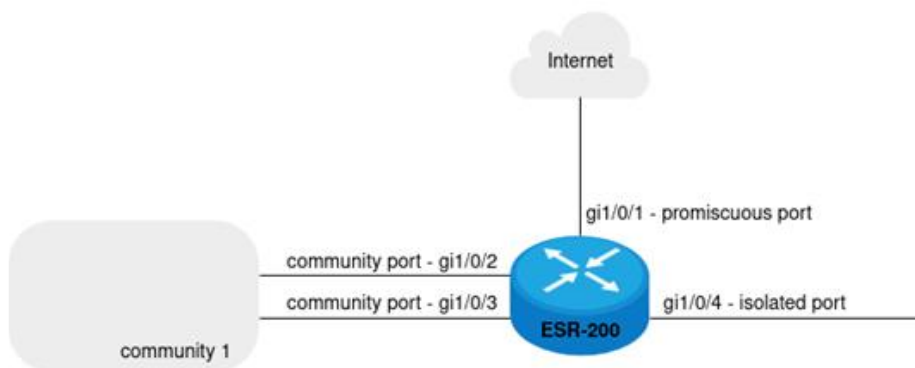
Технология Private VLAN (PVLAN) позволяет производить разграничение трафика на втором уровне модели OSI между портами маршрутизатора, которые находятся в одном широковещательном домене.

На маршрутизаторе может быть сконфигурировано три типа PVLAN-портов:

- promiscuous – это порт, который способен обмениваться данными между любыми интерфейсами, включая isolated- и community-порты PVLAN;
- isolated – это порт, который полностью изолирован от других портов внутри одного и того же PVLAN, но не от promiscuous-портов. PVLAN блокируют весь трафик, идущий в сторону isolated-портов, кроме трафика со стороны promiscuous-портов; пакеты со стороны isolated-портов могут передаваться только в сторону promiscuous-портов;
- community – это группа портов, которые могут обмениваться данными между собой и promiscuous-портами, эти интерфейсы отделены на втором уровне модели OSI от всех остальных community-интерфейсов, а также isolated-портов внутри PVLAN.

Задача:

Настроить изоляцию портов в одном широковещательном домене (PVLAN). Порты gi1/0/2 и gi1/0/3 должны относиться к community 1, порт gi1/0/4 должен быть изолирован в сторону promission port. В качестве promission port выступает gi1/0/1.



Решение:

Создайте VLAN 10:

```
rtt(config)# vlan 10
rtt(config-vlan)# exit
```

Настройте интерфейс gi1/0/1 в режиме «promiscuous port»:

```
rtt(config)# interface gigabitethernet 1/0/1
```

```
rtt(config-if-gi)# mode switchport
rtt(config-if-gi)# switchport forbidden default-vlan
rtt(config-if-gi)# switchport mode trunk
rtt(config-if-gi)# switchport trunk allowed vlan add 10
rtt(config-if-gi)# exit
```

Настройте интерфейсы gi1/0/2, gi1/0/3 в режиме «community port»:

```
rtt(config)# interface gigabitethernet 1/0/2
rtt(config-if-gi)# mode switchport
rtt(config-if-gi)# switchport protected-port
rtt(config-if-gi)# switchport community 1
rtt(config-if-gi)# switchport forbidden default-vlan
rtt(config-if-gi)# switchport mode trunk
rtt(config-if-gi)# switchport trunk allowed vlan add 10
rtt(config-if-gi)# exit
rtt(config)# interface gigabitethernet 1/0/3
rtt(config-if-gi)# mode switchport
rtt(config-if-gi)# switchport protected-port
rtt(config-if-gi)# switchport community 1
rtt(config-if-gi)# switchport forbidden default-vlan
rtt(config-if-gi)# switchport mode trunk
rtt(config-if-gi)# switchport trunk allowed vlan add 10
rtt(config-if-gi)# exit
```

Настройте интерфейс gi1/0/4 в режиме «isolated port»:

```
rtt(config)# interface gigabitethernet 1/0/4
rtt(config-if-gi)# mode switchport
rtt(config-if-gi)# switchport protected gigabitethernet 1/0/1
rtt(config-if-gi)# switchport forbidden default-vlan
rtt(config-if-gi)# switchport mode trunk
rtt(config-if-gi)# switchport trunk allowed vlan add 10
rtt(config-if-gi)# exit
```

Информацию о состоянии физических интерфейсов в режиме изоляции по группам можно узнать с помощью команды:

```
rtt# show interfaces protected-ports
```

10.3. Настройка LLDP

Link Layer Discovery Protocol (LLDP) — протокол канального уровня, позволяющий сетевому оборудованию оповещать оборудование, работающее в локальной сети, о своём существовании и передавать ему свои характеристики, а также получать от него аналогичные сведения.

10.3.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Активировать LLDP на маршрутизаторе.	<code>rtt(config)# lldp enable</code>	

Шаг	Описание	Команда	Ключи
2	Включить прием и обработку LLDPDU на физическом интерфейсе.	<code>rtt(config-if-gi) # lldp receive</code>	
3	Включить отправку LLDPDU на физическом интерфейсе.	<code>rtt(config-if-gi) # lldp transmit</code>	
4	Установить период отправки LLDPDU (необязательно).	<code>rtt(config) # lldp timer <SEC></code>	<SEC> – период времени в секундах, принимает значение [1..32768]. Значение по умолчанию: 30.
5	Установить период, в течение которого маршрутизатор хранит информацию, полученную по LLDP (необязательно).	<code>rtt(config) # lldp hold-multiplier <SEC></code>	<SEC> – период времени в секундах, принимает значение [1..10]. Значение по умолчанию: 4.
6	Установить IP-адрес, который будет передаваться в LLDP TLV в качестве management-address (необязательно).	<code>rtt(config) # lldp management-address <ADDR></code>	<ADDR> – IP-адрес, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. По умолчанию задается один из существующих.
7	Установить поле system-description, которое будет передаваться в LLDP TLV в качестве system-description (необязательно).	<code>rtt(config) # lldp system-description <DESCRIPTION></code>	<DESCRIPTION> – описание системы, задаётся строкой до 255 символов. По умолчанию содержит информацию о модели и версии ПО маршрутизатора.
8	Установить поле system-name, которое будет передаваться в LLDP TLV в качестве system-name (необязательно).	<code>rtt(config) # lldp system-name <NAME></code>	<NAME> – имя системы, задается строкой до 255 символов. По умолчанию совпадает с заданным hostname.

10.3.2. Пример настройки

Задача:

Организовать обмен и обработку LLDPDU между маршрутизаторами RTT-1 и RTT-2.



Решение:

1. Конфигурирование R1

Включим LLDP глобально на маршрутизаторе:

```
rtt(config)# lldp enable
```

Включим прием и отправку LLDPDU на интерфейсе gi 1/0/1:

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

2. Конфигурирование R2

Включим LLDP глобально на маршрутизаторе:

```
rtt(config)# lldp enable
```

Включим прием и отправку LLDPDU на интерфейсе gi 1/0/1:

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

Общую информацию по LLDP соседям можно посмотреть командой:

```
rtt# show lldp neighbors
```

Подробную информацию по соседу конкретного интерфейса можно посмотреть командой:

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

Общую статистику по LLDP можно посмотреть командой:

```
rtt# show lldp statistics
```

10.4. Настройка LLDP MED

LLDP MED – расширение стандарта LLDP, которое позволяет передавать сетевые политики: VLAN ID, DSCP, priority.

10.4.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Активировать LLDP на маршрутизаторе.	<code>rtt(config)# lldp enable</code>	
2	Включить отправку LLDPDU на физическом интерфейсе.	<code>rtt(config-if-gi)# lldp transmit</code>	

Шаг	Описание	Команда	Ключи
3	Активировать расширение MED LLDP на маршрутизаторе.	<code>rtt(config)# lldp med fast-start enable</code>	
4	Создать сетевую политику.	<code>rtt(config)# network-policy <NAME></code>	<NAME> – имя network-policy, задается строкой до 31 символа.
5	Указать тип приложения.	<code>rtt(config-net-policy)# application <APP_TYPE></code>	<p><APP-TYPE> – тип приложения, для которого будет срабатывать network-policy.</p> <p>Принимает значения:</p> <ul style="list-style-type: none"> • voice; • voice-signaling; • guest-voice; • guest-voice-signaling; • softphone-voice; • video-conferencing; • streaming-video; • video-signaling.
6	Установить значение DSCP (необязательно).	<code>rtt(config-net-policy)# dscp <DSCP></code>	<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].
7	Установить значение COS (необязательно).	<code>rtt(config-net-policy)# priority <PRIORITY></code>	<p><COS> – значение приоритета, принимает значения:</p> <ul style="list-style-type: none"> • best-effort – COS0; • background – COS1; • excellent-effort – COS2; • critical-applications – COS3; • video – COS4; • voice – COS5; • internetwork-control – COS6; • network-control – COS7.
8	Установить значение VLAN ID.	<code>rtt(config-net-policy)# vlan <VID> [tagged]</code>	<p><VID> – идентификационный номер VLAN, принимает значения [1...4094];</p> <ul style="list-style-type: none"> • tagged – ключ, при установке которого абонентское устройство будет отправлять Ethernet-фреймы указанного приложения в тегированном виде.
9	Установить сетевую политику на интерфейс.	<code>rtt(config-if-gi)# lldp network-policy <NAME></code>	<NAME> – имя network-policy, задается строкой до 31 символа.

10.4.2. Пример настройки Voice VLAN

Voice VLAN – VLAN ID, при получении которого IP-телефон переходит в режим trunk с заданным VLAN ID для приема и отправки VoIP-трафика. Передача VLAN ID осуществляется посредством расширения MED протокола LLDP.

Задача:

Необходимо разделить трафик телефонии и данных по разным VLAN, vid 10 для данных и vid 20 для телефонии и настроить отправку Voice VLAN с порта gi 1/0/1 RTT. При этом на IP-телефоне должен поддерживаться и быть включен Voice VLAN.



Решение:

Предварительно необходимо создать VLAN 10 и 20 и настроить интерфейс gi 1/0/1 в режиме trunk:

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

Включим LLDP и поддержку MED в LLDP глобально на маршрутизаторе:

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

Создадим и настроим сетевую политику таким образом, чтобы для приложения voice указывался VLAN ID 20:

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

Настроим LLDP на интерфейсе и установим на него сетевую политику:

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

10.5. Настройка протоколов семейства STP

Spanning Tree Protocol – сетевой протокол, основной задачей которого является приведение сети Ethernet с избыточными соединениями к древовидной топологии, исключающей петли. Сетевые устройства обмениваются служебными сообщениями, используя кадры специального формата (BPDU), и выборочно включают/отключают интерфейсы во избежание кольцевых топологий.

Rapid (быстрый) STP (RSTP) – является усовершенствованием протокола STP, характеризуется меньшим временем сходимости за счет использования механизма предложений и соглашений (proposal/agreement process) и улучшенной логикой отправки служебных BPDU-сообщений.

Ниже представлена сводная таблица по поддержке протоколов семейства xSTP.

Устройство	STP/ RSTP		MSTP
	На порту	на бридже	
R100/200	Да	Да	Нет
R800	Нет	Да	Да

10.5.1. Настройка протоколов STP и RSTP

Для активации протокола необходимо перевести работу интерфейса в L2-режим (mode switchport). По умолчанию используется протокол RSTP со следующими временными параметрами: Hello Time – 2 с, Forward Delay – 15 с, Max Age – 20 с.

10.5.1.1. Алгоритм настройки

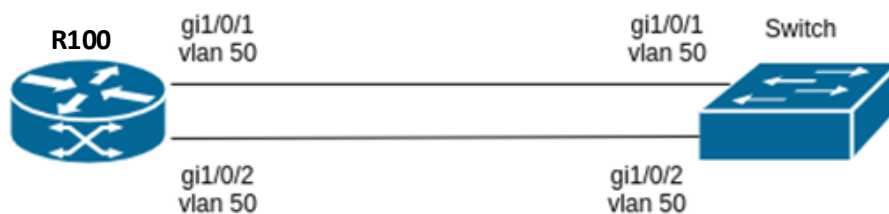
Шаг	Описание	Команда	Ключи
1	Включить STP в глобальном режиме конфигурации.	<code>rtt(config)# spanning-tree</code>	
2	Установить интервал времени, затрачиваемый на прослушивание и изучение состояний перед переключением в состояние передачи (необязательно).	<code>rtt(config)# spanning-tree forward-time <TIME></code>	<TIME> – время в секундах, принимает значения [4..30]. Значение по умолчанию: 15 секунд.
3	Установить интервал времени между отправкой BPDU-пакетов (необязательно).	<code>rtt(config)# spanning-tree hello-time <TIME></code>	<TIME> – время в секундах, принимает значения [1..10]. Значение по умолчанию: 2 секунды.
4	Установить время, в течение которого будет ожидать получения BPDU от ROOT-коммутатора (необязательно).	<code>rtt(config)# spanning-tree max-age <TIME></code>	<TIME> – время в секундах, принимает значения [6..40]. Значение по умолчанию: 20 секунд.

Шаг	Описание	Команда	Ключи
5	Выбрать тип протокола: STP или RSTP.	<code>rtt(config)# spanning-tree mode <MODE></code>	<p><MODE> – протокол семейства STP:</p> <ul style="list-style-type: none"> • STP – IEEE 802.1D Spanning Tree Protocol; • RSTP – IEEE 802.1W Rapid Spanning Tree Protocol; <p>Значение по умолчанию: RSTP.</p>
6	Установить метод расчет стоимости пути (необязательно).	<code>rtt(config)# spanning-tree pathcost method <short long></code>	<p>long – значение ценности в диапазоне [1..200000000];</p> <p>short – значение ценности в диапазоне [1..65535].</p> <p>Значение по умолчанию: short.</p>
7	Настроить приоритет связующего дерева STP (необязательно).	<code>rtt(config)# spanning-tree priority <PRIORITY></code>	<p><PRIORITY> – приоритет, указывается в диапазоне с шагом 4096 [0..61440].</p> <p>Значение по умолчанию: 32768.</p>
8	Перевести интерфейс в L2-режим для включения в работу в STP/RSTP.	<code>rtt(config-if-gi)# mode switchport</code>	
9	Установить стоимость на определенном интерфейсе (необязательно).	<code>rtt(config-if-gi)# spanning-tree cost <COST></code>	<p><COST> – стоимость пути, устанавливается в диапазоне [1..20000000].</p> <p>Значение по умолчанию: 4.</p>
10	Установить тип интерфейса (необязательно).	<code>rtt(config-if-gi)# spanning-tree link-type {point-to-point shared}</code>	<p>point-to-point – команда определяет интерфейс как «точка-точка»;</p> <p>shared – команда определяет интерфейс как «разветвленный».</p> <p>Значение по умолчанию: point-to-point.</p>
11	Установить приоритет интерфейса в связующем дереве STP (необязательно).	<code>rtt(config-if-gi)# spanning-tree port-priority <PRIORITY></code>	<p><PRIORITY> – приоритет, указывается в диапазоне с шагом 16 [0..240].</p>

10.5.1.2. Пример настройки

Задача:

Настроить на маршрутизаторе протокол STP для предотвращения петли с интервалом прослушивания и изучения сети 10 секунд и временем жизни связующего дерева 15 секунд.



Решение:

Для примера разберём схему с маршрутизатором и коммутатором, соединённых двумя линками.

По умолчанию на RTT включен протокол RSTP.

Перейдём в режим конфигурирования:

```
rtt# configure
```

Зададим протокол по умолчанию STP:

```
rtt(config)# spanning-tree mode stp
```

Установим время жизни связующего дерева – 15 секунд и интервал прослушивания и изучения сети – 10 секунд:

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

Вывод команды **show spanning-tree bridge global active**:

```
rtt# show spanning-tree bridge global active
```

Protocol version: STP

Root ID: [32768] a8:f9:4b:ad:5a:00

Root port: [128] gi1/0/1

Pathcost 32768

Message Age 300

Hello time: 2 Max age time: 20 Forward delay: 15

Bridge ID: [32768] a8:f9:4b:ad:8e:5d

Hello time: 2 Max age time: 15 Forward delay: 10

Transmit hold count: 6 Topology change: 0

Time since topology change: 16 Topology change count: 2

Name	State	Prio.Num	Cost	Status	Role	PortFast	Type
-----	-----	-----	-----	-----	-----	-----	-----
gi1/0/1	en	128.2	32768	FRW	Root	No	STP
gi1/0/2	en	128.3	32768	BLK	Altr	No	STP

10.5.2. Настройка протокола STP и RSTP в рамках bridge

Работа протоколов STP/RSTP также возможна в рамках выделенного bridge-домена, что дает возможность организовать кольцевую отказоустойчивую топологию в сетях с использованием L2-туннелей (сервисы L2TPv3 и Ethernet over GRE) или в сетях, устройства которых не поддерживают работу протоколов семейства xSTP.



Для корректной работы устройства должны поддерживать работу протоколов STP/RSTP в bridge-домене.

10.5.2.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Перейти в режим конфигурации bridge-домена для настройки протокола STP/RSTP.	<code>rtt(config)# bridge <BR-NUM></code>	<BR-NUM> – номер bridge.
2	Установить интервал времени, затрачиваемый на прослушивание и изучение состояний перед переключением в состояние передачи (необязательно).	<code>rtt(config-bridge)# spanning-tree forward-time <TIME></code>	<TIME> – время в секундах, принимает значения [4..30]. Значение по умолчанию: 15 секунд.
3	Установить интервал времени между отправкой BPDU-пакетов (необязательно).	<code>rtt(config-bridge)# spanning-tree hello-time <TIME></code>	<TIME> – время в секундах, принимает значения [1..10]. Значение по умолчанию: 2 секунды.
4	Установить время, в течение которого будет ожидаться получение BPDU от ROOT - коммутатора (необязательно).	<code>rtt(config-bridge)# spanning-tree max-age <TIME></code>	<TIME> – время в секундах, принимает значения [6..40]. Значение по умолчанию: 20 секунд.
5	Выбрать тип протокола: STP или RSTP.	<code>rtt(config-bridge)# spanning-tree mode <MODE></code>	<MODE> – протокол семейства STP: <ul style="list-style-type: none"> STP – IEEE 802.1D Spanning Tree Protocol; RSTP – IEEE 802.1W Rapid Spanning Tree Protocol. Значение по умолчанию: RSTP.

Шаг	Описание	Команда	Ключи
6	Установить метод расчета стоимости пути (необязательно).	<code>rtt(config-bridge)# spanning-tree pathcost method <short long></code>	long – значение ценности в диапазоне [1..200000000]; short – значение ценности в диапазоне [1..65535]. Значение по умолчанию: short.
7	Настроить приоритет бриджа – используется при выборе root бриджа в топологии (необязательно).	<code>rtt(config-bridge)# spanning-tree priority <PRIORITY></code>	<PRIORITY> – приоритет, указывается в диапазоне с шагом 4096 [0..61440]. Значение по умолчанию: 32768.
8	Включить бридж в работу с соответствующим интерфейсом.	<code>rtt(config-if-gi)# bridge-group <BR-NUM></code>	<BR-NUM> – номер bridge.
9	Установить стоимость на определенном интерфейсе (необязательно).	<code>rtt(config-if-gi)# spanning-tree cost</code>	<COST> – стоимость пути, устанавливается в диапазоне [1..200000000]. Значение по умолчанию: 4.
10	Установить тип интерфейса (необязательно).	<code>rtt(config-if-gi)# spanning-tree link-type {point-to-point shared}</code>	point-to-point – команда определяет интерфейс как «точка-точка»; shared – команда определяет интерфейс как «разветвленный». Значение по умолчанию: point-to-point.
11	Установить приоритет интерфейса в связующем дереве STP (необязательно).	<code>rtt(config-if-gi)# spanning-tree port-priority <PRIORITY></code>	<PRIORITY> – приоритет, указывается в диапазоне с шагом 16 [0..240].

Для корректной работы STP/RSTP в рамках bridge-домена интерфейсы необходимо добавить в bridge-домен следующим образом:

```
interface gigabitethernet 1/0/3
mode switchport
bridge-group 10
exit
```

10.5.2.2. Пример настройки

Задача:

С помощью протокола RSTP организовать резервирования сервиса L2VPN, построенного с помощью Ethernet over GRE.



Решение:

Настроим адресацию на RTT1 и RTT2:

```
RTT2(config)# interface gigabitethernet 1/0/1
RTT2(config-if-gi)# ip firewall disable
RTT2(config-if-gi)# ip address 198.51.100.2/30
RTT2(config-if-gi)# exit
RTT2(config)# interface gigabitethernet 1/0/2
RTT2(config-if-gi)# ip firewall disable
RTT2(config-if-gi)# ip address 198.51.100.6/30
RTT2(config-if-gi)# exit
RTT2(config)# do commit
RTT2(config)# do confirm
```

```
RTT1(config)# interface gigabitethernet 1/0/1
RTT1(config-if-gi)# ip firewall disable
RTT1(config-if-gi)# ip address 198.51.100.1/30
RTT1(config-if-gi)# exit
RTT1(config)# interface gigabitethernet 1/0/2
RTT1(config-if-gi)# ip firewall disable
RTT1(config-if-gi)# ip address 198.51.100.5/30
RTT1(config-if-gi)# exit
RTT1(config)# do commit
RTT1(config)# do confirm
```

Настроим Ethernet over GRE. Определим GRE 1 как основной канал, GRE 2 – резервный:

```
RTT1(config)# bridge 10
RTT1(config-bridge)# enable
RTT1(config-bridge)# exit
RTT1(config)# tunnel gre 1
RTT1(config-gre)# mode ethernet
RTT1(config-gre)# bridge-group 10
RTT1(config-gre)# local address 198.51.100.1
RTT1(config-gre)# remote address 198.51.100.2
RTT1(config-gre)# spanning-tree cost 10
RTT1(config-gre)# enable
RTT1(config-gre)# exit
RTT1(config)# tunnel gre 2
RTT1(config-gre)# mode ethernet
RTT1(config-gre)# bridge-group 10
RTT1(config-gre)# local address 198.51.100.5
RTT1(config-gre)# remote address 198.51.100.6
```

```

RTT1(config-gre)# spanning-tree cost 20
RTT1(config-gre)# enable
RTT1(config-gre)# exit
RTT1(config-if-gi)# do commit
RTT1(config-if-gi)# do confirm

RTT2(config)# bridge 10
RTT2(config-bridge)# enable
RTT2(config-bridge)# exit
RTT2(config)# tunnel gre 1
RTT2(config-gre)# mode ethernet
RTT2(config-gre)# bridge-group 10
RTT2(config-gre)# local address 198.51.100.2
RTT2(config-gre)# remote address 198.51.100.1
RTT2(config-gre)# spanning-tree cost 10
RTT2(config-gre)# enable
RTT2(config-gre)# exit
RTT2(config)# tunnel gre 2
RTT2(config-gre)# mode ethernet
RTT2(config-gre)# bridge-group 10
RTT2(config-gre)# local address 198.51.100.6
RTT2(config-gre)# remote address 198.51.100.5
RTT2(config-gre)# spanning-tree cost 20
RTT2(config-gre)# enable
RTT2(config-gre)# exit
RTT2(config-if-gi)# do commit
RTT2(config-if-gi)# do confirm

```

Добавим клиентские интерфейсы в соответствующий bridge-домен:

```

RTT1(config)# interface gigabitethernet 1/0/3
RTT1(config-if-gi)# mode switchport
RTT1(config-if-gi)# spanning-tree portfast
RTT1(config-if-gi)# bridge-group 10
RTT1(config-if-gi)# do commit
RTT1(config-if-gi)# do confirm

RTT2(config)# interface gigabitethernet 1/0/3
RTT2(config-if-gi)# mode switchport
RTT2(config-if-gi)# spanning-tree portfast
RTT2(config-if-gi)# bridge-group 10
RTT2(config-if-gi)# do commit
RTT2(config-if-gi)# do confirm

```

Проверим статус туннелей, чтобы убедиться в сходимости протокола RSTP:

```

RTT1# sh tu status

```

Tunnel	Admin state	Link state	MTU	Local IP	Remote IP	Last change
gre 1	Up	Up	1500	198.51.100.1	198.51.100.2	4 hours, 24 minutes and 58 seconds
gre 2	Up	Up	1500	198.51.100.5	198.51.100.6	4 hours, 23 minutes and 6 seconds

```

RTT1# sh spanning-tree bridge 10 active

```

```
Instance name: bridge 10
Protocol version: RSTP
Root ID: [32768] a8:f9:4b:ad:fe:d1
    Root port: [128] gre 1
    Pathcost 10
    Message Age 300
    Hello time: 2 Max age time: 20 Forward delay: 15
Bridge ID: [32768] e4:5a:d4:01:b7:ec
    Hello time: 2 Max age time: 20 Forward delay: 15
    Transmit hold count: 6 Topology change: 0
    Time since topology change: 1600 Topology change count: 7
```

Name	State	Prio.Num	Cost	Status	Role	PortFast	Type
-----	-----	-----	-----	-----	-----	-----	-----
gil/0/3	en	128.5	32768	FRW	Desg	Yes	RSTP
gre 1	en	128.3	10	FRW	Root	No	RSTP
gre 2	en	128.4	20	BLK	Altr	No	RSTP

Настройка сервиса завершена.

10.5.3. Настройка протокола MSTP

Протокол Multiple STP является современной реализацией RSTP, ключевой особенностью которой является поддержка VLAN. В MSTP каждый порт может работать с разным количеством VLAN, принадлежащих разным экземплярам (STI). Изменение состояния в одном из инстансов не оказывает влияния на состояние других экземпляров, что позволяет продолжать обработку трафика в случае блокировки одного из STI.



Протокол MSTP поддерживается только на R800.

10.5.3.1. Алгоритм настройки

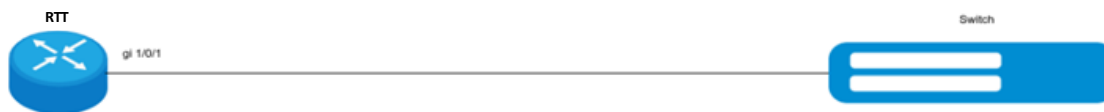
Шаг	Описание	Команда	Ключи
1	Выбрать тип протокола MSTP.	<code>rtt(config)# spanning tree mode <MODE></code>	<p><MODE> – протокол семейства STP:</p> <ul style="list-style-type: none"> • STP – IEEE 802.1D Spanning Tree Protocol; • RSTP – IEEE 802.1W Rapid Spanning Tree Protocol; • MST -IEEE 802.1S Multiple Spanning Tree Protocol. <p>Значение по умолчанию: RSTP.</p>

2	Установить приоритет для соответствующего инстанса перед остальными, использующими общий экземпляр MST (необязательно).	<code>rtt(config)# spanning-tree mst <INSTANCE> priority <PRIORITY></code>	<p><INSTANCE> – номер экземпляра MSTP [1..15].</p> <p><PRIORITY> – приоритет, указывается в диапазоне с шагом 4096 [0..61440].</p> <p>Значение по умолчанию: 32768.</p>
3	Установить максимальное количество транзитных узлов для пакета BPDU, необходимых для формирования дерева (необязательно).	<code>rtt(config)# spanning-tree mst max-hops <NUM></code>	<p><NUM> – количество транзитных узлов, принимает значения [6..40].</p> <p>Значение по умолчанию: 6.</p>
4	Установить принадлежность экземпляра MST к соответствующей группе VLAN (необязательно).	<code>rtt(config-mst)# instance <INSTANCE> vlan <VLAN></code>	<p><INSTANCE> – номер экземпляра MSTP [1..15].</p> <p><VLAN> – номер VLAN.</p>
5	Задать имя конфигурации MST (необязательно).	<code>rtt(config-mst)# name <NAME></code>	<NAME> – имя конфигурации MST [1..31].
6	Задать номер ревизии конфигурации MST (необязательно).	<code>rtt(config-bridge)# revision <REVISION></code>	<REVISION> – номер ревизии конфигурации MST [0..65535].
7	Установить стоимость интерфейса для соответствующего экземпляра MST.	<code>rtt(config-if-gi)# spanning-tree mst <INSTANCE> cost <COST></code>	<p><INSTANCE> – номер экземпляра MSTP [1..15].</p> <p><COST> – стоимость пути, устанавливается в диапазоне [1..20000000].</p> <p>Значение по умолчанию: 4.</p>
8	Установить приоритет порта для соответствующего интерфейса в экземпляре MST.	<code>rtt(config-if-gi)# spanning-tree mst <INSTANCE> port-priority <PRIORITY></code>	<p><INSTANCE> – номер экземпляра MSTP [1..15].</p> <p><PRIORITY> – приоритет, указывается в диапазоне с шагом 16 [0..240].</p>

10.5.3.2. Пример настройки

Задача:

Настроить протокол MSTP между устройствами для VLAN 100 и VLAN 200, выделив для этого соответствующие инстансы.



Решение:

Создадим на устройствах необходимые VLAN. Настроим порт в режиме General, трафик VLAN100, 200 будет отправляться тегированным:

```

rtt(config)# vlan 100
rtt(config-vlan)# exit
rtt(config)# vlan 200
rtt(config-vlan)# exit
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# mo switchport
rtt(config-if-gi)# switchport general allowed vlan add 100,200
  
```

```

Switch(config)#vlan 100,200
Switch(config-vlan-range)#vlan active
Switch(config-vlan-range)#exit
Switch(config)#interface gigabitethernet 0/1
Switch(config-if)#switchport general allowed vlan add 100,200
  
```

Включим протокол MSTP. Настроим инстансы и соответствующие им VLAN.

```

rtt(config)# spanning-tree mst configuration
rtt(config-mst)# name mst
rtt(config-mst)# revision 1
rtt(config-mst)# instance 1 vlan 100
rtt(config-mst)# instance 2 vlan 200
rtt(config-mst)# exit
  
```

```

Switch(config)#spanning-tree mode mst
Switch(config)#spanning-tree mst configuration
Switch(config-mst)# name mst
Switch(config-mst)# revision 1
Switch(config-mst)# instance 1 vlan 100
Switch(config-mst)# instance 2 vlan 200
  
```

Проверим работу протокола MSTP и состояние портов:

```

rtt# sh spanning-tree bridge global
##### MST instance 1. Vlans mapped: 100
      Regional Root ID: [32768] A8:F9:4B:AA:39:7B
      This switch is the Regional Root
                Time since topology change: 0 Topology change: 0
                Topology change count: 32 Max hops: 20
      Designated bridge ID: [32768] A8:F9:4B:AA:39:7B

##### MST instance 2. Vlans mapped: 200
      Regional Root ID: [8192] A8:F9:4B:AA:39:7B
      This switch is the Regional Root
  
```

```

Time since topology change: 0 Topology change: 1
Topology change count: 34 Max hops: 20
Designated bridge ID: [8192] A8:F9:4B:AA:39:7B

```

```
rtt# sh spanning-tree gigabitethernet 1/0/1
```

```
##### MST 1. Mapped Vlans: 100
```

```
Regional Root ID: [16384] A8:F9:4B:AA:39:7B
```

```
This switch is the Regional Root
```

Name	State	Prio.Num	Cost	Status	Role	PortFast	Type
gil/0/1	en	128.2305	4	FRW	Desg	No	P2P Inter

```
##### MST 2. Mapped Vlans: 200
```

```
Regional Root ID: [8192] A8:F9:4B:AA:39:7B
```

```
This switch is the Regional Root
```

Name	State	Prio.Num	Cost	Status	Role	PortFast	Type
gil/0/1	en	128.2305	4	FRW	Desg	No	P2P Inter

Настройка протокола MSTP завершена.

10.5.4. Настройка BPDU Guard

BPDU Guard — функция, позволяющая блокировать порт при поступлении BPDU. Это предотвращает случайное или злонамеренное создание петель в сети. Рекомендуется включать BPDU Guard на портах, к которым подключены конечные устройства. Функция может быть использована на физических, sub, q-in-q интерфейсах, а также на port-channel.



Функция недоступна на устройстве R800.

При обнаружении входящего BPDU порт переводится в статус errdisable. В логе появятся соответствующие сообщения:

```

2025-07-01T10:45:48+00:00 %MSTPD-W-BPDUGUARD: Received BPDU on port
gigabitethernet 1/0/3 with BPDU Guard - disabling port
2025-07-01T10:45:48+00:00 %LINK-W-ERRDISABLE: gigabitethernet 1/0/3 changed
state to ErrDisable, cause 'bpduguard'

```

По умолчанию порт не будет пытаться автоматически переходить в состояние "Up". Это можно сделать командой:

```
rtt# set interface active gigabitethernet 1/0/3
```

Возможна настройка автоматического восстановления. По умолчанию временной интервал составляет 300 секунд (значение меняется от 30 до 86400 секунд):

```

rtt(config)# errdisable recovery cause bpduguard
rtt(config)# errdisable recovery interval
30-86400 Specify the timeout interval in seconds

```

При попытке восстановления порта в логе появится сообщение:

```
rtt# 2025-07-02T03:25:25+00:00 %LINK-W-ERRDISABLE_RECOVERY: Attempting to
recover gigabitethernet 1/0/3 from ErrDisable state caused by 'bpduguard'
```

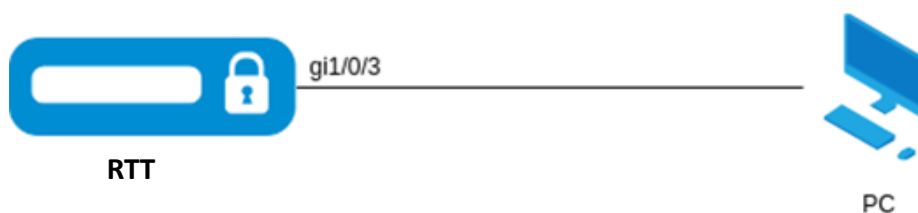
10.5.4.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	В контексте настройки интерфейса/туннеля активировать функцию BPDU Guard	<code>rtt(config-if-gi)# spanning-tree bpduguard</code>	
2	Включить автоматический перевод порта из состояния errDisable (не обязательно)	<code>rtt(config)# errdisable recovery cause bpduguard</code>	
3	Установить временной интервал, по истечении которого интерфейс будет переведен в состояние "UP"	<code>rtt(config)# errdisable recovery interval <TIME></code>	<TIME> - время в секундах, принимает значение в диапазоне [30...86400]. Значение по умолчанию: 300.

10.5.4.2. Пример настройки

Задача:

Включить BPDU Guard на физическом интерфейсе gigabitethernet 1/0/3, к которому подключено конечное устройство. При поступлении BPDU порт должен отключаться на 60 секунд и автоматически восстанавливаться.



Решение:

Включаем на gigabitethernet 1/0/3 BPDU Guard:

```
rtt(config)# interface gigabitethernet 1/0/3
rtt(config-if-gi)# spanning-tree bpduguard
```

Включаем автоматическое восстановление интерфейса и задаём временной интервал 60 секунд:

```
rtt(config)# errdisable recovery cause bpduguard
rtt(config)# errdisable recovery interval 60
```

Проверим настройки восстановления интерфейса:


```

rtt# show errdisable recovery
Timer interval: 60 seconds
Reason Automatic Recovery
-----
bpduguard Enabled

```

После перехода интерфейса в состояние блокировки, его статус будет отображаться как ErrDisable:

```

esr# show interfaces status
Interface          Admin   Link   MTU   MAC address      Last change      Mode
                  State   State
-----
gi1/0/1            Up      Up      1500   68:13:e2:7e:73:91 00,18:31:16      routerport
gi1/0/2            Up      Down    1500   68:13:e2:7e:73:92 00,18:31:22      routerport
gi1/0/3            ErrDisable Down    1500   68:13:e2:7e:73:93 00,00:00:08      switchport

```

Подробную информацию о блокировке интерфейсов можно посмотреть командой:

```

rtt# show interfaces status errdisable
Interface          Err-Disable Reason      Auto-Recovery Time Left
                  (sec)
-----
gi1/0/3            bpduguard              56

```

10.6. Настройка Bridge

Bridge (мост) — это способ соединения двух и более сегментов Ethernet на канальном уровне без использования протоколов более высокого уровня, таких как IP. Пакеты передаются на основе Ethernet-адресов, а не IP-адресов. Поскольку передача выполняется на канальном уровне (уровень 2 модели OSI), трафик протоколов более высокого уровня прозрачно проходит через мост.

10.6.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Добавить сетевой мост (bridge) в систему и перейти в режим настройки его параметров.	<code>rtt(config)# bridge <BRIDGE-ID></code>	<p><BRIDGE-ID> – идентификационный номер моста, принимает значения в диапазоне:</p> <ul style="list-style-type: none"> • для R100/200 – [1..250]; • для R800 – [1..500].
2	Активировать сетевой мост.	<code>rtt(config-bridge)# enable</code>	
3	Указать экземпляр VRF, в котором будет работать данный интерфейс (необязательно).	<code>rtt(config-bridge)# ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.

Шаг	Описание	Команда	Ключи
4	Назначить описание конфигурируемому сетевому мосту (необязательно).	<code>rtt(config-bridge) # description <DESCRIPTION></code>	<DESCRIPTION> – описание сетевого моста, задаётся строкой до 255 символов.
5	Связать саб-интерфейс, qinq-интерфейс, L2GRE-туннель или L2TPv3-туннель с сетевым мостом. Связанные интерфейсы/туннели и сетевые мосты автоматически становятся участниками общего L2-домена (необязательно).	<code>rtt(config-if-gi) # bridge-group <BRIDGE-ID></code> <code>rtt(config-if-l2tpv3) # bridge-group <BRIDGE-ID></code>	<BRIDGE-ID> – идентификационный номер моста, принимает значения в диапазоне: <ul style="list-style-type: none"> • для R100/200 – [1..250]; • для R800 – [1..500].
6	Связать текущий сетевой мост с VLAN. Все интерфейсы и L2-туннели, являющиеся членами назначаемого VLAN, автоматически включаются в сетевой мост и становятся участниками общего L2-домена (необязательно).	<code>rtt(config-bridge) # vlan <VID></code>	<VID> – идентификатор VLAN, задаётся в диапазоне [1..4094].
7	Указать размер MTU (Maximum Transmition Unit) пакетов, которые может пропускать данный bridge (необязательно; возможно, если в bridge включен только VLAN). MTU более 1500 будет активно только в случае применения команды "system jumbo-frames".	<code>rtt(config-bridge) # mtu <MTU></code>	<MTU> – значение MTU, принимает значения в диапазоне: [552..10000]. Значение по умолчанию: 1500.

Шаг	Описание	Команда	Ключи
8	Указать IPv4/IPv6-адрес и маску подсети для конфигурируемого интерфейса или включить получение IP-адреса динамически.	<pre>rtt(config-bridge)# ip address <ADDR/LEN> [unit <ID>]</pre> <p>или</p> <pre>rtt(config-bridge)# ip address <ADDR/LEN> secondary [unit <ID>]</pre>	<p><ADDR/LEN> – IP-адрес и длина маски подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].</p> <p><ID> – номер юнита, принимает значения [1..4].</p> <p>Ключ secondary указывает, что настроенный адрес является дополнительным IP-адресом. Если это ключевое слово отсутствует, настроенный адрес является основным IP-адресом. Возможно указать до 7 дополнительных IP-адресов.</p> <p>Дополнительные функции IPv4-адресации см. в документе «Справочник команд CLI».</p>
		<pre>rtt(config-bridge)# ipv6 address <IPv6- ADDR/LEN> [unit <ID>]</pre>	<p><IPv6-ADDR/LEN> – IP-адрес и префикс подсети, 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].</p> <p><ID> – номер юнита, принимает значения [1..4].</p> <p>Дополнительные функции IPv6-адресации см. в документе «Справочник команд CLI».</p> <p>Можно указать несколько IPv4/IPv6-адресов перечислением через запятую. Может быть назначено до 8 IPv4/IPv6-адресов на интерфейс.</p>
		<pre>rtt(config-bridge)# ip address dhcp</pre>	Дополнительные функции при работе ДНСП-клиента см. в документе «Справочник команд CLI».
9	Отключить на интерфейсе функции Firewall или включить интерфейс в зону безопасности (см. раздел Конфигурирование Firewall).	<pre>rtt(config-bridge)# ip firewall disable</pre>	
		<pre>rtt(config-bridge)# security-zone <NAME></pre>	<NAME> – имя зоны безопасности, задаётся строкой до 31 символа.

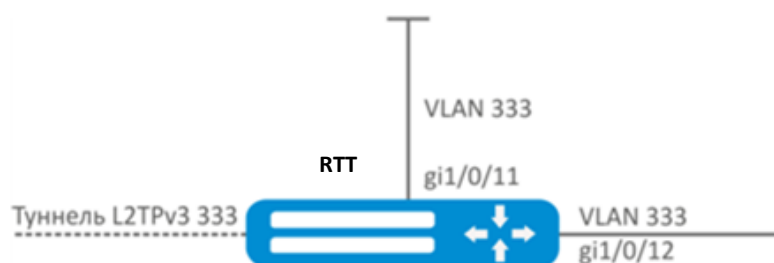
Шаг	Описание	Команда	Ключи
10	Включить запись статистики использования текущего интерфейса (необязательно).	<code>rtt(config-bridge)# history statistics</code>	
11	Задать интервал времени, за который усредняется статистика о нагрузке на bridge (необязательно).	<code>rtt(config-bridge)# load-average <TIME></code>	<TIME> – интервал в секундах, принимает значения [5..150]. Значение по умолчанию: 5.
12	Задать MAC-адрес сетевого моста, отличный от системного (необязательно).	<code>rtt(config-bridge)# mac-address <ADDR></code>	<ADDR> – MAC-адрес сетевого моста, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].
13	<p>Включить на bridge режим изоляции интерфейсов. В данном режиме обмен трафиком между членами сетевого моста запрещен (необязательно).</p> <p>Командой protected-ports exclude {<IF> <TUN> <VLAN>} исключаются из списка изолируемых сущности, включенные в сетевой мост.</p>	<code>rtt(config-bridge)# protected-ports <MODE></code>	<ul style="list-style-type: none"> none – изоляция интерфейсов отключена. В данном режиме коммутация кадров между членами сетевого моста разрешена; local – изоляция интерфейсов включена. В данном режиме коммутация кадров между членами сетевого моста запрещена; radius – изоляция интерфейсов включена. Для использования данного режима требуется настройка Wi-Fi контроллера туннелей в режиме "radius". В данном режиме коммутация кадров между членами сетевого моста запрещена, за исключением SoftGRE DATA туннелей. Для SoftGRE DATA-туннелей параметры изоляции запрашиваются у RADIUS-сервера.
		<code>rtt(config-bridge)# protected-ports exclude {<IF> <TUN> <VLAN>}</code>	<p><IF> – интерфейс, задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.</p> <p><TUN> – имя туннеля, задается в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.</p> <p><VLAN> – vlan, связанный с сетевым мостом.</p>

Шаг	Описание	Команда	Ключи
14	Запретить коммутацию трафика unknown-unicast (когда MAC-адрес назначения не содержится в таблице коммутации) в данном bridge (необязательно; применимо только на R800).	<code>rtt(config-bridge) # unknown-unicast- forwarding disable</code>	
15	Установить время жизни IPv4/IPv6-записей в ARP-таблице, изученных на данном bridge (необязательно).	<code>rtt(config-bridge) # ip arp reachable-time <TIME></code> или <code>rtt(config-bridge) # ipv6 nd reachable-time <TIME></code>	<TIME> – время жизни динамических MAC-адресов, в миллисекундах. Допустимые значения от 5000 до 100000000 миллисекунд. Реальное время обновления записи варьируется от $[0,5;1,5] * \text{<TIME>}$.
<p>Также для bridge-интерфейса возможно настроить:</p> <ul style="list-style-type: none"> • QoS в базовом или расширенном режимах (см. раздел Управление QoS); • проху (см. раздел Проксирование HTTP/HTTPS-трафика); • мониторинг трафика (см. разделы Настройка Netflow и Настройка sFlow); • функционал протоколов маршрутизации (см. раздел Управление маршрутизацией); • протокол VRRF (см. раздел Управление резервированием); • функционал IDS/IPS (см. раздел Настройка IPS/IDS). 			

10.6.2. Пример настройки bridge для VLAN и L2TPv3-туннеля

Задача:

Объединить в единый L2-домен интерфейсы маршрутизатора, относящиеся к локальной сети, и L2TPv3-туннель, проходящий по публичной сети. Для объединения использовать VLAN 333.



Решение:

Создадим VLAN 333:

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

Создадим зону безопасности «trusted»:

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

Добавим интерфейсы gi1/0/11, gi1/0/12 в VLAN 333:

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

Создадим bridge 333, привяжем к нему VLAN 333 и укажем членство в зоне «trusted»:

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

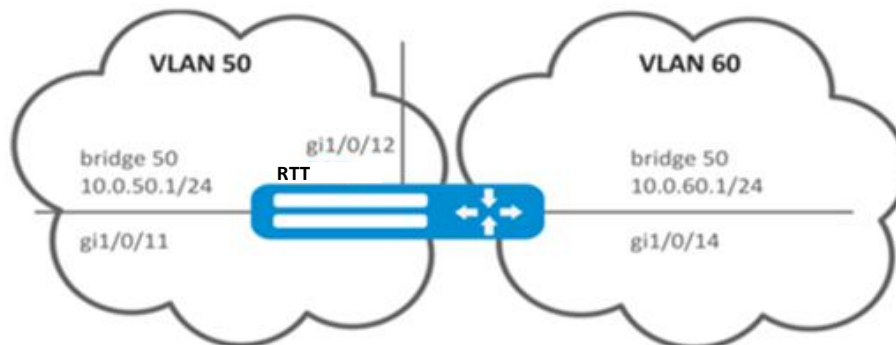
Установим принадлежность L2TPv3-туннеля к мосту, который связан с локальной сетью (настройка L2TPv3-туннеля рассматривается в разделе **Настройка L2TPv3-туннелей**). В общем случае идентификаторы моста и туннеля не должны совпадать с VID, как в данном примере.

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

10.6.3. Пример настройки bridge для VLAN

Задача:

Настроить маршрутизацию между VLAN 50 (10.0.50.0/24) и VLAN 60 (10.0.60.0/24). VLAN 50 должен относиться к зоне «LAN1», VLAN 60 – к зоне «LAN2». Необходимо разрешить свободную передачу трафика между зонами.



Решение:

Создадим VLAN 50, 60:

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

Создадим зоны безопасности «LAN1» и «LAN2»:

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

Назначим интерфейсам gi1/0/11, gi1/0/12 VLAN 50:

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

Назначим интерфейсу gi1/0/14 VLAN 60:

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

Создадим bridge 50, привяжем VLAN 50, укажем IP-адрес 10.0.50.1/24 и членство в зоне «LAN1»:

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

Создадим bridge 60, привяжем VLAN 60, укажем IP-адрес 10.0.60.1/24 и членство в зоне «LAN2»:

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

Создадим правила в Firewall, разрешающие свободное прохождение трафика между зонами:

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

Посмотреть членство интерфейсов в мосте можно командой:

```
rtt# show interfaces bridge
```

10.6.4. Пример настройки добавления/удаления второго VLAN-тега

Задача:

На интерфейс gigabitethernet 1/0/1 поступают Ethernet-кадры с различными VLAN-тегами. Необходимо перенаправить их в интерфейс gigabitethernet 1/0/2, добавив второй VLAN-ID 828. При поступлении на интерфейс gigabitethernet 1/0/2 Ethernet-кадров с VLAN-ID 828 данный тег должен быть удален и отправлен в интерфейс gigabitethernet 1/0/1.

Решение:

Создадим на маршрутизаторе bridge без VLAN и без IP-адреса:

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

Включим интерфейс gigabitethernet 1/0/1 в bridge 1:

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

Включим саб-интерфейс gigabitethernet 1/0/2.828 в bridge 1:

```
rtt(config)# interface gigabitethernet 1/0/2.828
rtt(config-if-sub)# bridge-group 1
rtt(config-if-sub)# exit
```



При добавлении второго VLAN-тега в Ethernet-кадр его размер увеличивается на 4 байта. На интерфейсе маршрутизатора gigabitethernet 1/0/2 и на всем оборудовании, передающем Q-in-Q кадры, необходимо увеличить MTU на 4 байта или более.

10.7. Настройка Dual-Homing



В текущей версии ПО функция поддерживается только на маршрутизаторе R800.

Dual-Homing – технология резервирования соединений, позволяет организовать надежное соединение ключевых ресурсов сети на основе наличия резервных линков.

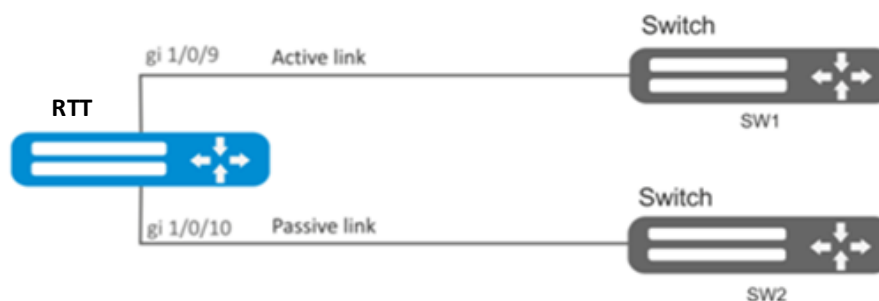
10.7.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Указать резервный интерфейс, на который будет происходить переключение при потере связи на основном.	rtt(config-if-gi)# backup interface<IF> vlan <VID>	<p><IF> – интерфейс, на который будет происходить переключение.</p> <p><VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094].</p> <p>Можно также задать диапазоном через «-» или перечислением через «,».</p>
2	Указать количество копий пакетов с одним и тем же MAC-адресом, которые будут отправлены в активный интерфейс при переключении (необязательно).	rtt(config)# backup-interface mac-duplicate <COUNT>	<COUNT> – количество копий пакетов, принимает значение [1..4].
3	Указать количество пакетов в секунду, которое будет отправлено в активный интерфейс при переключении (необязательно).	rtt(config)# backup-interfacemac-per-second<COUNT>	<COUNT> – количество MAC-адресов в секунду, принимает значение [50..400].
4	Указать, что необходимо осуществить переключение на основной интерфейс при восстановлении связи (необязательно).	rtt(config)# backup-interface preemption	

10.7.2. Пример настройки

Задача:

Организовать резервирование L2-соединений маршрутизатора RTT для VLAN 50-55 через устройства SW1 и SW2.



Решение:

Предварительно нужно выполнить следующие действия:

Создадим VLAN 50-55:

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

Необходимо отключить STP на интерфейсах gigabitethernet 1/0/9 и gigabitethernet 1/0/10, так как совместная работа данных протоколов невозможна:

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

Интерфейсы gigabitethernet 1/0/9 и gigabitethernet 1/0/10 добавим в VLAN 50-55 в режиме general:

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

Основной этап конфигурирования:

Сделаем интерфейс gigabitethernet 1/0/10 резервным для gigabitethernet 1/0/9:

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

Просмотреть информацию о резервных интерфейсах можно командой:

```
rtt# show interfaces backup
```

10.8. Настройка зеркалирования (SPAN/RSPAN)



В текущей версии ПО функция удаленного зеркалирования (RSPAN) поддерживается только на маршрутизаторах R800.

Зеркалирование трафика – функция маршрутизатора, предназначенная для перенаправления трафика с одного порта маршрутизатора на другой порт этого же маршрутизатора (локальное зеркалирование) или на удаленное устройство (удаленное зеркалирование).

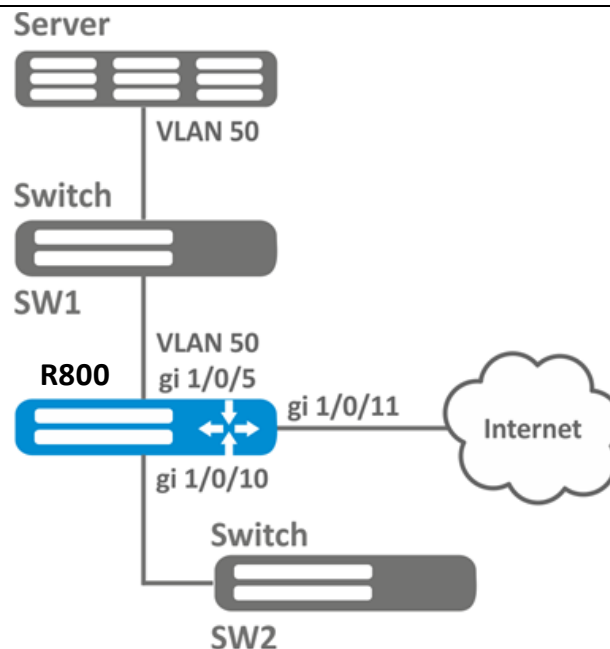
10.8.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Определить VLAN, по которому будет передаваться отзеркалированный трафик (в случае использования удаленного зеркалирования).	<code>rtt(config)# port monitor remote vlan <VID> <DIRECTION></code>	<p><VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094];</p> <p><DIRECTION> – направление трафика:</p> <ul style="list-style-type: none"> tx – зеркалирование в указанный VLAN только исходящего трафика; rx – зеркалирование в указанный VLAN только входящего трафика.
2	Включить режим удаленного зеркалирования (в случае использования удаленного зеркалирования).	<code>rtt(config)# port monitor remote</code>	
3	Определить режим порта, передающего отзеркалированный трафик (необязательно).	<code>rtt(config)# port monitor mode <MODE></code>	<p><MODE> – режим:</p> <ul style="list-style-type: none"> network – совмещенный режим передачи данных и зеркалирование (по умолчанию); monitor-only – только зеркалирование.
4	В режиме конфигурации интерфейса включить зеркалирование.	<code>rtt(config-if-gi)# port monitor interface <IF> [<DIRECTION>]</code>	<p><IF> – интерфейс, с которого будут зеркалироваться кадры;</p> <p><DIRECTION> – направление трафика:</p> <ul style="list-style-type: none"> tx – зеркалирование только исходящего трафика; rx – зеркалирование только входящего трафика.

10.8.2. Пример настройки

Задача:

Организовать удаленное зеркалирование трафика по VLAN 50 с интерфейса gi1/0/11 для передачи на сервер для обработки.



Решение:

Предварительно нужно выполнить следующие действия:

- Создать VLAN 50;
- На интерфейсе gi 1/0/5 добавить VLAN 50 в режиме general.

Основной этап конфигурирования:

Укажем VLAN, по которой будет передаваться зеркалированный трафик:

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

На интерфейсе gi 1/0/5 укажем порт для зеркалирования:

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

Укажем на интерфейсе gi 1/0/5 режим удаленного зеркалирования:

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

10.9. Настройка LACP

LACP — протокол для агрегирования каналов, позволяет объединить несколько физических каналов в один логический. Такое объединение позволяет увеличивать пропускную способность и надежность канала.

10.9.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Установить приоритет системы для протокола LACP.	<code>rtt(config)# lacp system-priority <PRIORITY></code>	<p><PRIORITY> – приоритет, указывается в диапазоне [1..65535].</p> <p>Значение по умолчанию: 1.</p>
2	Установить механизм балансировки нагрузки для групп агрегации каналов.	<code>rtt(config)# port-channel load-balance { src-dst-mac-ip src-dst-mac src-dst-ip src-dst-mac-ip-port }</code>	<ul style="list-style-type: none"> • 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.
3	Установить административный таймаут протокола LACP.	<code>rtt(config)# lacp timeout {short long }</code>	<ul style="list-style-type: none"> • long – длительное время таймаута; • short – короткое время таймаута. <p>Значение по умолчанию: long.</p>
4	Создать и перейти в режим конфигурирования агрегированного интерфейса.	<code>rtt(config)# interface port-channel { <ID> <UNIT>/<ID> }</code>	<p><UNIT> – номер устройства в группе устройств [1..4].</p> <p><CH> – порядковый номер группы агрегации каналов, принимает значения [1..12].</p>
		<code>rtt(config)# interface port-channel { <ID> <UNIT>/<ID> }.<S-VLAN></code>	<p><UNIT> – номер устройства в группе устройств [1..4].</p> <p><CH> – порядковый номер группы агрегации каналов, принимает значения [1..12].</p> <p><S-VLAN> – идентификатор создаваемого S-VLAN.</p>

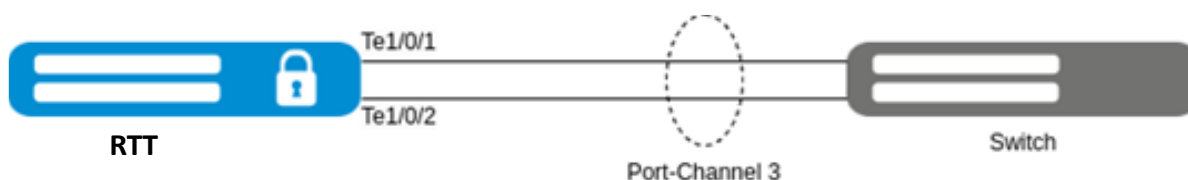
Шаг	Описание	Команда	Ключи
		<code>rtt(config)# interface port-channel { <ID> <UNIT>/<ID> }.<S-VLAN>.<C-VLAN></code>	<C-VLAN> – идентификатор создаваемого C-VLAN.
5	Настроить необходимые параметры агрегированного канала.	<code>rtt(config-if-port-channel)# mode switchport</code>	Установить интерфейс в режим L2
		<code>rtt(config-if-port-channel)# mode routerport</code>	Установить интерфейс в режим L3
6	Задать скорость (необязательно).	<code>rtt(config-if-port-channel)# speed <SPEED></code>	<p><SPEED> – значение скорости:</p> <ul style="list-style-type: none"> • 10M – значение скорости 10 Мбит/с; • 100M – значение скорости 100 Мбит/с; • 1000M – значение скорости 1000 Мбит/с; • 10G – значение скорости 10 Гбит/с. <p>Параметр наследуют все физические интерфейсы, принадлежащие данной группе агрегации каналов.</p> <p>Значение по умолчанию: 1000M</p>
7	Изменить размер MTU (MaximumTransmissionUnit). MTU более 1500 будет активно, только если применена команда system jumbo-frames (необязательно).	<code>rtt(config-if-port-channel)# mtu <MTU></code>	<p><MTU> – значение MTU в байтах.</p> <p>Параметр наследуют все физические интерфейсы, принадлежащие данной группе агрегации каналов.</p> <p>Значение по умолчанию: 1500.</p>
8	Перейти в режим конфигурирования физического интерфейса.	<code>rtt(config)# interface <IF-TYPE><IF-NUM></code>	<p><IF-TYPE> – тип интерфейса (gigabitethernet или tengigabitethernet).</p> <p><IF-NUM> – U/S/P – U-юнит (1), S – слот (0), P – порт.</p>

Шаг	Описание	Команда	Ключи
9	Включить физический интерфейс в группу агрегации каналов с указанием режима формирования группы агрегации каналов.	<code>rtt(config-if-gi)# channel-group <ID> mode <MODE></code>	<p><ID> – порядковый номер группы агрегации каналов, принимает значения [1..12].</p> <p><MODE> – режим формирования группы агрегации каналов:</p> <ul style="list-style-type: none"> auto – добавить интерфейс в динамическую группу агрегации с поддержкой протокола LACP; on – добавить интерфейс в статическую группу агрегации.
10	Установить LACP-приоритет интерфейса Ethernet.	<code>rtt(config-if-gi)# lacp port-priority <PRIORITY></code>	<p><PRIORITY> – приоритет, указывается в диапазоне [1..65535].</p> <p>Значение по умолчанию: 1.</p>
11	Установить интервал времени, в течение которого собирается статистика о нагрузке на интерфейс (необязательно).	<code>rtt(config-if-gi)# load- average <TIME></code>	<TIME> – интервал в секундах, принимает значения [5..150].
12	Включить запись статистики использования текущего интерфейса (необязательно).	<code>rtt(config-if-gi)# history statistics</code>	
<p>Также для агрегированного интерфейса возможно настроить:</p> <ul style="list-style-type: none"> IPv4/IPv6-адресацию (см. в документе «Справочник команд CLI».); Firewall (см. раздел Конфигурирование Firewall); QoS в базовом или расширенном режимах (см. раздел Управление QoS); проxy (см. раздел Проксирование HTTP/HTTPS-трафика); мониторинг трафика (см. разделы Настройка Netflow и Настройка sFlow); функционал протоколов маршрутизации (см. раздел Управление маршрутизацией); протокол VRRF (см. раздел Управление резервированием); функционал IDS/IPS (см. раздел Настройка IPS/IDS). 			

10.9.2. Пример настройки

Задача:

Настроить агрегированный канал между маршрутизатором RTT и коммутатором с помощью tengigabitethernet-интерфейсов со значением MTU 9000 в режиме Trunk для передачи всех VLAN, созданных на маршрутизаторе.



Решение:

Предварительная конфигурация:

Предварительно на устройствах необходимо включить поддержку Jumbo-фреймов. Для вступления изменений в силу требуется перезагрузка устройства:

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

Также на устройствах должны быть созданы необходимые VLAN для передачи в режиме Trunk.

Основной этап конфигурирования:

Создадим интерфейс port-channel 3:

```
rtt(config)# interface port-channel 3
```

Переведём интерфейс в режим switchport:

```
rtt(config-if-port-channel)# mode switchport
```

Зададим размер MTU = 9000:

```
rtt(config-if-port-channel)# mtu 9000
```

Установим значение скорости физических интерфейсов, на которых будет работать агрегированный интерфейс:

```
rtt(config-if-port-channel)# speed 10G
```

Настроим работу интерфейса в режиме trunk с передачей всех VLAN, созданных на маршрутизаторе:

```
rtt(config-if-port-channel)# switchport mode trunk
rtt(config-if-port-channel)# switchport trunk allowed vlan auto-all
```

Перейдём к настройке физических интерфейсов. Переведём интерфейсы tengigabitethernet1/0/1 и tengigabitethernet1/0/2 в режим работы switchport:

```
rtt(config)# interface tengigabitethernet 1/0/1-2
rtt(config-if-te)# mode switchport
```


Включим интерфейсы в созданную группу агрегации каналов с ID 3 в режиме auto с поддержкой протокола LACP:

```
rtt(config-if-te)# channel-group 3 mode auto
```

Дальнейшая конфигурация port-channel проводится как на обычном физическом интерфейсе.

11.УПРАВЛЕНИЕ QoS

QoS (Quality of Service) – технология предоставления различным классам трафика различных приоритетов в обслуживании. Использование службы QoS позволяет сетевым приложениям сосуществовать в одной сети, не уменьшая при этом пропускную способность других приложений.

11.1. Базовый QoS

В базовом режиме на маршрутизаторах RTT классификация (направление трафика в очередь) и перемаркировка работают только на входе (на интерфейсе, через который поступает трафик, должен быть включен QoS).

11.1.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Включить сервис QoS на интерфейсе/туннеле/сетевом мосту. Если на интерфейсе не назначена политика QoS, то интерфейс работает в режиме BasicQoS.	<code>rtt(config-if-gi) # qos enable</code>	
2	Установить режим доверия к значениям кодов 802.1p и DSCP во входящих пакетах (необязательно).	<code>rtt(config) # qos trust <MODE></code>	<p><MODE> – режим доверия к значениям кодов 802.1p и DSCP, принимает одно из следующих значений:</p> <ul style="list-style-type: none"> • dscp – режим доверия значениям кодов DSCP в IP-заголовке. Не IP-пакеты будут направлены в очередь по умолчанию. • cos – режим доверия значениям кодов 802.1p в теге 802.1q. Нетегированные пакеты будут направлены в очередь по умолчанию. • cos - dscp – режим доверия значениям кодов DSCP для IP-пакетов и значениям кодов 802.1p для остальных пакетов.

Шаг	Описание	Команда	Ключи
3	<p>Установить соответствие между значениями кодов DSCP входящих пакетов и исходящими очередями.</p> <p>Данное соответствие работает на входящие пакеты интерфейса/туннеля/моста, на котором включен QoS (необязательно).</p>	<pre>rtt(config)# qos map dscp-queue <DSCP> to <QUEUE></pre>	<p><DSCP> – классификатор обслуживания в IP-заголовке пакета, принимает значения [0..63];</p> <p><QUEUE> – идентификатор очереди, принимает значения [1..8].</p> <p>Значения по умолчанию:</p> <ul style="list-style-type: none"> • 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
4	<p>Установить соответствие между значениями кодов 802.1p входящих пакетов и исходящими очередями.</p> <p>Данное соответствие работает на входящие пакеты интерфейса/туннеля/моста, на котором включен QoS (необязательно).</p>	<pre>rtt(config)# qos map cos-queue <COS> to <QUEUE></pre>	<p><COS> – классификатор обслуживания в теге 802.1p пакета, принимает значения [0..7];</p> <p><QUEUE> – идентификатор очереди, принимает значения [1..8].</p> <p>Значения по умолчанию:</p> <ul style="list-style-type: none"> • 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
5	<p>Установить соответствие между значениями кодов DSCP входящих пакетов и кодов DSCP на выходе из устройства (в случае необходимости перемаркировки).</p> <p>Данное соответствие работает на входящие пакеты интерфейса/туннеля/моста, на котором включен QoS.</p>	<pre>rtt(config)# qos map dscp-queue <DSCP> to <DSCP></pre>	<p><DSCP> – классификатор обслуживания в IP-заголовке пакета, принимает значения [0..63].</p>

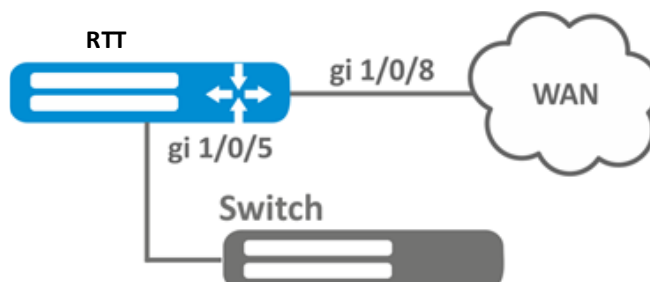
Шаг	Описание	Команда	Ключи
6	Включить изменения кодов DSCP в соответствии с таблицей DSCP-Mutation (в случае необходимости перемаркировки).	<code>rtt(config)# qos dscp mutation</code>	
7	Установить номер очереди по умолчанию, в которую попадает весь трафик, кроме IP, в режиме доверия DSCP-приоритетам.	<code>rtt(config)# qos queue default <QUEUE></code>	<QUEUE> – идентификатор очереди, принимает значения [1..8].
8	Задать количество приоритетных очередей. Оставшиеся очереди являются взвешенными (необязательно).	<code>rtt(config)# priority-queue out num-of-queues <VALUE></code>	<p><VALUE> – количество очередей, принимает значение [0..8], где:</p> <ul style="list-style-type: none"> • 0 – все очереди участвуют в WRR (WRR – механизм обработки очередей на основе веса); • 8 – все очереди обслуживаются как «strictpriority» (strictpriority – приоритетная очередь обслуживается сразу, как только появляются пакеты). <p>Приоритетные очереди выделяются, начиная с 8, в сторону уменьшения номера очереди.</p> <p>Значение по умолчанию: 8.</p>
9	Определить вес для соответствующих взвешенных очередей.	<code>rtt(config)# qos wrr-queue <QUEUE> bandwidth <WEIGHT></code>	<p><QUEUE> – идентификатор очереди, принимает значение [1..8];</p> <p><WEIGHT> – значение веса, принимает значение [1..255].</p> <p>Значение по умолчанию: вес 1 для всех очередей.</p>

Шаг	Описание	Команда	Ключи
10	<p>Установить ограничение скорости исходящего трафика для определенной очереди или интерфейса в целом.</p> <p>Команда актуальна только для BasicQoS-режима интерфейса.</p> <p>Если трафик на входе был классифицирован при помощи расширенного QoS, ограничение не сработает (в случае необходимости ограничения скорости входящего потока).</p>	<pre> rtt(config-if-gi) # traffic-shape { <BANDWIDTH> [BURST] queue <QUEUE><BANDWIDTH> [BURST] } </pre>	<p><QUEUE> – идентификатор очереди, принимает значение [1..8];</p> <p><BANDWIDTH> – средняя скорость трафика в Кбит/с, принимает значение [3000..10000000] для TenggigabitEthernet-интерфейсов и [64..1000000] для прочих интерфейсов и туннелей;</p> <p><BURST> – размер сдерживающего порога в Кбайт, принимает значение [4..16000]. По умолчанию 128 Кбайт.</p> <p>Значение по умолчанию: отключено.</p>
11	<p>Установить ограничение скорости входящего трафика (в случае необходимости ограничения скорости исходящего потока).</p>	<pre> rtt(config-if-gi) # rate-limit <BANDWIDTH> [BURST] </pre>	<p><BANDWIDTH> – средняя скорость трафика в Кбит/с, принимает значение [3000..10000000] для TenggigabitEthernet-интерфейсов и [64..1000000] для прочих интерфейсов и туннелей;</p> <p><BURST> – размер сдерживающего порога в Кбайт, принимает значение [4..16000]. По умолчанию 128 Кбайт.</p> <p>Значение по умолчанию: отключено.</p>

11.1.2. Пример настройки

Задача:

Настроить следующие ограничения на интерфейсе gigabitethernet 1/0/8: передавать трафик с DSCP 22 в восьмую приоритетную очередь, трафик с DSCP 14 в седьмую взвешенную очередь, установить ограничение по скорости в 60 Мбит/с для седьмой очереди.



Решение:

Для того чтобы восьмая очередь осталась приоритетной, а очереди с первой по седьмую стали взвешенными, ограничим количество приоритетных очередей до 1:

```
rtt(config)# priority-queue out num-of-queues 1
```

Перенаправим трафик с DSCP 22 в первую приоритетную очередь:

```
rtt(config)# qos map dscp-queue 22 to 8
```

Перенаправим трафик с DSCP 14 в седьмую взвешенную очередь:

```
rtt(config)# qos map dscp-queue 14 to 7
```

Включим QoS на входящем интерфейсе для корректной классификации трафика и направления в соответствующую очередь со стороны LAN:

```
rtt(config)# interface gigabitethernet 1/0/5
rtt(config-if-gi)# qos enable
rtt(config-if-gi)# exit
```

Включим QoS на интерфейсе со стороны WAN для правильной обработки очередей и ограничения полосы пропускания:

```
rtt(config)# interface gigabitethernet 1/0/8
rtt(config-if-gi)# qos enable
```

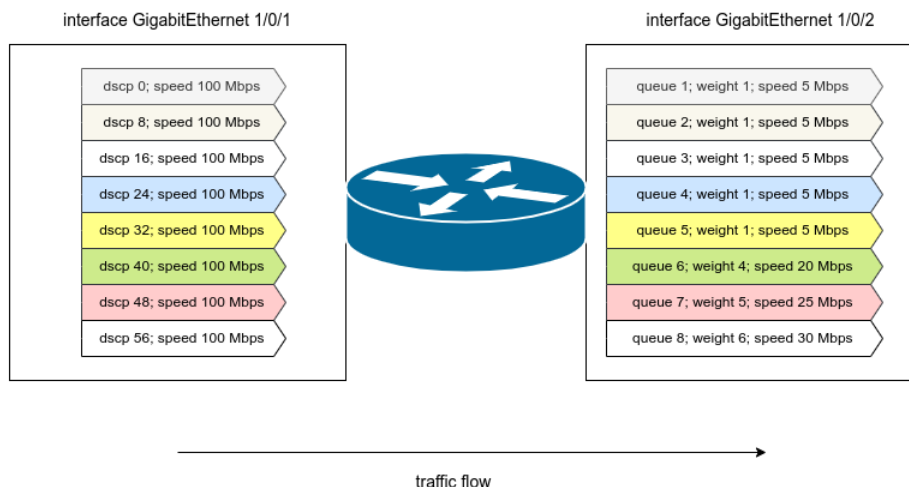
Установим ограничение по скорости в 60 Мбит/с для седьмой очереди:

```
rtt(config-if)# traffic-shape queue 7 60000
rtt(config-if)# exit
```

Просмотреть статистику по QoS можно командой:

```
rtt# show qos statistics gigabitethernet 1/0/8
```

11.1.3. Пример расчета пропускной способности для взвешенных очередей



В рамках данного примера произведем расчет пропускной способности взвешенных очередей. Результаты являются примерными и могут отличаться от практических значений, т. к. не учитывают влияние всплесков.

Пример конфигурации:

Конфигурация RTT

```
hostname RTT

qos wrr-queue 1 bandwidth 1
qos wrr-queue 2 bandwidth 1
qos wrr-queue 3 bandwidth 1
qos wrr-queue 4 bandwidth 1
qos wrr-queue 5 bandwidth 1
qos wrr-queue 6 bandwidth 4
qos wrr-queue 7 bandwidth 5
qos wrr-queue 8 bandwidth 6

interface gigabitethernet 1/0/1
 ip firewall disable
 ip address 10.100.0.1/30
 qos enable
exit

interface gigabitethernet 1/0/2
 ip firewall disable
 ip address 10.101.0.1/30
 traffic-shape 100000 512
 qos enable
exit
```

В приведенном примере настроены взвешенные очереди с соответствующими весами.

- На входящий интерфейс GigabitEthernet 1/0/1 поступают 8 потоков трафика с различными значениями DSCP со скоростью 100 Мбит/с каждый. По умолчанию маршрутизатор доверяет кодам DSCP, и распределение по очередям происходит в соответствии со следующей картой:

RTT#	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						

1. Найти суммарный вес всех очередей: $1+1+1+1+4+5+6 = 20$ (сложить все значения bandwidth из конфигурации);
2. С учетом значения шейпера (100 Мбит/с) найти пропускную способность очереди на единицу веса: $100/20 = 5$ Мбит/с;
3. Вычислить пропускную способность каждой очереди с учетом их весов:

Очередь 1:	1	*	5	=	5 Mbps;
Очередь 2:	1	*	5	=	5 Mbps;
Очередь 3:	1	*	5	=	5 Mbps;
Очередь 4:	1	*	5	=	5 Mbps;
Очередь 5:	1	*	5	=	5 Mbps;
Очередь 6:	4	*	5	=	20 Mbps;
Очередь 7:	5	*	5	=	25 Mbps;
Очередь 8:	6	*	5	=	30 Mbps;

11.2.1. Алгоритм настройки

В расширенном режиме на маршрутизаторах RTT классификация поступающего трафика возможна как на входящем, так и на исходящем интерфейсах.

Шаг	Описание	Команда	Ключи
1	Создать списки доступа для определения трафика, к которому должен быть применен расширенный QoS.		См. раздел Настройка списков доступа (ACL) .
2	Создать класс QoS и перейти в режим настройки параметров класса.	<code>rtt(config)# class-map <NAME></code>	<NAME> – имя создаваемого класса, задается строкой до 31 символа.
3	Задать описание класса QoS (необязательно).	<code>rtt(config-class-map)# description <description></code>	<description> – до 255 символов.
4	Определить трафик, относящийся к конфигурируемому классу по списку контроля доступа (ACL).	<code>rtt(config-class-map)# match access-group <NAME></code>	<NAME> – имя списка контроля доступа, задается строкой до 31 символа.
5	Задать значение для классификации по полю DSCP в заголовке IP-пакета для конфигурируемого класса.	<code>rtt(config-class-map)# match dscp <DSCP></code>	<DSCP> – значение кода DSCP, принимает значения [0..63].
6	Задать значение для классификации по полю EXP в MPLS заголовке для конфигурируемого класса.	<code>rtt(config-class-map)# match mpls experimental topmost <EXP></code>	<EXP> – значение поля EXP, принимает значения [0..7].
7	Определить трафик протокола NHRP в туннеле GRE к конфигурируемому классу.	<code>rtt(config-class-map)# match protocol nhrp</code>	
8	Задать значение кода IP Precedence, которое будет установлено в IP-пакетах, соответствующих конфигурируемому классу (невозможно назначать одновременно с полями DSCP и CoS) (при необходимости перемаркировки).	<code>rtt(config-class-map)# set ip-precedence <IPP></code>	<IPP> – значение кода IP Precedence, принимает значения [0..7].
9	Задать значение 802.1p приоритета, которое будет установлено в пакетах, соответствующих конфигурируемому классу (невозможно назначать одновременно с полями DSCP и IP Precedence) (при необходимости перемаркировки).	<code>rtt(config-class-map)# set cos <COS></code>	<COS> – значение 802.1p приоритета, принимает значения [0..7].

Шаг	Описание	Команда	Ключи
10	Задать значение кода DSCP, которое будет установлено в IP-пакетах, соответствующих конфигурируемому классу (невозможно назначать одновременно с полями IP Precedence и CoS) (при необходимости перемаркировки).	<code>rtt(config-class-map) # set dscp <DSCP></code>	<DSCP> – значение кода DSCP, принимает значения [0..63].
11	Создать политику QoS и осуществить переход в режим настройки параметров политики.	<code>rtt(config) # policy-map <NAME></code> <code>rtt(config-policy-map) #</code>	<NAME> – имя создаваемой политики, задается строкой до 31 символа.
12	Задать описание политики QoS (необязательно).	<code>rtt(config-policy-map) # description <description></code>	<description> – до 255 символов.
13	Установить гарантированную полосу пропускания исходящего трафика для политики в целом.	<code>rtt(config-policy-map) # shape average { <BANDWIDTH> percent <BANDWIDTH_PERCENT> } [BURST]</code>	<p><BANDWIDTH> – гарантированная полоса трафика в Кбит/с, принимает значение [64..10000000];</p> <p><BANDWIDTH_PERCENT> – гарантированная полоса трафика в %, рассчитывается от (в порядке от более приоритетного к менее приоритетному значению):</p> <ul style="list-style-type: none"> значения shape average корневой политики; значения traffic-shape на сетевом интерфейсе, bridge, туннеле; значения speed сетевого интерфейса. <p>Принимает значение [1..100].</p> <p><BURST> – размер сдерживающего порога в Кбайт, принимает значение [128..16000]. По умолчанию 128 Кбайт.</p>

Шаг	Описание	Команда	Ключи
14	Включить работу полисера (при необходимости).	<pre> rtt(config-policy-map)# police <RATE> [burst-conforming <BURST-CONFORM>] [conform-action <CONFORM-ACTION>] [exceed-action <EXCEED-ACTION>] [burst-excess <BURST-EXCEED>] [violate-action <VIOLATE-ACTION>]] </pre>	<p><RATE> – скорость пополнения токенами conform-корзины в Кбит/с;</p> <p><BURST-CONFORM> – размер conform-корзины в байтах;</p> <p><BURST-EXCEED> – размер excess-корзины в байтах;</p> <p><CONFORM-ACTION> – действие, которое необходимо выполнить с пакетом, для которого имеются токены conform-корзины, принимает значения { permit deny set-cos <COS> set-dscp <DSCP> };</p> <p><EXCEED-ACTION> – действие, которое необходимо выполнить с пакетом, если исчерпаны токены conform-корзины, но имеются токены excess-корзины, принимает значения { permit deny set-cos <COS> set-dscp <DSCP>};</p> <p><VIOLATE-ACTION> – действие, которое необходимо выполнить с пакетом, для которого исчерпаны токены excess-корзины, принимает значения { permit deny set-cos <COS> set-dscp <DSCP> };</p> <p><COS> – классификатор обслуживания в теге 802.1q пакета, принимает значения [0..7];</p> <p><DSCP> – значение кода DSCP, принимает значения [0..63].</p>
15	Включить автоматическое распределение полосы пропускания между классами, в которых нет настройки полосы пропускания, включая класс по умолчанию (в случае необходимости).	<pre> rtt(config-policy-map)# shape auto-distribution </pre>	

Шаг	Описание	Команда	Ключи
16	Включить указанный QoS-класс в политику и осуществить переход в режим настройки параметров класса в рамках политики.	<pre> rtt(config-policy-map) # class <NAME> rtt(config-class-policy-map) # </pre>	<NAME> – имя привязываемого класса, задается строкой до 31 символа. При указании значения «class-default» в данный класс попадает трафик, не классифицированный на входе.
17	Включить политику QoS в класс QoS для создания иерархического QoS.	<pre> rtt(config-class-policy-map) # service-policy <NAME> </pre>	<NAME> – имя политики, задается строкой до 31 символа. Вкладываемая политика должна быть уже создана.
18	Установить гарантированную полосу пропускания исходящего трафика для класса в рамках политики (при необходимости).	<pre> rtt(config-class-policy-map) # shape average { <BANDWIDTH> percent <BANDWIDTH_PERCENT> } [BURST] </pre>	<p><BANDWIDTH> – гарантированная полоса трафика в Кбит/с, принимает значение [64..10000000];</p> <p><BANDWIDTH_PERCENT> – гарантированная полоса трафика в %, рассчитывается от (в порядке от более приоритетного к менее приоритетному значению):</p> <ul style="list-style-type: none"> значения shape average корневой политики; значения traffic-shape на сетевом интерфейсе, bridge, туннеле; значения speed сетевого интерфейса. <p>Принимает значение [1..100].</p> <p><BURST> – размер сдерживающего порога в Кбайт, принимает значение [4..16000]. По умолчанию 128 Кбайт.</p>

Шаг	Описание	Команда	Ключи
19	<p>Установить разделяемую полосу пропускания исходящего трафика для определенного класса.</p> <p>Данную полосу класс может занять, если менее приоритетный класс не занял свою гарантированную полосу (при необходимости).</p>	<pre>rtt(config-class-policy-map)# shape peak { <BANDWIDTH> percent <BANDWIDTH_PERCENT> } [BURST]</pre>	<p><BANDWIDTH> – общая для priority class полоса трафика в Кбит/с, конкуренция происходит на основании приоритета класса, принимает значение [64..10000000];</p> <p><BANDWIDTH_PERCENT> – общая для priority class полоса трафика в %, конкуренция происходит на основании приоритета класса, рассчитывается от (в порядке от более приоритетного к менее приоритетному значению):</p> <ul style="list-style-type: none"> значения shape average корневой политики; значения traffic-shape на сетевом интерфейсе, bridge, туннеле; значения speed сетевого интерфейса. <p>Принимает значение [1..100].</p> <p><BURST> – размер сдерживающего порога в Кбайт, принимает значение [4..16000]. По умолчанию 128 Кбайт.</p>

Шаг	Описание	Команда	Ключи
20	Включить работу полисера для определенного класса (при необходимости).	<pre> rtt(config-class-policy-map)# police <RATE> [burst-conforming <BURST-CONFORM>] [conform-action <CONFORM-ACTION>] [exceed-action <EXCEED-ACTION>] [burst-excess <BURST-EXCEED> [violate-action <VIOLATE-ACTION>]] </pre>	<p><RATE> – скорость пополнения токенами conform-корзины Кбит/с;</p> <p><BURST-CONFORM> – размер conform-корзины в байтах;</p> <p><BURST-EXCEED> – размер excess-корзины в байтах;</p> <p><CONFORM-ACTION> – действие, которое необходимо выполнить с пакетом, для которого имеются токены conform-корзины, принимает значения { permit deny set-cos <COS> set-dscp <DSCP> };</p> <p><EXCEED-ACTION> – действие, которое необходимо выполнить с пакетом, если исчерпаны токены conform-корзины, но имеются токены excess-корзины, принимает значения { permit deny set-cos <COS> set-dscp <DSCP>;</p> <p><VIOLATE-ACTION> – действие, которое необходимо выполнить с пакетом, для которого исчерпаны токены excess-корзины, принимает значения { permit deny set-cos <COS> set-dscp <DSCP> };</p> <p><COS> – классификатор обслуживания в теге 802.1q пакета, принимает значения [0..7];</p> <p><DSCP> – значение кода DSCP, принимает значения [0..63].</p>

Шаг	Описание	Команда	Ключи
21	Определить режим работы класса (необязательно).	<code>rtt(config-class-policy-map)# mode <MODE></code>	<p><MODE> – режим класса:</p> <ul style="list-style-type: none"> • 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). <p>Значение по умолчанию: FIFO.</p>
22	Задать приоритет класса в WRR-процессе (при необходимости).	<code>rtt(config-class-policy-map)# priority class <PRIORITY></code>	<p><PRIORITY> – приоритет класса в WRR-процессе, принимает значения [1..8].</p> <p>Классы с наибольшим приоритетом обрабатываются в первую очередь.</p>
23	Перевести класс в режим StrictPriority и задать приоритет класса (при необходимости).	<code>rtt(config-class-policy-map)# priority level <PRIORITY></code>	<p><PRIORITY> – уровень приоритета в StrictPriority-процессе, принимает значения [1..8].</p> <p>Классы с наибольшим приоритетом обрабатываются в первую очередь. Значение по умолчанию: класс работает в режиме WRR, приоритет не задан.</p>
24	Задать значение кода DSCP, которое будет установлено в IP-пакетах, соответствующих конфигурируемому классу (невозможно назначать одновременно с полями IP Precedence и CoS) (при необходимости перемаркировки параметрами класса в рамках политики).	<code>rtt(config-class-map)# match dscp <DSCP></code>	<p><DSCP> – значение кода DSCP, принимает значения [0..63].</p>

Шаг	Описание	Команда	Ключи
25	Задать значение кода IP Precedence, которое будет установлено в IP-пакетах, соответствующих конфигурируемому классу (невозможно назначать одновременно с полями DSCP и CoS) (при необходимости перемаркировки параметрами класса в рамках политики).	<code>rtt(config-class-map)# set ip-precedence <IPP></code>	<IPP> – значение кода IP Precedence, принимает значения [0..7].
26	Задать значение 802.1p приоритета, которое будет установлено в пакетах, соответствующих конфигурируемому классу (невозможно назначать одновременно с полями DSCP и IP Precedence) (при необходимости перемаркировки параметрами класса в рамках политики).	<code>rtt(config-class-map)# set cos <COS></code>	<COS> – значение 802.1p приоритета, принимает значения [0..7].
27	Определить предельное количество виртуальных очередей (необязательно).	<code>rtt(config-class-policy-map)# fair-queue <QUEUE-LIMIT></code>	<QUEUE-LIMIT> – предельное количество виртуальных очередей, принимает значения в диапазоне [16..4096]. Значение по умолчанию: 16.
28	Определить предельное количество пакетов для виртуальной очереди (необязательно).	<code>rtt(config-class-policy-map)# queue-limit <QUEUE-LIMIT></code>	<QUEUE-LIMIT> – предельное количество пакетов в виртуальной очереди, принимает значения в диапазоне [2..4096]. Значение по умолчанию: 127.

Шаг	Описание	Команда	Ключи
29	Определить параметры RED (Random Early Detection) (при необходимости).	<pre> rtt(config-class-policy- map) # random-detect <LIMIT> <MIN> <MAX> <APS> <APS-NUM> <PROBABILITY> </pre>	<p> <LIMIT> – предельный размер очереди в байтах, принимает значения в диапазоне [1..1000000]; <MIN> – минимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000]; <MAX> – максимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000]; <APS> – средний размер пакета в байтах, принимает значение в диапазоне [1..10000000]; <APS-NUM> – количество пакетов среднего размера разрешенных для кратковременного пропуска, принимает значение в диапазоне [0..10000000]; <PROBABILITY> – вероятность отбрасывания пакетов, принимает значения [0..100]. </p> <p>При указании значений должны выполняться следующие правила:</p> <p> <MAX>> 2 * <MIN> <LIMIT>> 3 * <MAX> </p>

Шаг	Описание	Команда	Ключи
30	Определить параметры GRED (Generalized Random Early Detection) (при необходимости).	<pre> rtt(config-class-policy- map)# random-detect queue <QUEUE-NUM> [dscp <DSCP> precedence <IPP>] <LIMIT> <MIN> <MAX> <APS> <APS-NUM> <PROBABILITY> </pre>	<p> <QUEUE-NUM> – номер очереди [1..16]; <DSCP> – классификатор обслуживания в IP-заголовке пакета, принимает значения [0..63]; <IPP> – значение кода IP Precedence, принимает значения [0..7]; <PRECEDENCE> – значение IP Precedence [0..7]; <LIMIT> – предельный размер очереди в байтах, принимает значения в диапазоне [1..1000000]; <MIN> – минимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000]; <MAX> – максимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000]; <APS> – средний размер пакета в байтах, принимает значение в диапазоне [1..10000000]; <APS-NUM> – количество пакетов среднего размера разрешенных для кратковременного пропуска, принимает значение в диапазоне [0..10000000]; <PROBABILITY> – вероятность отбрасывания пакетов, принимает значения [0..100] </p> <p>При указании значений должны выполняться следующие правила:</p> <p> <MAX>> 2 * <MIN> <LIMIT>> 3 * <MAX> </p>

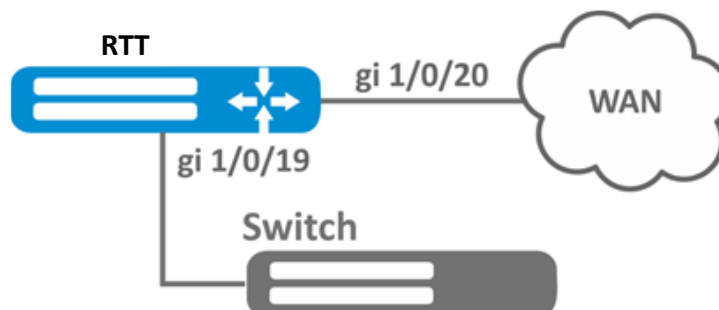
31	<p>Определить очередь по умолчанию для механизма GRED.</p>	<pre>rtt(config-class-policy-map)# random-detect queue default <QUEUE-NUM></pre>	<p>Данной командой определяется очередь по умолчанию для механизма GRED. Для применения данной команды предварительно необходимо настроить команду random-detect queue <QUEUE-NUM> dscp <DSCP>/precedence <0-7> <LIMIT> <MIN> <MAX> <APS> <APS-NUM> <PROBABILITY>. После применения команд random-detect queue <QUEUE-NUM> dscp <DSCP>/precedence <0-7> <LIMIT> <MIN> <MAX> <APS> <APS-NUM> <PROBABILITY> и random-detect queue default <QUEUE-NUM> поля dscp/precedence в заголовке IP-пакетов данной очереди будут игнорироваться.</p> <p><QUEUE-NUM> – номер очереди [1..16];</p> <p><DSCP> – классификатор обслуживания в IP-заголовке пакета, принимает значения [0..63];</p> <p><IPP> – значение кода IP Precedence, принимает значения [0..7];</p> <p><PRECEDENCE> – значение IP Precedence [0..7];</p> <p><LIMIT> – предельный размер очереди в байтах, принимает значения в диапазоне [1..1000000];</p> <p><MIN> – минимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];</p> <p><MAX> – максимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];</p> <p><APS> – средний размер пакета в байтах, принимает значение в диапазоне [1..1000000];</p>
----	--	--	--

Шаг	Описание	Команда	Ключи
			<p><APS-NUM> – количество пакетов среднего размера, разрешенных для кратковременного пропускания,</p> <p>принимает значение в диапазоне [0..10000000];</p> <p><PROBABILITY> – вероятность отбрасывания пакетов, принимает значения [0..100].</p>
32	Включить протокол компрессии tcp-заголовков для трафика отдельного класса (при необходимости).	<code>rtt(config-class-policy-map)# compression header ip tcp</code>	
33	Включить сервис QoS на интерфейсе/туннеле/сетевом мосту.	<code>rtt(config-if-gi)# qos enable</code>	
34	Назначить политику QoS на сконфигурируемом интерфейсе/туннеле/сетевом мосту для классификации входящего (input) или приоритизации исходящего (output) трафика.	<code>rtt(config-if-gi)# service-policy { input output } <NAME></code>	<NAME> – имя QoS-политики, задаётся строкой до 31 символа.

11.2.2. Пример настройки

Задача:

Классифицировать входящий трафик по подсетям (10.0.11.0/24, 10.0.12.0/24), произвести маркировку по DSCP (38 и 42) и разграничение по подсетям (40 Мбит/с и 60 Мбит/с), ограничить общую полосу до 250 Мбит/с, остальной трафик обрабатывать через механизм SFQ.



Решение:

Настроим списки доступа для фильтрации по подсетям, выходим в глобальный режим конфигурации:

```
rtt(config)# ip access-list extended fl1
rtt(config-acl)# rule 1
rtt(config-acl-rule)# action permit
rtt(config-acl-rule)# match protocol any
rtt(config-acl-rule)# match source-address 10.0.11.0 255.255.255.0
rtt(config-acl-rule)# match destination-address any
rtt(config-acl-rule)# enable
rtt(config-acl-rule)# exit
rtt(config-acl)# exit
rtt(config)# ip access-list extended fl2
rtt(config-acl)# rule 1
rtt(config-acl-rule)# action permit
rtt(config-acl-rule)# match protocol any
rtt(config-acl-rule)# match source-address 10.0.12.0 255.255.255.0
rtt(config-acl-rule)# match destination-address any
rtt(config-acl-rule)# enable
rtt(config-acl-rule)# exit
rtt(config-acl)# exit
```

Создаем классы fl1 и fl2, указываем соответствующие списки доступа, настраиваем маркировку:

```
rtt(config)# class-map fl1
rtt(config-class-map)# set dscp 38
rtt(config-class-map)# match access-group fl1
rtt(config-class-map)# exit
rtt(config)# class-map fl2
rtt(config-class-map)# set dscp 42
rtt(config-class-map)# match access-group fl2
rtt(config-class-map)# exit
```

Создаём политику и определяем ограничение общей полосы пропускания:

```
rtt(config)# policy-map fl
rtt(config-policy-map)# shape average 250000
```

Осуществляем привязку класса к политике, настраиваем ограничение полосы пропускания и выходим:

```
rtt(config-policy-map)# class fl1
rtt(config-class-policy-map)# shape average 40000
rtt(config-class-policy-map)# exit
rtt(config-policy-map)# class fl2
rtt(config-class-policy-map)# shape average 60000
rtt(config-class-policy-map)# exit
```

Для настройки ограничения полосы пропускания в процентах необходимо использовать команду **shape average percent**.

Для другого трафика настраиваем класс с режимом SFQ:

```
rtt(config-policy-map)# class class-default
```

```

rtt(config-class-policy-map)# mode sfq
rtt(config-class-policy-map)# fair-queue 800
rtt(config-class-policy-map)# exit
rtt(config-policy-map)# exit

```

Включаем QoS на интерфейсах, политику на входе интерфейса gi 1/0/19 для классификации и на выходе gi1/0/20 для применения ограничений и режима SFQ для класса по умолчанию:

```

rtt(config)# interface gigabitethernet 1/0/19
rtt(config-if-gi)# qos enable
rtt(config-if-gi)# service-policy input fl
rtt(config-if-gi)# exit
rtt(config)# interface gigabitethernet 1/0/20
rtt(config-if-gi)# qos enable
rtt(config-if-gi)# service-policy output fl
rtt(config-if-gi)# exit

```

Для просмотра статистики используется команда:

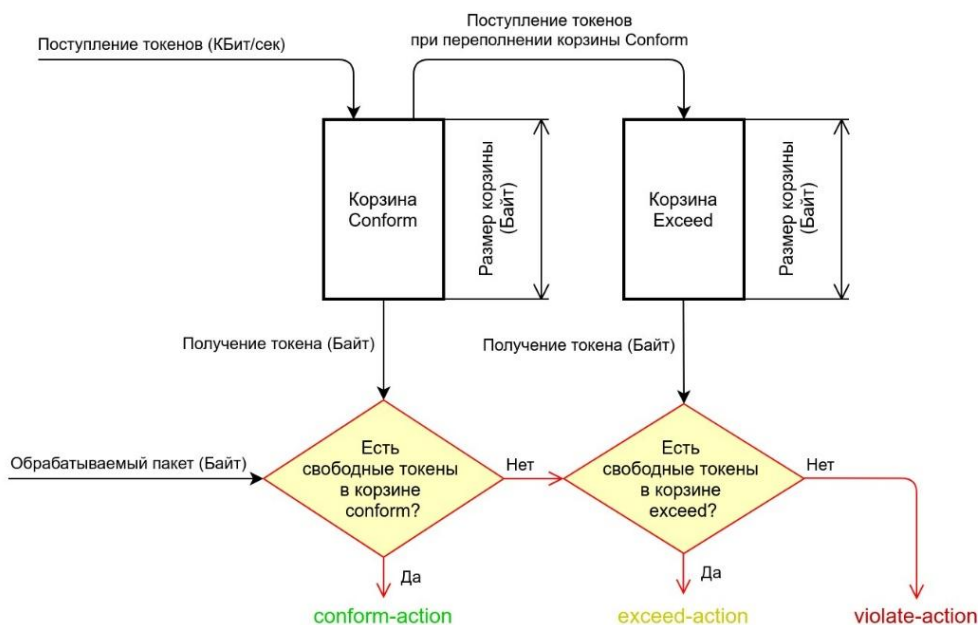
```

rtt# show qos policy statistics gigabitethernet 1/0/20

```

11.2.3. Механизм работы полисера

Механизм полисера реализован по алгоритму однокоростного трехцветного полисера (Single Rate Three Color Marker/Policers).



У такого механизма есть две корзины токенов (conform и exceed), каждая из которых имеет свой размер. Conform-корзина наполняется токенами до определенного размера с течением времени. Корзина exceed наполняется излишками токенов корзины conform.

Обрабатываемый RTT трафик забирает из корзин токены размером, соответствующим размеру пакета. Если размер пришедшего пакета покрывает токены корзины conform, то пакет окрасится в

«зелёный», и для него применится действие, соответствующее настройке conform-action. Если токенов в conform-корзине не хватило, но есть достаточно токенов в корзине exceed, то пакет окрасится в «жёлтый», и применится действие exceed-action. Если токенов недостаточно в обеих корзинах, то пакет будет окрашен в «красный», и применится действие violate-action.

Действия conform-action, exceed-action, violate-action определяются одним из следующих вариантов:

- пропустить (permit);
- отбросить (deny);
- пропустить и изменить cos/dscp (set cos, set dscp).

11.3. MPLS QoS

QoS в MPLS позволяет реализовать управление приоритетами трафика внутри MPLS-домена путём использования EXP-поля в заголовке MPLS. Это даёт возможность классифицировать, маркировать и обрабатывать трафик в соответствии с политиками DiffServ, обеспечивая нужный уровень сервиса для критически важных приложений в L3VPN и других сценариях. Рассмотрим поведение маршрутизатора по умолчанию с полями DSCP и EXP в различных сценариях:

Если RTT выступает в роли PE-маршрутизатора:

- При инкапсуляции IP в MPLS происходит наследование старших трех битов DSCP исходного IP-заголовка в поле EXP сервисной и транспортной меток;
- При инкапсуляции IP в MPLS, а затем в GRE, наследуются старшие три бита исходного IP-заголовка в поле EXP сервисной и транспортной меток. Значение DSCP исходного IP-заголовка наследуется в поле DSCP внешнего IP-заголовка;
- При декапсуляции MPLS значение поля DSCP исходного IP-пакета не изменяется.

Если RTT выступает в роли P-маршрутизатора:

- При операции Swap label значение поля EXP будет унаследовано;
- При операции Explicit null значение поля EXP будет унаследовано;
- При инкапсуляции в GRE происходит наследование поля EXP в три старших бита поля DSCP внешнего IP-заголовка.

Если RTT выступает в роли ASBR-маршрутизатора:

- При операции push label (добавление новых меток) значение поля EXP будет наследовано в поле EXP всех новых меток;
- При операции pop label значение поля EXP не будет унаследовано в стек нижестоящих MPLS меток.

12.УПРАВЛЕНИЕ МАРШРУТИЗАЦИЕЙ

12.1. Политика фильтрации маршрутной информации

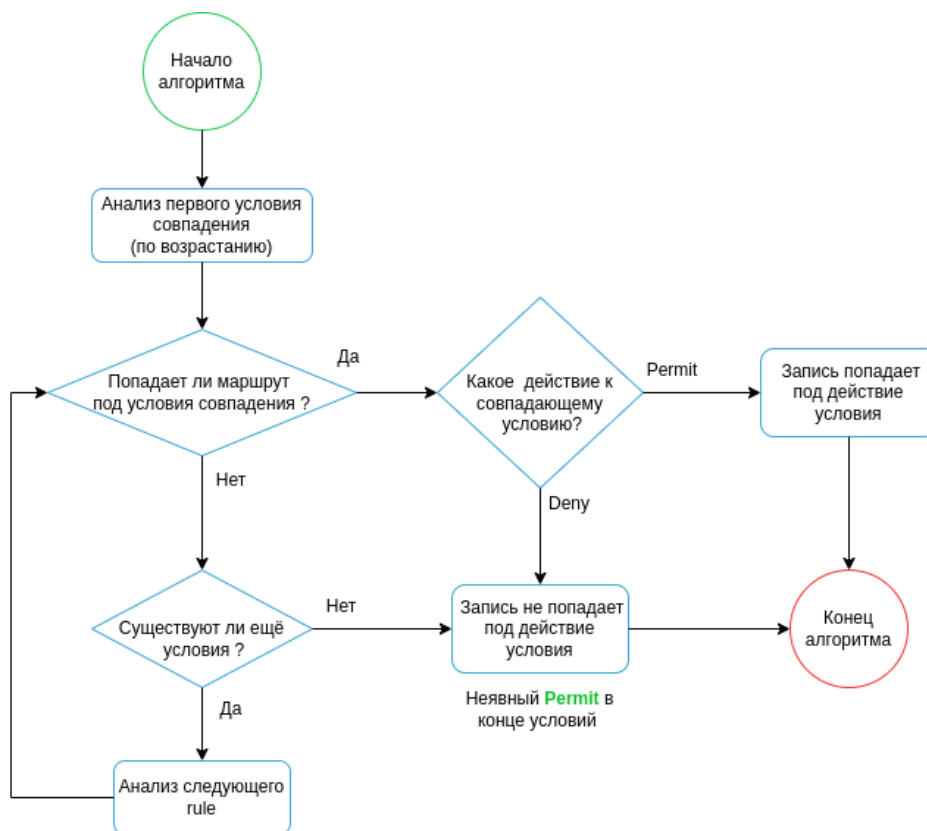
Фильтрация маршрутной информации в динамических протоколах маршрутизации осуществляется с помощью `prefix-list` и `route-map`. Структурно они состоят из последовательных условий, совпадений (`match`) и действий, применяемых для заданных условий – `permit/deny`. В зависимости от политики обработки маршрутов (импорт или экспорт) конечное неявное условие (правило) может быть разным для некоторых протоколов маршрутизации.

Правила фильтрации маршрутов в `prefix-list` и `route-map` выглядят следующим образом:

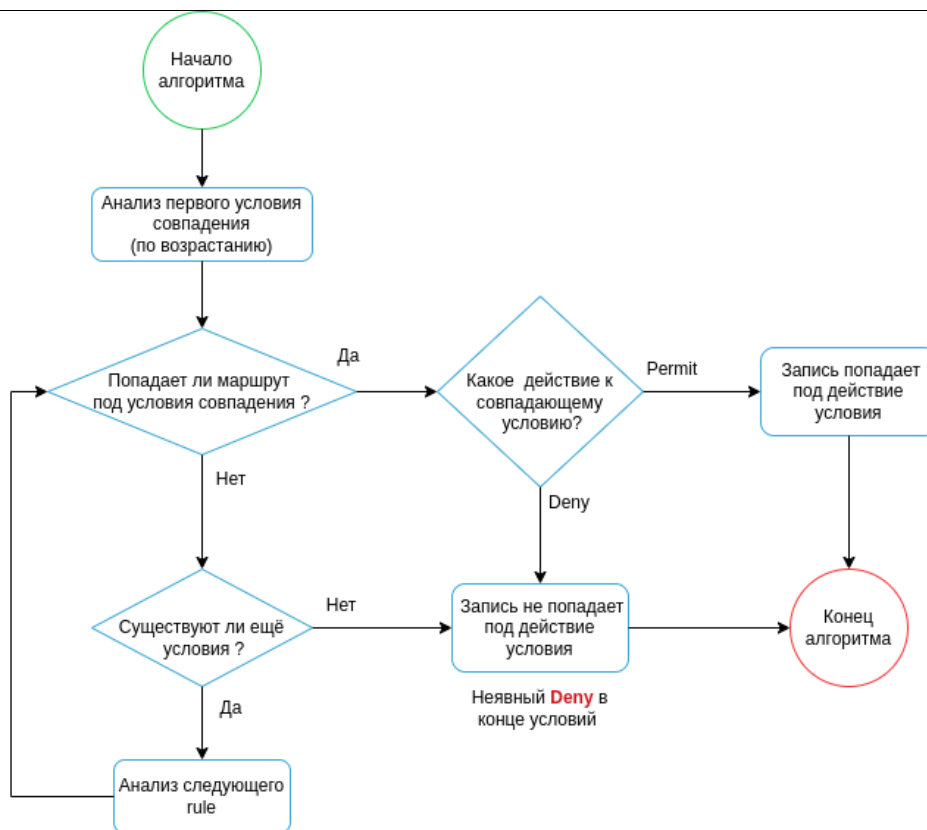
- В `route-map`: маршруты сопоставляются с IP-префиксами по очереди в списке правил в соответствии с их нумерациями – от самого младшего до самого старшего.
- В `prefix-list`: маршруты сопоставляются с IP-префиксами по очереди в списке IP-префиксов, указанных в `prefix-list` в том порядке, котором они были заданы при создании/редактировании этого `prefix-list`.
- Если маршрут соответствует префиксу, указанному в последовательности или правиле, к нему применяются описанные действия, и он перестает сопоставляться с другими префиксами в списке IP-префиксов.
- Если маршруты не соответствуют ни одному префиксу в списке IP-префиксов, к ним применяется неявное правило в конце `prefix-list` или `route-map`.

Ниже приведены блок-схемы с автоматом состояний обработки правил фильтрации маршрутной информации для политик `import` и `export`.

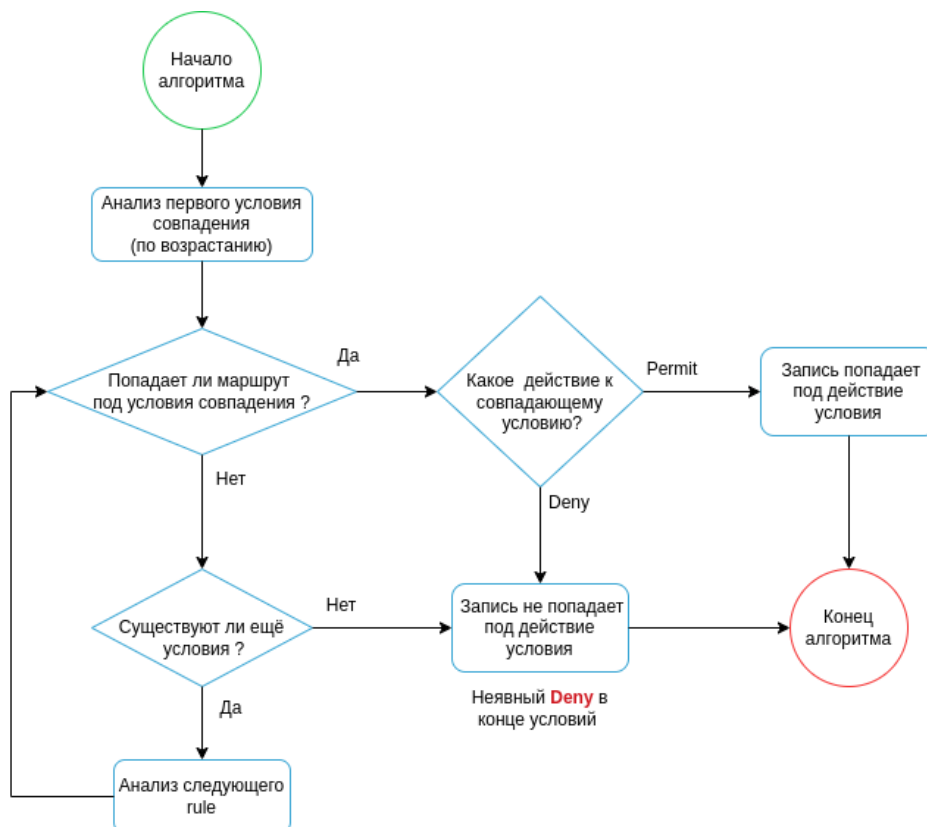
Обработка правил фильтрации маршрутной информации политики `export` для протоколов RIP, OSPF, IS-IS, iBGP:



Обработка правил фильтрации маршрутной информации политики import для протоколов RIP, OSPF, IS-IS, iBGP:



Обработка правил фильтрации маршрутной информации политик import/export для протокола eBGP:



Ниже приведены обобщенные таблицы по протоколам:

12.1.1. Протокол RIP

in/out	Политика по умолчанию	Способы анонсирования	Способы фильтрации	Уровни применения политик фильтрации
Import	Получение маршрутной информации не ограничено.	Network, Redistribute	Route-map — последнее (неявное) правило <u>запрещает</u> все, что явно не разрешено предыдущими правилами. Prefix-list — последнее (неявное) правило <u>запрещает</u> все, что явно не разрешено предыдущими правилами.	Процесс RIP
Export	Без отдельных команд анонсирования маршрутизатор не отправляет маршрутную информацию.		Prefix-list — последнее (неявное) правило <u>разрешает</u> все, что явно не запрещено предыдущими правилами. Prefix-list — последнее (неявное) правило <u>разрешает</u> все, что явно не запрещено предыдущими правилами.	

12.1.2. Протокол OSPF

in/out	Политика по умолчанию	Способы анонсирования	Способы фильтрации	Уровни применения политик фильтрации
Import	Получение маршрутной информации не ограничено.	Redistribute	Route-map — последнее (неявное) правило <u>разрешает</u> все, что явно не разрешено предыдущими правилами.	Процесс OSPF
		Route-map, Prefix-list	Route-map — последнее (неявное) правило <u>запрещает</u> все, что явно не разрешено предыдущими правилами. Prefix-list — последнее (неявное) правило <u>запрещает</u> все, что явно не разрешено предыдущими правилами.	
Export	Анонсируется информация о интерфейсах, на	Route-map, Prefix-list	Route-map — последнее (неявное) правило <u>разрешает</u> все, что явно не запрещено предыдущими правилами.	

in/out	Политика по умолчанию	Способы анонсирования	Способы фильтрации	Уровни применения политик фильтрации
	которых включен протокол OSPF.		<p>Prefix-list — последнее (неявное) правило <u>разрешает</u> все, что явно не запрещено предыдущими правилами.</p> <p><i>Фильтрация анонсируемой маршрутной информации возможна для следующих типов OSPF-маршрутов: E2, E1.</i></p>	

12.1.3. Протокол IS-IS

in/out	Политика по умолчанию	Способы анонсирования	Способы фильтрации	Уровни применения политик фильтрации
Import	Получение маршрутной информации не ограничено.	Network, Redistribute	<p>Route-map — последнее (неявное) правило <u>запрещает</u> все, что явно не разрешено предыдущими правилами.</p> <p>Prefix-list — последнее (неявное) правило <u>запрещает</u> все, что явно не разрешено предыдущими правилами.</p>	Процесс IS-IS
Export	Анонсируется информация о интерфейсах, на которых включен протокол IS-IS.		<p>Route-map — последнее (неявное) правило <u>разрешает</u> все, что явно не запрещено предыдущими правилами.</p> <p>Prefix-list — последнее (неявное) правило <u>разрешает</u> все, что явно не запрещено предыдущими правилами.</p>	

12.1.4. Протокол iBGP

in/out	Политика по умолчанию	Способы Анонсирования	Способы фильтрации	Уровни применения политик фильтрации
Import	Получение маршрутной информации не ограничено.	Network, Redistribute	Route-map — последнее (неявное) правило <u>запрещает</u> все, что явно не разрешено предыдущими правилами. Prefix-list — последнее (неявное) правило <u>запрещает</u> все, что явно не разрешено предыдущими правилами.	address-family, peer-group, neighbor
Export	Анонсируются все маршруты, попавшие в RIB по протоколу BGP.		Route-map — последнее (неявное) правило <u>разрешает</u> все, что явно не запрещено предыдущими правилами. Prefix-list — последнее (неявное) правило <u>разрешает</u> все, что явно не запрещено предыдущими правилами.	

12.1.5. Протокол eBGP

in/out	Политика по умолчанию	Способы Анонсирования	Способы фильтрации	Уровни применения политик фильтрации
Import	Получение маршрутной информации не ограничено	Network, Redistribute	Route-map — последнее (неявное) правило <u>запрещает</u> все, что явно не разрешено предыдущими правилами. Prefix-list — последнее (неявное) правило <u>запрещает</u> все, что явно не разрешено предыдущими правилами.	address-family, peer-group, neighbor

in/out	Политика по умолчанию	Способы Анонсирования	Способы фильтрации	Уровни применения политик фильтрации
Export	Анонсирование маршрутов <u>запрещено</u> до применения разрешающего route-map или prefix-list		Route-map — последнее (неявное) правило <u>запрещает</u> все, что явно не разрешено предыдущими правилами. Prefix-list — последнее (неявное) правило <u>запрещает</u> все, что явно не разрешено предыдущими правилами.	

12.2. Конфигурирование статических маршрутов

Статическая маршрутизация — вид маршрутизации, при котором маршруты указываются в явном виде при конфигурации маршрутизатора без использования протоколов динамической маршрутизации.

12.2.1. Алгоритм настройки

Добавить статический маршрут возможно командой в режиме глобальной конфигурации:

```
rtt(config)# ip route [ vrf <VRF> ] <SUBNET> { { <NEXTHOP> [ resolve ] [ bfd ] [
unit <ID> ] | interface <IF> | tunnel <TUN> | blackhole | unreachable | prohibit
} [ track <TRACK-ID> ] [ name <NAME>] } | wan load-balance rule <RULE> } [
<METRIC> ]
no ip route [ vrf <VRF> ] <SUBNET> [ <METRIC> ] [ unit <ID> ]
```

- <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;
- <ID> — номер юнита, принимает значения [1..4];
- <IF> — имя IP-интерфейса, задаётся в виде, описанном в разделе **Типы и порядок именования интерфейсов маршрутизатора**;
- <TUN> — имя туннеля, задаётся в виде, описанном в разделе **Типы и порядок именования туннелей маршрутизатора**;
- <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);
- <METRIC> – метрика маршрута, принимает значения [0..255];
- <TRACK-ID> – идентификатор Tracking-объекта. Если маршрут привязан к Tracking-объекту, то он появится в системе только при выполнении всех условий, заданных в объекте;
- <NAME> – имя (описание) маршрута, текстовая переменная длиной до 31 символа;
- bfd – при указании данного ключа активируется удаление статического маршрута в случае недоступности next-hop. Для работы данного механизма должен быть запущен механизм BFD с IP-адресом next-hop.

Проверка next-hop при помощи протокола BFD. В случае недоступности next-hop маршрут удаляется.

Для добавления статического IPv6-маршрута к указанной подсети используется команда:

```
rtt(config)# ipv6 route [ vrf <VRF> ] <SUBNET> { { <NEXTHOP> [ resolve ] [bfd] [
unit <ID> ] | interface <IF> | blackhole | unreachable | prohibit [ <METRIC> ] [
name <NAME>] } | wan load-balance rule <RULE> [ <METRIC> ] }
no ipv6 route [ vrf <VRF> ] <SUBNET> [ <METRIC> ] [ unit <ID> ]
```

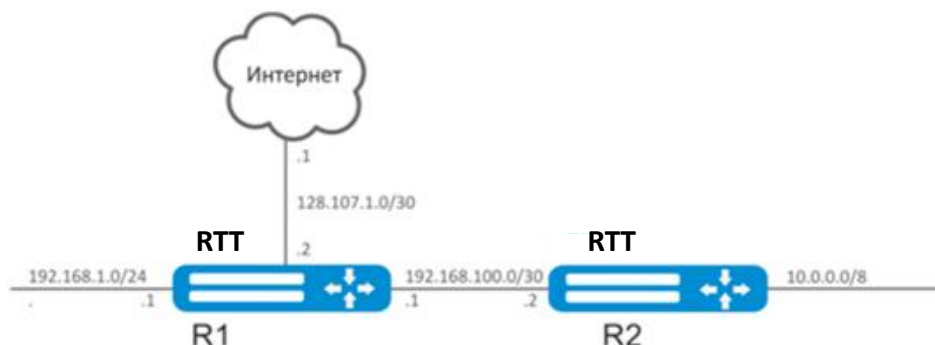
- <VRF> – имя экземпляра VRF, задается строкой до 31 символа;
- <SUBNET> – адрес назначения, может быть задан в следующих видах:
 - X:X:X:X – IPv6-адрес хоста, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];
 - X:X:X:X/EE – IPv6-адрес подсети с маской в виде префикса, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].
- <NEXTHOP> – IPv6-адрес шлюза, задаётся 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;
- <ID> – номер юнита, принимает значения [1..4];
- <IF> – имя IP-интерфейса, задаётся в виде, описанном в разделе **Типы и порядок именования интерфейсов маршрутизатора**;
- 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);
- [METRIC] – метрика маршрута, принимает значения [0..255];
- <NAME> – имя (описание) маршрута, текстовая переменная длиной до 31 символа;

- bfd – при указании данного ключа активируется проверка next-hop при помощи протокола BFD. В случае недоступности next-hop маршрут удаляется.

12.2.2. Пример настройки

Задача:

Настроить доступ к сети Internet для пользователей локальных сетей 192.168.1.0/24 и 10.0.0.0/8, используя статическую маршрутизацию. На устройстве R1 создать шлюз для доступа к сети Internet. Трафик внутри локальной сети должен маршрутизироваться внутри зоны LAN, трафик из сети Internet должен относиться к зоне WAN.



Решение:

Зададим имя устройства для маршрутизатора R1:

```
rtt# hostname R1
```

Для интерфейса gi1/0/1 укажем адрес 192.168.1.1/24 и зону «LAN». Через данный интерфейс R1 будет подключен к сети 192.168.1.0/24:

```
rtt(config)# interface gi1/0/1
rtt(config-if-gi)# security-zone LAN
rtt(config-if-gi)# ip address 192.168.1.1/24
rtt(config-if-gi)# exit
```

Для интерфейса gi1/0/2 укажем адрес 192.168.100.1/30 и зону «LAN». Через данный интерфейс R1 будет подключен к устройству R2 для последующей маршрутизации трафика:

```
rtt(config)# interface gi1/0/2
rtt(config-if-gi)# security-zone LAN
rtt(config-if-gi)# ip address 192.168.100.1/30
rtt(config-if-gi)# exit
```

Для интерфейса gi1/0/3 укажем адрес 128.107.1.2/30 и зону «WAN». Через данный интерфейс R1 будет подключен к сети Internet:

```
rtt(config)# interface gi1/0/3
rtt(config-if-gi)# security-zone WAN
```



```
rtt(config-if-gi)# ip address 128.107.1.2/30
rtt(config-if-gi)# exit
```

Создадим маршрут для взаимодействия с сетью 10.0.0.0/8, используя в качестве шлюза устройство R2 (192.168.100.2):

```
rtt(config)# ip route 10.0.0.0/8 192.168.100.2
```

Создадим маршрут для взаимодействия с сетью Internet, используя в качестве nexthop шлюз провайдера (128.107.1.1):

```
rtt(config)# ip route 0.0.0.0/0 128.107.1.1
```

Зададим имя устройства для маршрутизатора R2:

```
rtt# hostname R2
```

Для интерфейса gi1/0/1 укажем адрес 10.0.0.1/8 и зону «LAN». Через данный интерфейс R2 будет подключен к сети 10.0.0.0/8:

```
rtt(config)# interface gi1/0/1
rtt(config-if-gi)# security-zone LAN
rtt(config-if-gi)# ip address 10.0.0.1/8
rtt(config-if-gi)# exit
```

Для интерфейса gi1/0/2 укажем адрес 192.168.100.2/30 и зону «LAN». Через данный интерфейс R2 будет подключен к устройству R1 для последующей маршрутизации трафика:

```
rtt(config)# interface gi1/0/2
rtt(config-if-gi)# security-zone LAN
rtt(config-if-gi)# ip address 192.168.100.2/30
rtt(config-if-gi)# exit
```

Создадим маршрут по умолчанию, указав в качестве nexthop IP-адрес интерфейса gi1/0/2 маршрутизатора R1 (192.168.100.1):

```
rtt(config)# ip route 0.0.0.0/0 192.168.100.1
```

Проверить таблицу маршрутов можно командой:

```
rtt# show ip route
```

12.3. Конфигурирование статических multipath-маршрутов

Статические multipath-маршруты позволяют использовать стратегию ECMP для эффективного распределения трафика между несколькими равнозначными маршрутами, без необходимости применения динамических протоколов маршрутизации.

12.3.1. Алгоритм настройки

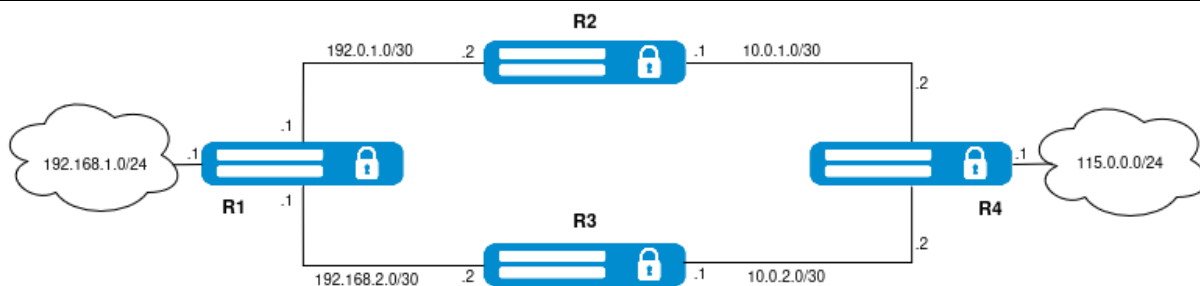
Шаг	Описание	Команда	Ключи
1	Создать статический multipath ipv4 маршрут.	<pre> rtt(config)# ip route multipath [vrf <VRF>] <SUBNET> [track <TRACK-ID>] [name <NAME>] [<METRIC>] </pre>	<p><VRF> – имя экземпляра VRF, задается строкой до 31 символа;</p> <p><SUBNET> – адрес назначения, может быть задан в следующих видах:</p> <ul style="list-style-type: none"> • 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]. <p><TRACK-ID> – идентификатор Tracking-объекта. Если маршрут привязан к Tracking-объекту, то он появится в системе только при выполнении всех условий, заданных в объекте;</p> <p><NAME> – имя (описание) маршрута, текстовая переменная длиной до 31 символа;</p> <p><METRIC> – метрика маршрута, принимает значения [0..255].</p>
2	Создать статический multipath ipv6 маршрут.	<pre> rtt(config)# ipv6 route multipath [vrf <VRF>] <SUBNET> [name <NAME>] [<METRIC>] </pre>	<p><VRF> – имя экземпляра VRF, задается строкой до 31 символа;</p> <p><SUBNET> – адрес назначения, может быть задан в следующих видах:</p> <ul style="list-style-type: none"> • 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]. <p><NAME> – имя (описание) маршрута, текстовая переменная длиной до 31 символа;</p> <p><METRIC> – метрика маршрута, принимает значения [0..255].</p>

Шаг	Описание	Команда	Ключи
3	Настроить шлюзы для маршрута.	<pre> rtt(config)# gateway { <NEXTHOP> [<WEIGHT>] [bfd] [unit <ID>] [track <TRACK-ID>] <IF> [<WEIGHT>] <TUN> [<WEIGHT>] } </pre>	<p><NEXTHOP> – адрес шлюза, может быть задан в следующих видах:</p> <ul style="list-style-type: none"> • AAA.BBB.CCC.DDD – IP-адрес шлюза, где каждая часть принимает значения [0..255]; • X:X:X:X::X – IPv6-адрес шлюза, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]; <p><WEIGHT> – метрика шлюза, принимает значения [0..255].</p> <p><ID> – номер юнита, принимает значения [1..4];</p> <p><IF> – имя IP-интерфейса, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;</p> <p><TUN> – имя туннеля, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.</p> <p><TRACK-ID> – идентификатор Tracking-объекта. Если шлюз привязан к Tracking-объекту, то он появится в системе только при выполнении всех условий, заданных в объекте. Доступно только для ipv4;</p> <p>bfd – при указании данного ключа активируется проверка доступности шлюза при помощи протокола BFD. В случае недоступности шлюз удаляется из таблицы маршрутизации.</p>

12.3.2. Пример настройки

Задача:

Обеспечить связность сетей 192.168.1.0/24 и 115.0.0.0/24, распределяя трафик между R2 и R3 с помощью статического multipath-маршрута.



Решение:

Конфигурация R1:

R1

```
rtt# hostname R1
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# description "to LAN"
rtt(config-if-gi)# ip firewall disable
rtt(config-if-gi)# ip address 192.168.1.1/24
rtt(config-if-gi)# exit
rtt(config)# interface gigabitethernet 1/0/3
rtt(config-if-gi)# description "to R2"
rtt(config-if-gi)# ip firewall disable
rtt(config-if-gi)# ip address 192.0.1.1/30
rtt(config-if-gi)# exit
rtt(config)# interface gigabitethernet 1/0/4
rtt(config-if-gi)# description "to R3"
rtt(config-if-gi)# ip address 192.0.2.1/30
rtt(config-if-gi)# exit
rtt(config)# ip route multipath 115.0.0.0/24
rtt(config-multipath-route)# gateway 192.0.1.2
rtt(config-multipath-route)# gateway 192.0.2.2
rtt(config-multipath-route)# exit
```

Конфигурация R2:

R2

```
rtt(config)# hostname R2
rtt(config)# interface gigabitethernet 1/0/3
rtt(config-if-gi)# description "to R1"
rtt(config-if-gi)# ip firewall disable
rtt(config-if-gi)# ip address 192.0.1.2/30
rtt(config-if-gi)# exit
rtt(config)# interface gigabitethernet 1/0/4
rtt(config-if-gi)# description "to R4"
rtt(config-if-gi)# ip firewall disable
rtt(config-if-gi)# ip address 10.0.1.1/30
rtt(config-if-gi)# exit
rtt(config)#
rtt(config)# ip route 115.0.0.0/24 10.0.1.2
rtt(config)# ip route 192.168.0.0/24 192.0.1.1
```

Конфигурация R3:

R3

```
rtt(config)# hostname R3
rtt(config)# interface gigabitethernet 1/0/3
rtt(config-if-gi)# description "to R1"
rtt(config-if-gi)# ip firewall disable
rtt(config-if-gi)# ip address 192.0.2.2/30
rtt(config-if-gi)# exit
rtt(config)# interface gigabitethernet 1/0/4
rtt(config-if-gi)# description "to R4"
rtt(config-if-gi)# ip firewall disable
rtt(config-if-gi)# ip address 10.0.2.1/30
rtt(config-if-gi)# exit
rtt(config)# ip route 115.0.0.0/24 10.0.2.2
rtt(config)# ip route 192.168.0.0/24 192.0.2.1
```

Конфигурация R4:

R4

```
rtt(config)# hostname R4
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# description "to LAN"
rtt(config-if-gi)# ip firewall disable
rtt(config-if-gi)# ip address 115.0.0.1/24
rtt(config-if-gi)# exit
rtt(config)# interface gigabitethernet 1/0/3
rtt(config-if-gi)# description "to R2"
rtt(config-if-gi)# ip firewall disable
rtt(config-if-gi)# ip address 10.0.1.2/30
rtt(config-if-gi)# exit
rtt(config)# interface gigabitethernet 1/0/4
rtt(config-if-gi)# description "to R3"
rtt(config-if-gi)# ip firewall disable
rtt(config-if-gi)# ip address 10.0.2.2/30
rtt(config-if-gi)# exit
rtt(config)# ip route multipath 192.168.0.0/24
rtt(config-multipath-route)# gateway 10.0.1.1
rtt(config-multipath-route)# gateway 10.0.2.1
rtt(config-multipath-route)# exit
```

Проверить таблицу маршрутов можно командой:

```
rtt# show ip route
```

12.4. Настройка RIP

RIP — дистанционно-векторный протокол динамической маршрутизации, который использует количество транзитных участков в качестве метрики маршрута. Максимальное количество транзитных участков (hop), разрешенное в RIP, равно 15. Каждый RIP-маршрутизатор по умолчанию вещает в сеть

свою полную таблицу маршрутизации один раз в 30 секунд. RIP работает на 3 уровне стека TCP/IP, используя UDP-порт 520.

12.4.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Настроить приоритетность протокола RIP-маршрутизации для основной таблицы маршрутизации (необязательно).	<code>rtt(config)# ip protocols rip preference <VALUE></code>	<VALUE> – приоритетность протокола, принимает значения в диапазоне [1..255]. Значение по умолчанию: RIP (100).
2	Настроить емкость таблиц маршрутизации протокола RIP (необязательно).	<code>rtt(config)# ip protocols rip max-routes <VALUE></code>	<VALUE> – количество маршрутов протокола RIP в маршрутной таблице, принимает значения в диапазоне [1..10000]; Значение по умолчанию: 10000.
3	Создать списки IP-подсетей, которые в дальнейшем будут использоваться для фильтрации анонсируемых и получаемых IP-маршрутов.	<code>rtt(config)# ip prefix-list <NAME></code>	<NAME> – имя конфигурируемого списка подсетей, задаётся строкой до 31 символа.
4	Разрешить (permit) или запретить (deny) списки префиксов.	<div> <code>rtt(config-pl)# permit {object-group <OBJ-GROUP-NETWORK-NAME> <ADDR/LEN> <IPV6-ADDR/LEN> } [{ eq <LEN> le <LEN> ge <LEN> [le <LEN>] }]</code> </div> <div> <code>rtt(config-pl)# deny {object-group <OBJ-GROUP-NETWORK-NAME> <ADDR/LEN> <IPV6-ADDR/LEN> } [{ eq <LEN> le <LEN> ge <LEN> [le <LEN>] }]</code> </div>	<p><OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа;</p> <p><LEN> – длина префикса, принимает значения [1..32] в IP-списках префиксов;</p> <ul style="list-style-type: none"> • 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.
5	Перейти в режим настройки параметров RIP-процесса.	<code>rtt(config)# router rip</code> <code>rtt(config-rip)#</code>	
6	Включить RIP-протокол.	<code>rtt(config-rip)# enable</code>	

Шаг	Описание	Команда	Ключи
7	Определить алгоритм аутентификации протокола RIP (необязательно).	<code>rtt(config-rip)# authentication algorithm { cleartext md5 }</code>	<ul style="list-style-type: none"> cleartext – пароль, передается открытым текстом; md5 – пароль, хешируется по алгоритму md5.
8	Установить пароль для аутентификации с соседом (необязательно).	<code>rtt(config-rip)# authentication key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<p><CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов;</p> <p><ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).</p>
9	Определить список паролей для аутентификации через алгоритм хеширования md5 (необязательно).	<code>rtt(config-rip)# authentication key-chain <KEYCHAIN></code>	<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.
10	Выключить анонсирование маршрутов на интерфейсах/туннелях/bridge, где это не нужно (необязательно).	<code>rtt(config-rip)# passive-interface {<IF> <TUN> }</code>	<p><IF> – интерфейс и идентификатор;</p> <p><TUN> – имя и номер туннеля.</p>
11	Установить временной интервал, по истечении которого производится анонсирование (необязательно).	<code>rtt(config-rip)# timers update <TIME></code>	<p><TIME> – время в секундах, принимает значения [12..65535].</p> <p>Значение по умолчанию: 180 секунд.</p>
12	Установить временной интервал корректности маршрутной записи без обновления (необязательно).	<code>rtt(config-rip)# timers invalid <TIME></code>	<p><TIME> – время в секундах, принимает значения [12..65535].</p> <p>Значение по умолчанию: 180 секунд.</p>
13	Установить временной интервал, по истечении которого производится удаление маршрута (необязательно).	<code>rtt(config-rip)# timers flush <TIME></code>	<p><TIME> – время в секундах, принимает значения [12..65535].</p> <p>При установке значения нужно учитывать следующее правило: «timersinvalid + 60»</p> <p>Значение по умолчанию: 240 секунд.</p>
14	Включить анонсирование подсетей.	<code>rtt(config-rip)# network <ADDR/LEN></code>	<p><ADDR/LEN> – адрес подсети, указывается в следующем формате:</p> <p>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].</p>

Шаг	Описание	Команда	Ключи
15	Добавить фильтрацию подсетей во входящих или исходящих обновлениях (необязательно).	<code>rtt(config-rip) # prefix-list <PREFIX- LIST-NAME> { in out }</code>	<p><PREFIX-LIST-NAME> – имя сконфигурированного списка подсетей, задаётся строкой до 31 символа.</p> <ul style="list-style-type: none"> • in – фильтрация входящих маршрутов; • out – фильтрация анонсируемых маршрутов.
16	Включить анонсирование маршрутов, полученных альтернативным способом (необязательно).	<code>rtt(config-rip) # redistribute static [route-map <NAME>]</code>	<NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых статических маршрутов, задаётся строкой до 31 символа.
		<code>rtt(config-rip) # redistribute connected [route-map <NAME>]</code>	<NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых напрямую подключенных подсетей, задаётся строкой до 31 символа.
		<code>rtt(config-rip) # redistribute ospf <ID><ROUTE-TYPE> [route-map <NAME>]</code>	<p><ID> – номер процесса, может принимать значение [1..65535];</p> <p><ROUTE-TYPE> – тип маршрута:</p> <ul style="list-style-type: none"> • intra - area – анонсирование маршрутов OSPF-процесса в пределах зоны; • inter - area – анонсирование маршрутов OSPF-процесса между зонами; • external 1 – анонсирование внешних маршрутов OSPF-формата 1; • external 2 – анонсирование внешних маршрутов OSPF-формата 2; <p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых OSPF-маршрутов, задаётся строкой до 31 символа.</p>

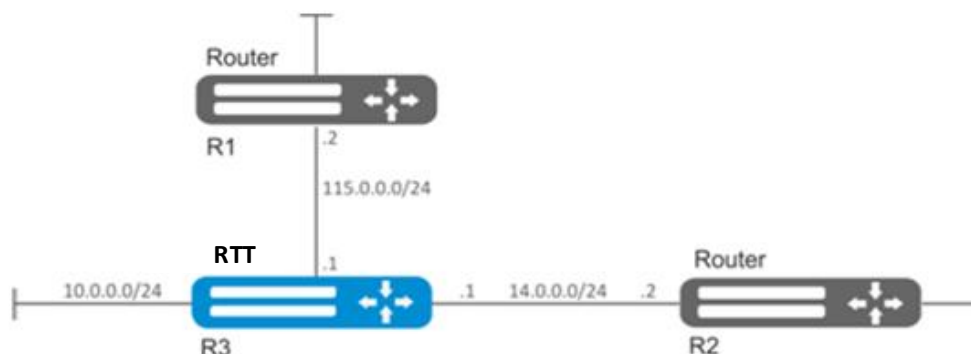
Шаг	Описание	Команда	Ключи
		<pre>rtt(config-rip) # redistribute isis <ID><ROUTE-TYPE> [route-map <NAME>]</pre>	<p><ID> – номер процесса, может принимать значение [1..65535];</p> <p><ROUTE-TYPE> – тип маршрута:</p> <ul style="list-style-type: none"> • level-1 – анонсирование маршрутов ISIS-процесса уровня 1; • level-2 – анонсирование маршрутов ISIS-процесса уровня 2; • inter-area – анонсирование межзоновых маршрутов ISIS-процесса. <p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых ISIS-маршрутов, задаётся строкой до 31 символа.</p>
		<pre>rtt(config-rip) # redistribute bgp <AS> [route-map <NAME>]</pre>	<p><AS> – номер автономной системы, может принимать значения [1..4294967295];</p> <p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых BGP-маршрутов, задаётся строкой до 31 символа.</p>
17	Перейти в режим конфигурирования интерфейса/туннеля/сетевого моста.	<pre>rtt(config) # interface <IF-TYPE><IF-NUM></pre>	<p><IF-TYPE> – тип интерфейса;</p> <p><IF-NUM> – F/S/P – F-фрейм (1), S – слот (0), P – порт.</p>
		<pre>rtt(config) # tunnel <TUN-TYPE><TUN-NUM></pre>	<p><TUN-TYPE> – тип туннеля;</p> <p><TUN-NUM> – номер туннеля.</p>
		<pre>rtt(config) # bridge <BR-NUM></pre>	<p><BR-NUM> – номер bridge.</p>
18	Установить величину метрики RIP-маршрутов на интерфейсе (необязательно).	<pre>rtt(config-if-gi) # ip rip metric <VALUE></pre>	<p><VALUE> – величина метрики, задаётся в размере [0..32767].</p> <p>Значение по умолчанию: 5.</p>

Шаг	Описание	Команда	Ключи
19	Установить режим анонсирования маршрутов по протоколу RIP (необязательно).	<code>rtt(config-if-gi)# ip rip mode <MODE></code>	<p><MODE> – режим анонсирования маршрутов:</p> <ul style="list-style-type: none"> • multicast – маршруты анонсируются в многоадресном режиме; • broadcast – маршруты анонсируются в широковещательном режиме; • unicast – маршруты анонсируются в unicast-режиме соседям. <p>Значение по умолчанию: multicast.</p>
20	Задать IP-адрес соседа для установления отношения в unicast-режиме анонсирования маршрутов (необязательно).	<code>rtt(config-if-gi)# ip rip neighbor <ADDR></code>	<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
21	Включить суммаризацию подсетей (необязательно).	<code>rtt(config-if-gi)# ip rip summary-address <ADDR/LEN></code>	<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].
22	Включить протокол BFD для протокола RIP (необязательно).	<code>rtt(config-if-gi)# ip rip bfd-enable</code>	

12.4.2. Пример настройки

Задача:

Настроить на маршрутизаторе протокол RIP для обмена маршрутной информацией с соседними маршрутизаторами. Маршрутизатор должен анонсировать статические маршруты и подсети 115.0.0.0/24, 14.0.0.0/24, 10.0.0.0/24. Анонсирование маршрутов должно происходить каждые 25 секунд.



Решение:

Предварительно нужно настроить IP-адреса на интерфейсах согласно схеме сети, приведенной на рисунке выше.

Перейдём в режим конфигурирования протокола RIP:

```
rtt(config)# router rip
```

Укажем подсети, которые будут анонсироваться протоколом: 115.0.0.0/24, 14.0.0.0/24 и 10.0.0.0/24:

```
rtt(config-rip)# network 115.0.0.0/24
rtt(config-rip)# network 14.0.0.0/24
rtt(config-rip)# network 10.0.0.0/24
```

Для анонсирования протоколом статических маршрутов выполним команду:

```
rtt(config-rip)# redistribute static
```

Настроим таймер, отвечающий за отправку маршрутной информации:

```
rtt(config-rip)# timers update 25
```

После установки всех требуемых настроек включим протокол:

```
rtt(config-rip)# enable
```

Для того чтобы просмотреть таблицу маршрутов RIP, воспользуемся командой:

```
rtt# show ip rip
```



Помимо настройки протокола RIP необходимо в firewall разрешить UDP-порт 520.

12.5. Настройка RIPng

RIPng – дистанционно-векторный протокол динамической маршрутизации, использующий алгоритм Беллмана-Форда для нахождения наилучшего маршрута. Данная версия протокола включает в себя поддержку работы с IPv6. RIPng работает на 3 уровне стека TCP/IP, используя UDP-порт 521.

12.5.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Настроить приоритетность протокола RIPng для основной таблицы маршрутизации (необязательно).	<code>rtt(config)# ipv6 protocols rip preference <VALUE></code>	<VALUE> – приоритетность протокола, принимает значения в диапазоне [1..255]. Значение по умолчанию: RIPng (100).
2	Настроить емкость таблиц маршрутизации протокола RIPng (необязательно).	<code>rtt(config)# ipv6 protocols rip max-routes <VALUE></code>	<VALUE> – количество маршрутов протокола RIP в маршрутной таблице, принимает значения в диапазоне [1..10000]; Значение по умолчанию: 10000.
3	Создать списки IPv6-подсетей, которые в дальнейшем будут использоваться для фильтрации анонсируемых и получаемых IPv6-маршрутов.	<code>rtt(config)# ipv6 prefix-list <NAME></code>	<NAME> – имя конфигурируемого списка подсетей, задаётся строкой до 31 символа.
4	Разрешить (permit) или запретить (deny) списки префиксов.	<div> <code>rtt(config-pl)# permit {object-group <OBJ-GROUP-NETWORK-NAME> <IPV6-ADDR/LEN> } [{ eq <LEN> le <LEN> ge <LEN> [le <LEN>] }]</code> </div> <div> <code>rtt(config-pl)# deny {object-group <OBJ-GROUP-NETWORK-NAME> <IPV6-ADDR/LEN> } [{ eq <LEN> le <LEN> ge <LEN> [le <LEN>] }]</code> </div>	<p><OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа;</p> <p><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];</p> <p><LEN> – длина префикса, принимает значения [1..128] в IPv6-списках префиксов;</p> <ul style="list-style-type: none"> 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.
5	Перейти в режим настройки параметров RIPng-процесса.	<code>rtt(config)# ipv6 router rip</code> <code>rtt(config-ripng)#</code>	
6	Включить протокол RIPng.	<code>rtt(config-ripng)# enable</code>	

Шаг	Описание	Команда	Ключи
7	Отключить анонсирование маршрутов на интерфейсах/туннелях/bridge, где это не нужно (необязательно).	<code>rtt(config-ripng)# passive-interface {<IF> <TUN> <BR-NUM> }</code>	<IF> – интерфейс и идентификатор; <BR-NUM> – номер bridge; <TUN> – имя и номер туннеля.
8	Установить временной интервал, по истечении которого производится анонсирование (необязательно).	<code>rtt(config-ripng)# timers update <TIME></code>	<TIME> – время в секундах, принимает значения [12..65535]. Значение по умолчанию: 180 секунд.
9	Установить временной интервал корректности маршрутной записи без обновления (необязательно).	<code>rtt(config-ripng)# timers invalid <TIME></code>	<TIME> – время в секундах, принимает значения [12..65535]. Значение по умолчанию: 180 секунд.
10	Установить временной интервал, по истечении которого производится удаление маршрута (необязательно).	<code>rtt(config-ripng)# timers flush <TIME></code>	<TIME> – время в секундах, принимает значения [12..65535]. При установке значения нужно учитывать следующее правило: «timersinvalid + 60» Значение по умолчанию: 240 секунд.
11	Включить анонсирование подсетей.	<code>rtt(config-ripng)# network <IPV6-ADDR/LEN></code>	<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].
12	Добавить фильтрацию подсетей во входящих или исходящих обновлениях (необязательно).	<code>rtt(config-ripng)# prefix-list <PREFIX- LIST-NAME> { in out }</code>	<PREFIX-LIST-NAME> – имя сконфигурированного списка подсетей, задаётся строкой до 31 символа. <ul style="list-style-type: none"> • in – фильтрация входящих маршрутов; • out – фильтрация анонсируемых маршрутов.
13	Включить анонсирование маршрутов, полученных альтернативным способом (необязательно).	<code>rtt(config-ripng)# redistribute static [route-map <NAME>]</code>	<NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых статических маршрутов, задаётся строкой до 31 символа.

Шаг	Описание	Команда	Ключи
		<pre>rtt(config-ripng) # redistribute connected [route-map <NAME>]</pre>	<p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых напрямую подключенных подсетей, задаётся строкой до 31 символа.</p>
		<pre>rtt(config-ripng) # redistribute ospf <ID><ROUTE-TYPE> [route-map <NAME>]</pre>	<p><ID> – номер процесса, может принимать значение [1..65535];</p> <p><ROUTE-TYPE> – тип маршрута:</p> <ul style="list-style-type: none"> • intra - area – анонсирование маршрутов OSPF-процесса в пределах зоны; • inter - area – анонсирование маршрутов OSPF-процесса между зонами; • external 1 – анонсирование внешних маршрутов OSPF-формата 1; • external 2 – анонсирование внешних маршрутов OSPF-формата 2. <p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых OSPF-маршрутов, задаётся строкой до 31 символа.</p>
		<pre>rtt(config-ripng) # redistribute isis <ID><ROUTE-TYPE> [route-map <NAME>]</pre>	<p><ID> – номер процесса, может принимать значение [1..65535];</p> <p><ROUTE-TYPE> – тип маршрута:</p> <ul style="list-style-type: none"> • level-1 – анонсирование маршрутов ISIS-процесса уровня 1; • level-2 – анонсирование маршрутов ISIS-процесса уровня 2; • inter-area – анонсирование межзоновых маршрутов ISIS-процесса. <p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых ISIS-маршрутов, задаётся строкой до 31 символа.</p>

Шаг	Описание	Команда	Ключи
		<pre> rtt(config-ripng) # redistribute bgp <AS> [route-map <NAME>] </pre>	<p><AS> – номер автономной системы, может принимать значения [1..4294967295];</p> <p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых BGP-маршрутов, задаётся строкой до 31 символа.</p>
14	Перейти в режим конфигурирования интерфейса/туннеля/сетевого моста.	<pre> rtt(config) # interface <IF-TYPE><IF-NUM> </pre>	<p><IF-TYPE> – тип интерфейса;</p> <p><IF-NUM> – F/S/P – F-фрейм (1), S – слот (0), P – порт.</p>
		<pre> rtt(config) # tunnel <TUN-TYPE><TUN-NUM> </pre>	<p><TUN-TYPE> – тип туннеля;</p> <p><TUN-NUM> – номер туннеля.</p>
		<pre> rtt(config) # bridge <BR-NUM> </pre>	<p><BR-NUM> – номер bridge.</p>
15	Установить величину метрики RIPng-маршрутов на интерфейсе (необязательно).	<pre> rtt(config-if-gi) # ipv6 rip metric <VALUE> </pre>	<p><VALUE> – величина метрики, задаётся в размере [0..32767].</p> <p>Значение по умолчанию: 5.</p>
16	Включить суммаризацию подсетей (необязательно).	<pre> rtt(config-if-gi) # ipv6 rip summary-address <IPv6-ADDR/LEN> </pre>	<p><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].</p>
17	Включить протокол BFD для протокола RIP (необязательно).	<pre> rtt(config-if-gi) # ipv6 rip bfd-enable </pre>	

12.5.2. Пример настройки

Задача:

Настроить на маршрутизаторах протокол RIPng для обмена маршрутной информацией. Маршрутизаторы должны анонсировать адреса, присвоенные Loopback-интерфейсам.



Решение:

Предварительно нужно настроить IPv6-адреса на интерфейсах согласно схеме сети, приведенной выше.

На первом маршрутизаторе перейдем в режим конфигурирования протокола RIPng и укажем сети, которые будут анонсироваться протоколом:

```
RTT1(config)# ipv6 router rip
RTT1(config-ripng)# network c00:0:1409:3900::1/128
```

На втором маршрутизаторе произведем аналогичные действия:

```
RTT2(config)# ipv6 router rip
RTT2(config-ripng)# network c00:0:1409:3900::2/128
```

Активируем протокол RIPng на обоих маршрутизаторах:

```
RTT1(config-ripng)# enable
RTT2(config-ripng)# enable
```

Проверяем распространение маршрутной информации:

```
RTT1# sh ipv6 route rip
R      * fc00:0:1409:3900::2/128 [100/2]          via fe80::aaf9:4bff:fead:fed2
on gil/0/1 [rip 06:01:33]

RTT2# sh ipv6 route rip
R      * fc00:0:1409:3900::2/128 [100/2]          via fe80::aaf9:4bff:fead:fed1
on gil/0/1 [rip 06:01:33]
```

На этом базовая настройка протокола RIPng закончена.

12.6. Настройка OSPF

OSPF — протокол динамической маршрутизации, основанный на технологии отслеживания состояния канала (link-state technology) и использующий для нахождения кратчайшего пути алгоритм Дейкстры.

12.6.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Настроить приоритетность протокола OSPF-маршрутизации для основной таблицы маршрутизации (необязательно).	<code>rtt(config)# ip protocols ospf preference <VALUE></code>	<VALUE> – приоритетность протокола, принимает значения в диапазоне [1..255]. Значение по умолчанию: 150.
		<code>rtt(config-vrf)# ip protocols ospf preference <VALUE></code>	

Шаг	Описание	Команда	Ключи
2	Настроить емкость таблиц маршрутизации протокола OSPF (необязательно).	<code>rtt(config)# ip protocols ospf max-routes <VALUE></code>	<p><VALUE> – количество маршрутов протокола OSPF в маршрутной таблице, принимает значения в диапазоне:</p> <ul style="list-style-type: none"> • для R800 – [1..500000]; • для R100/200 – [1..300000]. <p>Значение по умолчанию для глобального режима:</p> <ul style="list-style-type: none"> • для R800 – (500000); • для R100/200 – (300000). <p>Значение по умолчанию для VRF: 0.</p>
		<code>rtt(config)# ipv6 protocols ospf max-routes <VALUE></code>	
3	Включить вывод информации о состоянии отношений с соседями для протокола маршрутизации OSPF (необязательно).	<code>rtt(config)# router ospf log-adjacency-changes</code>	
		<code>rtt(config)# ipv6 router ospf log-adjacency-changes</code>	
4	Создать списки IP-подсетей, которые в дальнейшем будут использоваться для фильтрации анонсируемых и получаемых IP-маршрутов (необязательно).	<code>rtt(config)# ip prefix-list <NAME></code>	<p><NAME> – имя конфигурируемого списка подсетей, задаётся строкой до 31 символа.</p>
		<code>rtt(config)# ipv6 prefix-list <NAME></code>	
5	Разрешить (permit) или запретить (deny) списки префиксов (необязательно).	<code>rtt(config-pl)# permit [{ object-group <OBJ-GROUP-NETWORK-NAME> <ADDR/LEN> <IPV6-ADDR/LEN> }] [{ eq <LEN> le <LEN> ge <LEN> le <LEN> }]</code>	<p><OBJ-GROUP-NETWORK-NAME> – имя профиля IPv4/IPv6-адресов, задаётся строкой до 31 символа;</p>

Шаг	Описание	Команда	Ключи
		<pre> rtt(config-pl)# deny [{ object-group <OBJ- GROUP-NETWORK-NAME> <ADDR/LEN> > <IPV6- ADDR/LEN> }] [{ eq <LEN> le <LEN> ge <LEN> [le <LEN>] }] </pre>	<p><ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><LEN> – длина префикса, принимает значения [1..32] в IP-списках префиксов;</p> <ul style="list-style-type: none"> • 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.
6	Добавить OSPF-процесс в систему и осуществить переход в режим настройки параметров OSPF-процесса.	<pre> rtt(config)# router ospf <ID> [vrf <VRF>] </pre> <pre> rtt(config)# ipv6 router ospf <ID> [vrf <VRF>] </pre>	<p><ID> – номер автономной системы процесса, принимает значения [1..65535]</p> <p><VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации.</p>
7	Установить идентификатор маршрутизатора для данного OSPF-процесса.	<pre> rtt(config-ospf)# router-id { <ID> <IF> <TUN> } </pre> <pre> rtt(config-ipv6-ospf)# router-id { <ID> <IF> <TUN> } </pre>	<p><ID> – идентификатор маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p> <p><IF> – интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.</p> <p><TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.</p>
8	Определить приоритетность маршрутов процесса OSPF (необязательно).	<pre> rtt(config-ospf)# preference <VALUE> </pre> <pre> rtt(config-ipv6-ospf)# preference <VALUE> </pre>	<p><VALUE> – приоритетность маршрутов процесса OSPF, принимает значения в диапазоне [1..255].</p>
9	Определить референсное значение для	<pre> rtt(config-ospf)# auto- cost reference bandwidth <VALUE> </pre>	

Шаг	Описание	Команда	Ключи
	автоматического расчёта стоимости (cost) интерфейсов (необязательно).	<code>rtt(config-ipv6-ospf) # auto-cost reference bandwidth <VALUE></code>	<VALUE> – референсное значение для расчета стоимости интерфейса в диапазоне [1..100000000K]. Значение по умолчанию: 100000K.
10	Определить максимальное количество равнозначных маршрутов до цели (необязательно).	<code>rtt(config-ospf) # maximum-path <PATHS></code>	<PATHS> – количество равноценных маршрутов до цели, принимает значения в диапазоне [1..32]. Значение по умолчанию: 16.
		<code>rtt(config-ipv6-ospf) # maximum-path <PATHS></code>	
11	Включить совместимость с RFC 1583 (необязательно).	<code>rtt(config-ospf) # compatible rfc1583</code>	
		<code>rtt(config-ipv6-ospf) # compatible rfc1583</code>	
12	Добавить фильтрацию подсетей во входящих или исходящих обновлениях (необязательно).	<code>rtt(config-ospf) # prefix-list <PREFIX- LIST-NAME> { in out }</code>	<PREFIX-LIST-NAME> – имя сконфигурированного списка подсетей, задаётся строкой до 31 символа. <ul style="list-style-type: none"> • in – фильтрация входящих маршрутов; • out – фильтрация анонсируемых маршрутов.
		<code>rtt(config-ipv6-ospf) # prefix-list <PREFIX- LIST-NAME> { in out }</code>	
13	Включить анонсирование маршрутов, полученных альтернативным способом (необязательно).	<code>rtt(config-ospf) # redistribute static [metric <TYPE> <METRIC>] [route-map <NAME>]</code>	<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2; <METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535]. <NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых статических маршрутов, задаётся строкой до 31 символа.
		<code>rtt(config-ipv6-ospf) # redistribute static [metric <TYPE> <METRIC>] [route-map <NAME>]</code>	
		<code>rtt(config-ospf) # redistribute connected [metric <TYPE> <METRIC>] [route-map <NAME>]</code>	<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2; <METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535]. <NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых напрямую подключенных подсетей, задаётся строкой до 31 символа.
		<code>rtt(config-ipv6-ospf) # redistribute connected [metric <TYPE> <METRIC>] [route-map <NAME>]</code>	

Шаг	Описание	Команда	Ключи
		<pre> rtt(config-ospf)# redistribute rip [metric <TYPE> <METRIC>] [route-map <NAME>] </pre>	<p><TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2;</p> <p><METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535].</p> <p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых RIP-маршрутов, задаётся строкой до 31 символа.</p>
		<pre> rtt(config-ospf)# redistribute isis <ID> <ROUTE-TYPE> [metric <TYPE> <METRIC>] [route-map <NAME>] </pre>	<p><TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2;</p> <p><METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535].</p>
		<pre> rtt(config-ipv6-ospf)# redistribute isis <ID> <ROUTE-TYPE> [route- map <NAME>] </pre>	<p><ID> – номер процесса, может принимать значение [1..65535].</p> <p><ROUTE-TYPE> – тип маршрута:</p> <ul style="list-style-type: none"> • level-1 – аносирование маршрутов ISIS-процесса уровня 1; • level-2 – аносирование маршрутов ISIS-процесса уровня 2; • inter-area – аносирование межзоновых маршрутов ISIS-процесса. <p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых BGP-маршрутов, задаётся строкой до 31 символа.</p>

Шаг	Описание	Команда	Ключи
		<pre> rtt(config-ospf) # redistribute bgp <AS> [metric <TYPE> <METRIC>] [route-map <NAME>] </pre>	<p><TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2;</p> <p><METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535].</p> <p><AS> – номер автономной системы, может принимать значения [1..4294967295];</p> <p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых BGP-маршрутов, задаётся строкой до 31 символа.</p>
14	Активировать OSPF-процесс.	<pre> rtt(config-ospf) # enable rtt(config-ipv6-ospf) # enable </pre>	
15	Создать OSPF-область и перейти в режим конфигурирования области.	<pre> rtt(config-ospf) # area <AREA_ID> rtt(config-ipv6-ospf) # area <AREA_ID> </pre>	<p><AREA_ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p>
16	Включить анонсирование подсетей (необязательно).	<pre> rtt(config-ospf-area) # network <ADDR/LEN> rtt(config-ipv6-ospf- area) # network <IPV6- ADDR/LEN> </pre>	<p><ADDR/LEN> – адрес подсети, указывается в следующем формате:</p> <p>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].</p> <p><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].</p>
17	Определить тип области (необязательно).	<pre> rtt(config-ospf-area) # area-type <TYPE> [no- summary] </pre>	<p><TYPE> – тип области:</p>

Шаг	Описание	Команда	Ключи
		<pre>rtt(config-ipv6-ospf-area)# area-type <TYPE> [no-summary]</pre>	<ul style="list-style-type: none"> stub – устанавливает значение stub (типичная область); no-summary – команда в связке с параметром «stub» образует область «totallystubby» (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 totallynssa (автоматически генерирует маршрут по умолчанию как межобластной).
18	Включить генерацию маршрута по умолчанию для NSSA или stub-области и анонсирование его в качестве Type-7 или Type-3 LSA соответственно (необязательно).	<pre>rtt(config-ospf-area)# default-information-originate</pre> <pre>rtt(config-ipv6-ospf-area)# default-information-originate</pre>	
19	Определить тип метрики маршрута по умолчанию для NSSA-области (необязательно).	<pre>rtt(config-ospf-area)# default-metric-type <TYPE></pre> <pre>rtt(config-ipv6-ospf-area)# default-metric-type <TYPE></pre>	<ul style="list-style-type: none"> type-1 – устанавливает тип метрики E1 для маршрута по умолчанию в NSSA-области; type-2 – устанавливает тип метрики E2 для маршрута по умолчанию в NSSA-области.
20	Активировать OSPF-область.	<pre>rtt(config-ospf-area)# enable</pre> <pre>rtt(config-ipv6-ospf-area)# enable</pre>	
21	Установить виртуальное соединение между основной и удаленными областями, имеющими между ними несколько областей (необязательно).	<pre>rtt(config-ospf-area)# virtual-link <ID></pre> <pre>rtt(config-ipv6-ospf-area)# virtual-link <ID></pre>	<ID> – идентификатор маршрутизатора, с которым устанавливается виртуальное соединение, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
22	Установить интервал времени в секундах, по	<pre>rtt(config-ospf-vlink)# retransmit-interval <TIME></pre>	

Шаг	Описание	Команда	Ключи
	истечения которого маршрутизатор повторно отправит пакет, который не получил подтверждения о получении (необязательно).	<code>rtt(config-ipv6-ospf-vlink)# restransmit-interval <TIME></code>	<p><TIME> – время в секундах, принимает значения [1..65535].</p> <p>Значение по умолчанию: 5 секунд.</p>
23	Установить интервал времени в секундах, по истечении которого маршрутизатор отправляет следующий hello-пакет (необязательно).	<code>rtt(config-ospf-vlink)# hello-interval <TIME></code> <code>rtt(config-bgp-neighbor)# fall-over bfd</code>	<p><TIME> – время в секундах, принимает значения [1..65535].</p> <p>Значение по умолчанию: 10 секунд.</p>
		<code>rtt(config-ipv6-ospf-vlink)# hello-interval <TIME></code>	
24	<p>Установить интервал времени в секундах, по истечении которого сосед будет считаться неактивным (необязательно).</p> <p>Этот интервал должен быть кратным значению «hello-interval».</p>	<code>rtt(config-ospf-vlink)# dead-interval <TIME></code>	<p><TIME> – время в секундах, принимает значения [1..65535]. Для (config-ipv6-ospf-vlink) - [2..65535]</p> <p>Значение по умолчанию: 40 секунд.</p>
		<code>rtt(config-ipv6-ospf-vlink)# dead-interval <TIME></code>	
25	Определяется интервал времени в секундах, по истечении которого маршрутизатор выберет DR в сети (необязательно).	<code>rtt(config-ospf-vlink)# wait-interval <TIME></code>	<p><TIME> – время в секундах, принимает значения [2..65535].</p> <p>Значение по умолчанию: 40 секунд.</p>
		<code>rtt(config-ipv6-ospf-vlink)# wait-interval <TIME></code>	
26	Определить алгоритм аутентификации (необязательно).	<code>rtt(config-ospf-vlink)# authentication algorithm <ALGORITHM></code>	<p><ALGORITHM> – алгоритм аутентификации:</p> <ul style="list-style-type: none"> • cleartext – пароль, передается открытым текстом (доступно только для RIP и OSPF-VLINK); • md 5 – пароль, хешируется по алгоритму md5.
27	Установить пароль для аутентификации с соседом (необязательно).	<code>rtt(config-ospf-vlink)# authentication key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<p><CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов.</p> <p><ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).</p>

Шаг	Описание	Команда	Ключи
28	Определить список паролей для аутентификации через алгоритм хеширования md5 (необязательно).	<code>rtt(config-ospf-vlink)# authentication key chain <KEYCHAIN></code>	<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.
29	Активировать виртуальное соединение (необязательно).	<code>rtt(config-ospf-vlink)# enable</code>	
30	Перейти в режим конфигурирования интерфейса/туннеля/сетевого моста.	<code>rtt(config)# interface <IF-TYPE><IF-NUM></code>	<IF-TYPE> – тип интерфейса; <IF-NUM> – F/S/P – F-фрейм (1), S – слот (0), P – порт.
		<code>rtt(config)# tunnel <TUN-TYPE><TUN-NUM></code>	<TUN-TYPE> – тип туннеля; <TUN-NUM> – номер туннеля.
		<code>rtt(config)# bridge <BR-NUM></code>	<BR-NUM> – номер bridge.
31	Определить принадлежность интерфейса/туннеля/сетевого моста к определенному OSPF-процессу.	<code>rtt(config-if-gi)# ip ospf instance <ID></code>	<ID> – номер процесса, принимает значения [1..65535].
		<code>rtt(config-if-gi)# ipv6 ospf instance <ID></code>	
32	Определить принадлежность интерфейса к определенной области OSPF-процесса.	<code>rtt(config-if-gi)# ip ospf area <AREA_ID></code>	<AREA_ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		<code>rtt(config-if-gi)# ipv6 ospf area <AREA_ID></code>	
33	Включить маршрутизацию по протоколу OSPF на интерфейсе.	<code>rtt(config-if-gi)# ip ospf</code>	
		<code>rtt(config-if-gi)# ipv6 ospf</code>	
34	Включить режим, в котором OSPF-процесс будет игнорировать значение MTU интерфейса во входящих Database Description-пакетах (необязательно).	<code>rtt(config-if-gi)# ip ospf mtu-ignore</code>	
		<code>rtt(config-if-gi)# ipv6 ospf mtu-ignore</code>	
35	Определить алгоритм аутентификации протокола OSPF (необязательно).	<code>rtt(config-if-gi)# ip ospf authentication algorithm <ALGORITHM></code>	<ALGORITHM> – алгоритм аутентификации: <ul style="list-style-type: none"> • cleartext – пароль, передается открытым текстом; • md 5 – пароль, хешируется по алгоритму md5.

Шаг	Описание	Команда	Ключи
36	Установить пароль для аутентификации с OSPF-соседом при передаче пароля открытым текстом (необязательно).	<code>rtt(config-if-gi)# ip ospf authentication key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<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...).
37	Определить список паролей для аутентификации по алгоритму хеширования md5 с соседом (необязательно).	<code>rtt(config-if-gi)# ip ospf authentication key-chain <KEYCHAIN></code>	<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.
38	Определить пропускную способность интерфейса для расчёта стоимости (cost) интерфейса (необязательно).	<code>rtt(config-if-gi)# ip ospf bandwidth <VALUE></code> <code>rtt(config-if-gi)# ipv6 ospf bandwidth <VALUE></code>	<VALUE> – пропускная способность интерфейса, принимает значения [1..100000000K].
39	Определить интервал времени в секундах, по истечении которого маршрутизатор выберет DR в сети (необязательно).	<code>rtt(config-if-gi)# ip ospf wait-interval <TIME></code> <code>rtt(config-if-gi)# ipv6 ospf wait-interval <TIME></code>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 40 секунд.
40	Установить интервал времени в секундах, по истечении которого маршрутизатор повторно отправит пакет, на который не получил подтверждения о получении (необязательно).	<code>rtt(config-if-gi)# ip ospf retransmit-interval <TIME></code> <code>rtt(config-if-gi)# ipv6 ospf retransmit-interval <TIME></code>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 5 секунд.
41	Установить интервал времени в секундах, по истечении которого маршрутизатор отправляет следующий hello-пакет (необязательно).	<code>rtt(config-if-gi)# ip ospf hello-interval <TIME></code> <code>rtt(config-if-gi)# ipv6 ospf hello-interval <TIME></code>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 10 секунд.
42	Установить интервал времени в секундах, по истечении которого сосед будет считаться неактивным (необязательно). Этот интервал должен быть кратным значению hello-interval.	<code>rtt(config-if-gi)# ip dead-interval <TIME></code> <code>rtt(config-if-gi)# ipv6 dead-interval <TIME></code>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 40 секунд.
43		<code>rtt(config-if-gi)# ip poll-interval <TIME></code>	

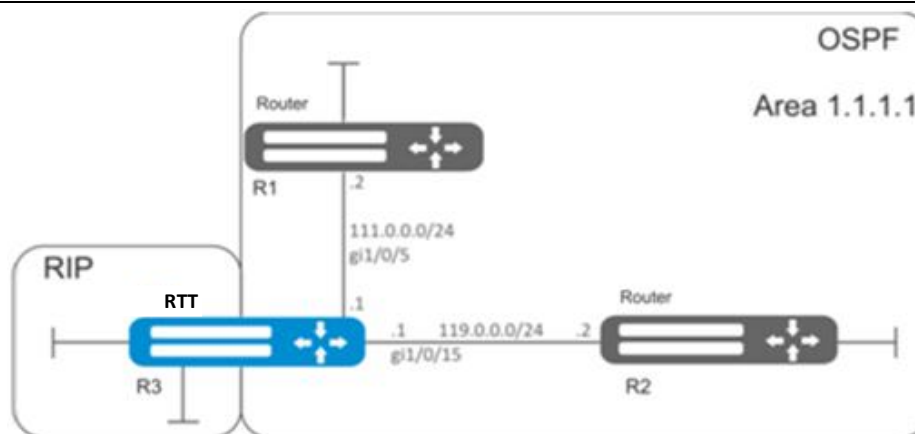
Шаг	Описание	Команда	Ключи
	Установить интервал времени, в течение которого NBMA-интерфейс ждет, прежде чем отправить hello-пакет соседу, даже в случае, если сосед неактивен (необязательно).	<code>rtt(config-if-gi) # ipv6 poll-interval <TIME></code>	<p><TIME> – время в секундах, принимает значения [1 .. 65535].</p> <p>Значение по умолчанию: 120 секунд.</p>
44	Задать статический IP-адрес соседа для установления отношения в NBMA и P2MP (Point-to-MultiPoint) сетях (необязательно).	<code>rtt(config-if-gi) # ip ospf neighbor <IP> [non-eligible]</code>	<p><IP> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p> <p>non-eligible – опциональный параметр, запрещает устройству участвовать в процессе выбора DR в NBMA-сетях. По умолчанию при создании <code>ipv6 ospf neighbor</code> соединение устанавливается в режиме <code>eligible</code>.</p>
		<code>rtt(config-if-gi) # ipv6 ospf neighbor <IP> [non-eligible]</code>	<p><IPV6-ADDR> – IPv6-адрес соседа, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];</p> <p>non-eligible – опциональный параметр, запрещает устройству участвовать в процессе выбора DR в NBMA-сетях. По умолчанию при создании <code>ipv6 ospf neighbor</code> соединение устанавливается в режиме <code>eligible</code>.</p>
45	Определить тип сети для установления OSPF-соседства (необязательно).	<code>rtt(config-if-gi) # ip ospf network <TYPE></code>	<p><TYPE> – тип сети:</p> <ul style="list-style-type: none"> • 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. <p>Значение по умолчанию: <code>broadcast</code>.</p>
		<code>rtt(config-if-gi) # ipv6 ospf network <TYPE></code>	
46		<code>rtt(config-if-gi) # ip ospf passive-interface</code>	

Шаг	Описание	Команда	Ключи
	Перевести интерфейс в пассивный режим работы. В этом режиме не рассылаются hello-пакеты, не устанавливаются отношения соседства, но подключенная сеть анонсируется соседям (необязательно).	<code>rtt(config-if-gi)# ipv6 ospf passive-interface</code>	
47	Установить приоритет маршрутизатора, который используется для выбора DR и BDR (необязательно).	<code>rtt(config-if-gi)# ip ospf priority <VALUE></code>	<VALUE> – приоритет интерфейса, принимает значения [1..65535].
		<code>rtt(config-if-gi)# ipv6 ospf priority <VALUE></code>	Значение по умолчанию: 128.
48	Установить величину метрики на интерфейсе или туннеле (необязательно).	<code>rtt(config-if-gi)# ip ospf cost <VALUE></code>	<VALUE> – величина метрики, задаётся в размере [0..32767].
		<code>rtt(config-if-gi)# ipv6 ospf cost <VALUE></code>	Значение по умолчанию: 10.
49	Включить протокол BFD для протокола OSPF (необязательно).	<code>rtt(config-if-gi)# ip ospf bfd-enable</code>	
		<code>rtt(config-if-gi)# ipv6 ospf bfd-enable</code>	
50	Включить механизм ttl-security hops (необязательно).	<code>rtt(config-if-gi)# ip ospf ttl-security-hops <VALUE></code>	<VALUE> – значение ttl, задаётся в размере [1..255].
		<code>rtt(config-if-gi)# ipv6 ospf ttl-security-hops <VALUE></code>	Значение по умолчанию: 0.

12.6.2. Пример настройки

Задача:

Настроить протокол OSPF на маршрутизаторе для обмена маршрутной информацией с соседними маршрутизаторами. Маршрутизатор должен находиться в области с идентификатором 1.1.1.1 и анонсировать маршруты, полученные по протоколу RIP.



Решение:

Предварительно нужно настроить IP-адреса на интерфейсах согласно схеме, приведенной на рисунке выше.

Создадим OSPF-процесс с идентификатором 10 и перейдем в режим конфигурирования протокола OSPF:

```
rtt(config)# router ospf 10
```

Создадим и включим требуемую область:

```
rtt(config-ospf)# area 1.1.1.1
rtt(config-ospf-area)# enable
rtt(config-ospf-area)# exit
```

Включим анонсирование маршрутной информации из протокола RIP:

```
rtt(config-ospf)# redistribute rip
```

Включим OSPF-процесс:

```
rtt(config-ospf)# enable
rtt(config-ospf)# exit
```

Соседние маршрутизаторы подключены к интерфейсам gi1/0/5 и gi1/0/15. Для установления соседства с другими маршрутизаторами привяжем их к OSPF-процессу и области. Далее включим на интерфейсе маршрутизацию по протоколу OSPF:

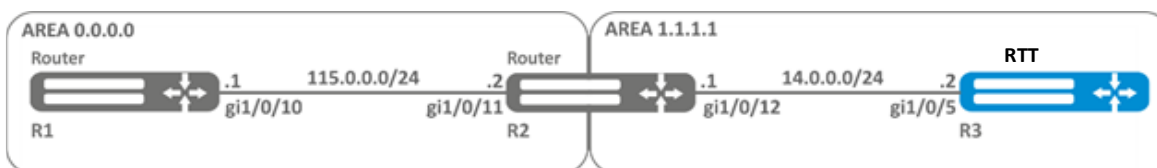
```
rtt(config)# interface gigabitethernet 1/0/5
rtt(config-if-gi)# ip ospf instance 10
rtt(config-if-gi)# ip ospf area 1.1.1.1
rtt(config-if-gi)# ip ospf
rtt(config-if-gi)# exit
rtt(config)# interface gigabitethernet 1/0/15
rtt(config-if-gi)# ip ospf instance 10
rtt(config-if-gi)# ip ospf area 1.1.1.1
rtt(config-if-gi)# ip ospf
```

```
rtt(config-if-gi)# exit
rtt(config)# exit
```

12.6.3. Пример настройки OSPF stub area

Задача:

Изменить тип области 1.1.1.1, область должна быть тупиковой.



Решение:

Предварительно нужно настроить протокол OSPF и IP-адреса на интерфейсах согласно схеме, приведенной на рисунке выше.

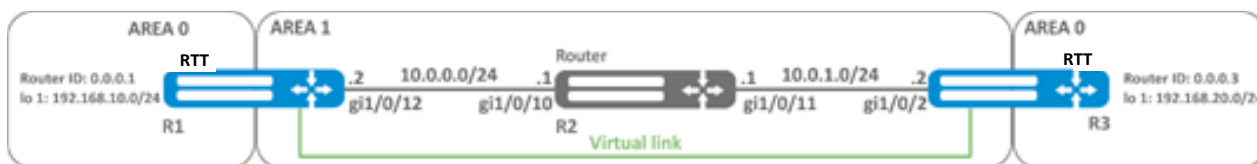
Изменим тип области на тупиковый. На каждом маршрутизаторе из области 1.1.1.1 в режиме конфигурирования области выполним команду:

```
rtt(config-ospf-area)# area-type stub
```

12.6.4. Пример настройки Virtual link

Задача:

Объединить две магистральные области в одну с помощью virtual link.



Решение:



В firewall необходимо разрешить протокол OSPF (89).

Virtual link — это специальное соединение, которое позволяет соединять разорванную на части зону или присоединить зону к магистральной через другую зону. Настраивается между двумя пограничными маршрутизаторами зоны (Area Border Router, ABR).

Предварительно нужно настроить протокол OSPF и IP-адреса на интерфейсах согласно схеме, приведенной на рисунке выше.

На маршрутизаторе R1 перейдем в режим конфигурирования области 1.1.1.1:

```
rtt(config-ospf)# area 1.1.1.1
```

Создадим virtual link с идентификатором 0.0.0.3 и включим его:

```
rtt(config-ospf-area)# virtual-link 0.0.0.3
rtt(config-ospf-vlink)# enable
```

На маршрутизаторе R3 перейдем в режим конфигурирования области 1.1.1.1:

```
rtt(config-ospf)# area 1.1.1.1
```

Создадим virtual link с идентификатором 0.0.0.1 и включим его:

```
rtt(config-ospf-area)# virtual-link 0.0.0.1
rtt(config-ospf-vlink)# enable
```

Рассмотрим таблицу маршрутизации на маршрутизаторе R1:

```
rtt# show ip route
C    * 10.0.0.0/24      [0/0]    dev gil/0/12,          [direct 00:49:34]
O    * 10.0.1.0/24      [150/20] via 10.0.0.1 on gil/0/12,    [ospf1 00:49:53] (0.0.0.3)
O    * 192.168.20.0/24  [150/30] via 10.0.0.1 on gil/0/12, [ospf1 00:50:15] (0.0.0.3)
C    * 192.168.10.0/24 [0/0]    dev lo1,              [direct 21:32:01]
```

Рассмотрим таблицу маршрутизации на маршрутизаторе R3:

```
rtt# show ip route
O    * 10.0.0.0/24      [150/20] via 10.0.1.1 on gil/0/12, [ospf1 14:38:35] (0.0.0.2)
C    * 10.0.1.0/24      [0/0]    dev gil/0/12,          [direct 14:35:34]
C    * 192.168.20.0/24  [0/0]    dev lo1,              [direct 14:32:58]
O    * 192.168.10.0/24 [150/30] via 10.0.1.1 on gil/0/12, [ospf1 14:39:54] (0.0.0.1)
```

Так как OSPF считает виртуальный канал частью области, в таблице маршрутизации R1 маршруты, полученные от R3, отмечены как внутризональные и наоборот.

Для просмотра соседей можно воспользоваться следующей командой:

```
rtt# show ip ospf neighbors 10
```

Таблицу маршрутов протокола OSPF можно просмотреть командой:

```
rtt# show ip ospf 10
```

12.7. Настройка BGP

Протокол BGP предназначен для обмена информацией о достижимости подсетей между автономными системами (далее АС), то есть группами маршрутизаторов под единым техническим управлением, использующими протокол внутридоменной маршрутизации для определения маршрутов внутри себя и протокол междоменной маршрутизации для определения маршрутов доставки пакетов в другие АС. Передаваемая информация включает в себя список АС, к которым имеется доступ через данную систему. Выбор наилучших маршрутов осуществляется исходя из правил, принятых в сети.

12.7.1. Алгоритм настройки



Для установления BGP-сессии необходимо в firewall разрешить TCP-порт 179.

Шаг	Описание	Команда	Ключи
1	Настроить приоритетность протокола BGP-маршрутизации для основной таблицы маршрутизации (необязательно).	<code>rtt(config)# ip protocols bgp preference <VALUE></code>	<p><VALUE> – приоритетность протокола, принимает значения в диапазоне [1..255].</p> <p>Значение по умолчанию: BGP (170).</p>
2	Настроить емкость таблиц маршрутизации протокола BGP (необязательно при использовании глобальной таблицы маршрутизации).	<code>rtt(config)# ip protocols bgp max-routes <VALUE></code> <code>rtt(config)# ipv6 protocols bgp max-routes <VALUE></code> <code>rtt(config-vrf)# ip protocols bgp max-routes <VALUE></code> <code>rtt(config-vrf)# ipv6 protocols bgp max-routes <VALUE></code>	<p><VALUE> – количество маршрутов протокола BGP в маршрутной таблице, принимает значения в диапазоне:</p> <ul style="list-style-type: none"> • для R800 – [1..5000000]; • для R100/200 – [1..2500000]. <p>Значение по умолчанию для глобальной таблицы маршрутизации:</p> <ul style="list-style-type: none"> • для R800 – [5000000]; • для R100/200 – [2500000]. <p>Значение по умолчанию для VRF: 0.</p>
3	Включить вывод информации о состоянии отношений с соседями для протокола маршрутизации BGP (необязательно).	<code>rtt(config)# router bgp log-neighbor-changes</code> <code>rtt(config)# ipv6 router bgp log-neighbor-changes</code>	
4	Включить ESMР и определяется максимальное количество равноценных маршрутов до цели.	<code>rtt(config)# router bgp maximum-paths <VALUE></code>	<p><VALUE> – количество допустимых равноценных маршрутов до цели, принимает значения [1..16].</p>

Шаг	Описание	Команда	Ключи
5	Выбрать метод фильтрации для передаваемой информации между роутерами (обязательно при конфигурировании eBGP для анонсирования подсетей).		
5.1.1	При выборе метода фильтрации на основе route-map создать список правил, который в дальнейшем будет использоваться для фильтрации анонсируемых и получаемых IP-маршрутов.	<code>rtt(config)# route-map <NAME></code>	<NAME> – имя конфигурируемых правил маршрутизации, задаётся строкой до 31 символа.
5.1.2	Создать правило.	<code>(config-route-map)# rule <ORDER></code>	<ORDER> – номер правила, принимает значения [1 .. 10000].
5.1.3	Определить список подсетей, которые затрагиваются правилом.	<code>rtt(config-route-map-rule)#match ip address { <ADDR/LEN> object-group <OBJ-GRP- NETNAME> } [{ eq <LEN> le <LEN> ge <LEN 1> [le <LEN 2>] }]</code>	<ADDR/LEN> – IP-адрес и маска подсети, задается в виде: <ul style="list-style-type: none"> • AAA.BBB.CCC.DDD/EE – IP-адрес подсети с маской в

Шаг	Описание	Команда	Ключи
		<pre> rtt(config-route-map- rule)#match ipv6 address { <IPv6- ADDR/LEN> object-group <OBJ-GRP-NETNAME> } [{ eq <LEN> le <LEN> ge <LEN 1> [le <LEN 2>] }] </pre>	<p>форме префикса, где AAA-DDD принимают значения [0..255] и EE принимает значения [1..32];</p> <p><IPv6-ADDR/LEN> – IPv6-адрес и маска подсети, задается в виде:</p> <ul style="list-style-type: none"> X:X:X:X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128]; <p><OBJ-GRP-NETNAME> – имя профиля IP-адресов, задается строкой до 31 символа. При использовании фильтрации по object-group их необходимо создать заранее;</p> <p><LEN>, <LEN 1>, <LEN 2> – длина префикса, принимает значения [1..32] в IP-списках префиксов для IPv4 и [1..128] для IPv6;</p> <p>eq – при указании команды длина префикса должна соответствовать указанной;</p> <p>le – при указании команды длина префикса должна быть меньше либо соответствовать указанной;</p> <p>ge – при указании команды длина префикса должна быть больше либо соответствовать указанной;</p> <p>ge <LEN 1> le <LEN 2> – при указании команды длина префикса должна быть больше либо соответствовать <LEN>, но меньше или равна <LEN1>.</p>
5.1.4	Разрешить (permit) или запретить (deny) действие для указанных подсетей в правиле.	<pre> rtt(config-route-map- rule)# action {deny permit} </pre>	
5.2.1		<pre> rtt(config)# ip prefix- list <NAME> </pre>	

Шаг	Описание	Команда	Ключи
	При выборе метода фильтрации на основе префикс-листов создать списки IP-подсетей, которые в дальнейшем будут использоваться для фильтрации анонсируемых и получаемых IP-маршрутов.	<pre>rtt(config)# ipv6 prefix-list <NAME></pre>	<NAME> – имя конфигурируемого списка подсетей, задаётся строкой до 31 символа.
5.2.2	Разрешить (permit) или запретить (deny) списки префиксов.	<pre>rtt(config-pl)# permit { <ADDR/LEN> object-group <OBJ-GRP- NETNAME>} [{ eq <LEN> le <LEN> ge <LEN 1> [le <LEN 2>] }] rtt(config-pl)# deny {<ADDR/LEN> object-group <OBJ-GRP- NETNAME>} [{ eq <LEN> le <LEN> ge <LEN 1> [le <LEN 2>] }] rtt(config-ipv6-pl)# permit { <IPV6- ADDR/LEN> object-group <OBJ-GRP-NETNAME>} [{ eq <LEN> le <LEN> ge <LEN 1> [le <LEN 2>] }]</pre>	<p><ADDR/LEN> – IP-адрес и маска подсети, задаётся в виде:</p> <ul style="list-style-type: none"> • 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]; <p><IPV6-ADDR/LEN> – IPv6-адрес и маска подсети, задаётся в виде:</p> <ul style="list-style-type: none"> • X:X:X:X::X/EE, где каждая часть X принимает значения в шестнадцатеричном формате

Шаг	Описание	Команда	Ключи
		<pre> rtt(config-ipv6-pl)# deny {<IPV6-ADDR/LEN> object-group <OBJ-GRP- NETNAME> } [{ eq <LEN> le <LEN> ge <LEN 1> [le <LEN 2>] }] </pre>	<p>[0..FFFF] и EE принимает значения [1..128];</p> <p><OBJ-GRP-NETNAME> – имя профиля IP-адресов, задаётся строкой до 31 символа. При использовании фильтрации по object-group их необходимо создать заранее;</p> <p><LEN>, <LEN 1>, <LEN 2> – длина префикса, принимает значения [1..32] в IP-списках префиксов для IPv4 и [1..128] для IPv6;</p> <p>eq – при указании команды длина префикса должна соответствовать указанной;</p> <p>le – при указании команды длина префикса должна быть меньше либо соответствовать указанной;</p> <p>ge – при указании команды длина префикса должна быть больше либо соответствовать указанной;</p> <p>ge <LEN 1> le <LEN 2> – при указании команды длина префикса должна быть больше либо соответствовать <LEN> но меньше или равна <LEN1>.</p>
6	Добавить BGP-процесс в систему и осуществить переход в режим настройки параметров BGP-процесса.	<pre> rtt(config)# router bgp <AS> </pre>	<p><AS> – номер автономной системы процесса, принимает значения [1..4294967295].</p>
7	Установить идентификатор маршрутизатора.	<pre> rtt(config-bgp)# router- id { <ID> <IF> <TUN> } </pre>	<p><ID> – идентификатор маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p> <p><IF> – интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.</p> <p><TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.</p>

Шаг	Описание	Команда	Ключи
8	Установить идентификатор Route-Reflector кластера, которому принадлежит BGP-процесс маршрутизатора (при необходимости).	<code>rtt(config-bgp) # cluster-id <ID></code>	<ID> – идентификатор Route-Reflector кластера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
9	Включить генерацию и отправку маршрута по умолчанию, если маршрут по умолчанию есть в таблице маршрутизации FIB (необязательно).	<code>rtt(config-bgp) # default-information-originate</code>	
10	Установить временной интервал, по истечении которого идет проверка соединения со встречной стороной (необязательно).	<code>rtt(config-bgp-af) # timers keepalive <TIME></code>	<TIME> – время в секундах, принимает значения [1..32767]. Значение по умолчанию: 60 секунд.
11	Установить временной интервал, по истечении которого встречная сторона считается недоступной (необязательно).	<code>rtt(config-bgp-af) # timers holdtime <TIME></code>	<TIME> – время в секундах, принимает значения [2..65535]. Значение по умолчанию: 180 секунд.
12	Установить время минимальной и максимальной задержки, в течение которого запрещено устанавливать соединение, в целях защиты от частых разрывов соединения (необязательно).	<code>rtt(config-bgp) # timers error-wait <TIME1> <TIME2></code>	<TIME1> – время минимальной задержки в секундах, принимает значения [1..65535]; <TIME2> – время максимальной задержки в секундах, принимает значения [1..65535].
13	Определить глобальный алгоритм аутентификации с соседями (при необходимости).	<code>rtt(config-bgp) # authentication algorithm <ALGORITHM></code>	<ALGORITHM> – алгоритм шифрования: <ul style="list-style-type: none"> md5 – пароль шифруется по алгоритму md5. Значение по умолчанию: шифрование не используется.
14	Установить глобальный пароль для аутентификации с соседями (используется совместно с "authentication algorithm").	<code>rtt(config-bgp) # authentication key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<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...).
15	Активировать BGP-процесс.	<code>rtt(config-bgp) # enable</code>	

Шаг	Описание	Команда	Ключи
16	Определить тип конфигурируемой маршрутной информации и перейти в данный режим настройки.	<code>rtt(config-bgp)# address-family { ipv4 ipv6 } unicast</code>	ipv 4 – семейство IPv4; ipv 6 – семейство IPv6.
17	Включить анонсирование маршрутов процессом BGP полученных альтернативным образом (при необходимости).	<code>rtt(config-bgp-af)# redistribute static [route-map <NAME>]</code>	<NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых статических маршрутов, задаётся строкой до 31 символа.
		<code>rtt(config-bgp-af)# redistribute connected [route-map <NAME>]</code>	<NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых напрямую подключенных подсетей, задаётся строкой до 31 символа.
		<code>rtt(config-bgp-af)# redistribute rip [route-map <NAME>]</code>	<NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых RIP-маршрутов, задаётся строкой до 31 символа.
		<code>rtt(config-bgp-af)# redistribute ospf <ID> <ROUTE-TYPE 1> [<ROUTE- TYPE 2>] [<ROUTE-TYPE 3>] [<ROUTE-TYPE 4>] [route-map <NAME>]</code>	<p><ID> – номер процесса, может принимать значение {1..65535};</p> <p><ROUTE-TYPE> – тип маршрута:</p> <ul style="list-style-type: none"> • intra - area – анонсирование маршрутов OSPF-процесса в пределах зоны; • inter - area – анонсирование маршрутов OSPF-процесса между зонами; • external 1 – анонсирование внешних маршрутов OSPF-формата 1; • external 2 – анонсирование внешних маршрутов OSPF-формата 2; <p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых OSPF-маршрутов, задаётся строкой до 31 символа.</p>

Шаг	Описание	Команда	Ключи
		<code>rtt(config-bgp-af) # redistribute bgp <AS> [route-map <NAME>]</code>	<p><AS> – номер автономной системы, может принимать значения [1..4294967295];</p> <p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых BGP-маршрутов, задаётся строкой до 31 символа.</p>
18	Включить анонсирование подсетей.	<code>rtt(config-bgp-af) # network <ADDR/LEN> [route-map <NAME>]</code>	<p><ADDR/LEN> – адрес подсети, указывается в одном из следующих формате:</p> <ul style="list-style-type: none"> • 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]. <p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых BGP-маршрутов, задаётся строкой до 31 символа.</p>
19	Осуществить выход из режима глобального конфигурирования анонсов маршрутной информации процесса BGP	<code>rtt(config-bgp-af) # exit</code>	
20	Добавить BGP-соседа и осуществить переход в режим настройки параметров BGP-соседа.	<code>rtt(config-bgp) # neighbor <ADDR> <IPv6-ADDR></code>	<p><ADDR> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><IPv6-ADDR> – IPv6-адрес клиента, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].</p>
21	Задать описание соседа (необязательно).	<code>rtt(config-bgp-neighbor) # description <DESCRIPTION></code>	<DESCRIPTION> – описание соседа, задаётся строкой до 255 символов.

Шаг	Описание	Команда	Ключи
22	Установить временной интервал, по истечении которого идет проверка соединения со встречной стороной (необязательно).	<code>rtt(config-bgp-neighbor)# timers keepalive <TIME></code>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 60 секунд.
23	Установить временной интервал, по истечении которого встречная сторона считается недоступной (необязательно).	<code>rtt(config-bgp-neighbor)# timers holdtime <TIME></code>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 180 секунд.
24	Установить время минимальной и максимальной задержки, в течение которого запрещено устанавливать соединение, в целях защиты от частых разрывов соединения (необязательно).	<code>rtt(config-bgp)# timers error-wait <TIME1> <TIME2></code>	<TIME1> – время минимальной задержки в секундах, принимает значения [1..65535]; <TIME2> – время максимальной задержки в секундах, принимает значения [1..65535]. Значение по умолчанию: 60 и 300 секунд.
25	Установить номер автономной системы BGP-соседа.	<code>rtt(config-bgp-neighbor)# remote-as <AS></code>	<AS> – номер автономной системы, принимает значения [1..4294967295].
26	Разрешить подключение к соседям, которые находятся не в напрямую подключенных подсетях (необязательно).	<code>rtt(config-bgp-neighbor)# ebgp-multihop <NUM></code>	<NUM> – максимальное количество хопов при установке EBGП (используется для TTL).
27	Указать, что BGP-сосед является Route-Reflector клиентом (необязательно).	<code>rtt(config-bgp-neighbor)# route-reflector-client</code>	

Шаг	Описание	Команда	Ключи
28	Задать IP/IPv6-адрес маршрутизатора, который будет использоваться в качестве IP/IPv6-адреса источника в отправляемых обновлениях маршрутной информации BGP (необязательно).	<pre>rtt(config-bgp-neighbor)# update-source { <ADDR> <IPv6-ADDR> <IF> <TUN> }</pre>	<p><ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><IPv6-ADDR> – IPv6-адрес источника, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].</p> <p><IF> – интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.</p> <p><TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.</p>
29	Включить режим, в котором разрешен приём маршрутов в BGP-атрибуте, AS Path которых содержит номера автономной системы процесса (необязательно).	<pre>rtt(config-bgp-neighbor)# allow-local-as <NUMBER></pre>	<NUMBER> – пороговое число вхождений номера автономной системы процесса в атрибуте AS Path, при которых маршрут будет принят, диапазон допустимых значений [1..10].
30	Включить BFD-протокол на конфигурируемом BGP-соседе (необязательно, используется совместно с параметром update-source).	<pre>rtt(config-bgp-neighbor)# fall-over bfd</pre>	
31	Включить режим, при котором соседство BGP разрывается, как только указанный маршрут к соседу удаляется из таблицы маршрутизации.	<pre>rtt(config-bgp-neighbor)# fall-over route-map [route-map <NAME>]</pre>	<NAME> – имя маршрутной карты, которая будет использоваться для отслеживания наличия маршрута, задаётся строкой до 31 символа.
32	Определить алгоритм аутентификации с соседом (необязательно).	<pre>rtt(config-bgp-neighbor)# authentication algorithm <ALGORITHM></pre>	<p><ALGORITHM> – алгоритм шифрования:</p> <p>md5 – пароль шифруется по алгоритму md5.</p>

Шаг	Описание	Команда	Ключи
33	Установить пароль для аутентификации с соседом (необязательно).	<code>rtt(config-bgp-neighbor)# authentication key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<p><CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов;</p> <p><ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).</p>
34	Сделать соседство активным.	<code>rtt(config-bgp-neighbor)# enable</code>	
35	Определить тип конфигурируемой маршрутной информации соседа и перейти в данный режим настройки.	<code>rtt(config-bgp-neighbor)# address-family { ipv4 ipv6 vpnv4 } unicast</code>	<p>ipv 4 – семейство IPv4;</p> <p>ipv 6 – семейство IPv6;</p> <p>vpnv4 – семейство VPNv4.</p>
36	При выборе режима фильтрации на основе префикс-листов добавить фильтрацию подсетей во входящих или исходящих обновлениях (обязательно при конфигурировании eBGP для анонсирования подсетей).	<code>rtt(config-bgp-neighbor-af)# prefix-list <PREFIX-LIST-NAME> { in out }</code>	<p><PREFIX-LIST-NAME> – имя сконфигурированного списка подсетей, задаётся строкой до 31 символа.</p> <p>in – фильтрация входящих маршрутов;</p> <p>out – фильтрация анонсируемых маршрутов.</p>
37	Задать режим, в котором BGP-соседу в обновлении наряду с другими маршрутами всегда отправляется маршрут по умолчанию. (необязательно, отсутствует для vpnv4).	<code>rtt(config-bgp-neighbor-af)# default-originate</code>	
38	Задать режим, в котором все обновления отправляются BGP-соседу с указанием в качестве next-hop IP-адреса исходящего интерфейса локального маршрутизатора. По умолчанию изменяет next-hop только eBGP-маршрутов (необязательно, отсутствует для vpnv4).	<code>rtt(config-bgp-neighbor-af)# next-hop-self [all]</code>	all – заменить next-hop для eBGP-, iBGP-маршрутов.
39	Определить приоритетность маршрутов, получаемых от соседа (необязательно).	<code>rtt(config-bgp-neighbor-af)# preference <VALUE></code>	<p><VALUE> – приоритетность маршрутов соседа, принимает значения в диапазоне [1..255].</p> <p>Значение по умолчанию: 170.</p>

Шаг	Описание	Команда	Ключи
40	Задать режим, в котором перед отправлением обновления из BGP-атрибута AS Path маршрутов удаляются частные номера автономных систем (в соответствии с RFC 6996) (необязательно, отсутствует для vrpv4).	<code>rtt(config-bgp-neighbor-af)# remove-private-as [{ all nearest replace }]</code>	<p>all – удалить все частные номера AS из AS-path;</p> <p>nearest – заменить ближайшие частные AS в AS-path на рядом стоящую публичную AS;</p> <p>replace – заменить все частные номера AS номером текущего процесса BGP.</p> <p>Значение по умолчанию: all.</p>
41	Включить обмен маршрутной информацией.	<code>rtt(config-bgp-neighbor-af)# enable</code>	
42	Задать режим, в котором маршрутизатор будет представляться указанным номером автономной системы вместо реального номера автономной системы (необязательно).	<code>rtt(config-bgp-neighbor)# local-as <AS></code>	<AS> – номер автономной системы, принимает значения [1..4294967295].
43	Не добавлять указанный в local-as номер автономной системы в AS-Path при приеме маршрута (необязательно).	<code>rtt(config-bgp-local-as)# no-prepend</code>	
44	Добавлять в AS-Path только номер автономной системы, указанный в local-as (необязательно).	<code>rtt(config-bgp-local-as)# replace-as</code>	
45	Включить агрегирование маршрутной информации (необязательно).	<code>rtt(config-bgp)# aggregate-address { <ADDR/LEN> <IPV6-ADDR/LEN> }</code>	<p><ADDR/LEN> – IP-адрес и маска подсети, задается в виде:</p> <ul style="list-style-type: none"> 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]; <p><IPV6-ADDR/LEN> – IPv6-адрес и маска подсети, задается в виде:</p> <ul style="list-style-type: none"> X:X:X:X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128];

Шаг	Описание	Команда	Ключи
46	Задать route-map для установки дополнительных условий агрегирования маршрутов (необязательно).	<code>rtt(config-bgp-aggregate)# advertise-map [route-map <NAME>]</code>	<NAME> – имя маршрутной карты, которая будет использоваться для задания условий агрегирования BGP-маршрутов, задаётся строкой до 31 символа.
47	Добавлять в AS-Path агрегированного маршрута номера автономных систем из AS-Path его компонентов (необязательно).	<code>rtt(config-bgp-aggregate)# as-set</code>	
48	Задать route-map для установки дополнительных атрибутов агрегированного маршрута (необязательно).	<code>rtt(config-bgp-aggregate)# attribute-map [route-map <NAME>]</code>	<NAME> – имя маршрутной карты, которая будет использоваться для задания атрибутов агрегированного BGP-маршрута, задаётся строкой до 31 символа.
49	Подавлять все компоненты агрегированного маршрута (необязательно).	<code>rtt(config-bgp-aggregate)# summary-only</code>	
50	Задать route-map для подавления компонентов агрегированного маршрута (необязательно).	<code>rtt(config-bgp-aggregate)# suppress-map [route-map <NAME>]</code>	<NAME> – имя маршрутной карты, которая будет использоваться для задания подавляемых компонентов агрегированного BGP-маршрута, задаётся строкой до 31 символа.
51	Задать возможность динамически устанавливать BGP-сессию без указания конкретного адреса соседа. Соседство может быть установлено с любым адресом, попадающим в указанную подсеть (необязательно).	<code>rtt(config-bgp-aggregate)# listen-range { <ADDR/LEN> <IPV6-ADDR/LEN> }</code>	<p><ADDR/LEN> – IP-адрес и маска подсети, задается в виде:</p> <ul style="list-style-type: none"> 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]; <p><IPV6-ADDR/LEN> – IPv6-адрес и маска подсети, задается в виде:</p> <ul style="list-style-type: none"> X:X:X:X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].
52	Разрешить возможность динамической установки BGP-сессии только с соседями, которые имеют определённые номера AS.	<code>rtt(config-bgp-listen)# as-range <AS-PATH></code>	<AS-PATH> – список номеров автономных систем, задается в виде AS-AS,AS,AS-AS, принимает значения [1..4294967295].

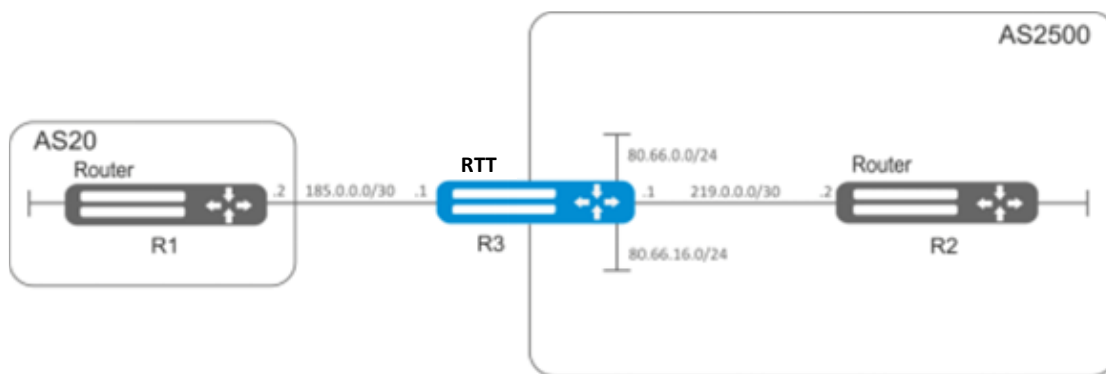
Часто бывает, особенно при конфигурировании iBGP, что в одном bgp-процессе необходимо настроить несколько bgp neighbor с одинаковыми параметрами. Во избежание избыточности конфигурации рекомендуется использовать bgp peer-group, в которой возможно описать общие параметры, а в конфигурации bgp neighbor просто указать причастность к bgp peer-group.

12.7.2. Пример настройки

Задача:

Настроить BGP-протокол на маршрутизаторе R3 со следующими параметрами:

- собственные подсети: 80.66.0.0/24, 80.66.16.0/24;
- анонсирование подсетей, подключенных напрямую;
- собственная AS 2500;
- первое соседство – подсеть 219.0.0.0/30, собственный IP-адрес 219.0.0.1, IP-адрес соседа 219.0.0.2, AS2500;
- второе соседство – подсеть 185.0.0.0/30, собственный IP-адрес 185.0.0.1, IP-адрес соседа 185.0.0.2, AS20.



Решение:

Сконфигурируем необходимые сетевые интерфейсы:

```
rtt-R3(config)# interface gigabitethernet 1/0/1
rtt-R3(config-if-gi)# ip address 185.0.0.1/30
rtt-R3(config-if-gi)# exit
rtt-R3(config)# interface gigabitethernet 1/0/2
rtt-R3(config-if-gi)# ip address 219.0.0.1/30
rtt-R3(config-if-gi)# exit
rtt-R3(config)# interface gigabitethernet 1/0/3
rtt-R3(config-if-gi)# ip address 80.66.0.1/24
rtt-R3(config-if-gi)# exit
rtt-R3(config)# interface gigabitethernet 1/0/4
rtt-R3(config-if-gi)# ip address 80.66.16.1/24
rtt-R3(config-if-gi)# exit
```

Сконфигурируем firewall для приема маршрутизатором BGP-трафика из зоны безопасности WAN:

```
rtt-R3(config)# object-group service og_bgp
```

```
rtt-R3(config-object-group-service)# port-range 179
rtt-R3(config-object-group-service)# exit
rtt-R3(config)# security zone wan
rtt-R3(config-zone)# exit
rtt-R3(config)# security zone-pair wan self
rtt-R3(config-zone-pair)# rule 100
rtt-R3(config-zone-pair-rule)# match protocol tcp
rtt-R3(config-zone-pair-rule)# match destination-port object-group og_bgp
rtt-R3(config-zone-pair-rule)# action permit
rtt-R3(config-zone-pair-rule)# enable
rtt-R3(config-zone-pair-rule)# exit
rtt-R3(config-zone-pair)# exit
```

И укажем принадлежность интерфейсов к зоне безопасности:

```
rtt-R3(config)# interface gigabitethernet 1/0/1
rtt-R3(config-if-gi)# security-zone wan
rtt-R3(config-if-gi)# exit
rtt-R3(config)# interface gigabitethernet 1/0/2
rtt-R3(config-if-gi)# security-zone wan
rtt-R3(config-if-gi)# exit
```

Создадим route-map, который будет использоваться в дальнейшем при настройке разрешающих анонсов роутерам из другой AS:

```
rtt-R3(config)# route-map bgp-general
rtt-R3(config-route-map)# rule 1
rtt-R3(config-route-map-rule)# match ip address 80.66.0.0/24
rtt-R3(config-route-map-rule)# action permit
rtt-R3(config-route-map-rule)# exit
rtt-R3(config-route-map)# rule 2
rtt-R3(config-route-map-rule)# match ip address 80.66.16.0/24
rtt-R3(config-route-map-rule)# action permit
rtt-R3(config-route-map-rule)# exit
rtt-R3(config-route-map)# exit
```

Создадим BGP процесс для AS 2500 и войдем в режим конфигурирования параметров процесса:

```
rtt(config)# router bgp 2500
```

Сконфигурируем анонсирование подсетей, подключенных напрямую:

```
rtt-R3(config-bgp)# address-family ipv4 unicast
rtt-R3(config-bgp-af)# redistribute connected
rtt-R3(config-bgp-af)# exit
```

Создадим соседство с роутером R2 по iBGP:

```
rtt-R3(config-bgp)# neighbor 219.0.0.2
rtt-R3(config-bgp-neighbor)# remote-as 2500
rtt-R3(config-bgp-neighbor)# enable
```

И включим обмен IPv4-маршрутами:

```
rtt-R3(config-bgp-neighbor)# address-family ipv4 unicast
rtt-R3(config-bgp-neighbor-af)# enable
rtt-R3(config-bgp-neighbor-af)# exit
rtt-R3(config-bgp-neighbor)# exit
```

Создадим соседство с роутером R1 по eBGP:

```
rtt-R3(config-bgp)# neighbor 185.0.0.2
rtt-R3(config-bgp-neighbor)# remote-as 20
rtt-R3(config-bgp-neighbor)# enable
```

И включим обмен IPv4-маршрутами, разрешив необходимые маршруты для анонса при помощи заранее подготовленного route-map:

```
rtt-R3(config-bgp-neighbor)# address-family ipv4 unicast
rtt-R3(config-bgp-neighbor-af)# route-map bgp-general out
rtt-R3(config-bgp-neighbor-af)# enable
rtt-R3(config-bgp-neighbor-af)# exit
rtt-R3(config-bgp-neighbor)# exit
```

Включим работу протокола:

```
rtt-R3(config-bgp)# enable
rtt-R3(config-bgp)# exit
```

Информацию о BGP-пирах можно посмотреть командой:

```
rtt# show bgp neighbors
```

Таблицу маршрутов протокола BGP можно просмотреть с помощью команды:

```
rtt# show bgp ipv4 unicast
```

12.7.3. Политика выбора лучшего маршрута в протоколе BGP

В процессе работы BGP обычно вычисляет один лучший маршрут до каждой полученной подсети. Если нет более приоритетного маршрута, полученного при помощи другого протокола маршрутизации до этой подсети, то маршрут устанавливается в таблицу маршрутизации.



Если включен механизм ECMP (`router bgp maximum-paths ..`), то в таблицу маршрутизации могут попасть до 16 активных маршрутов до одной подсети.

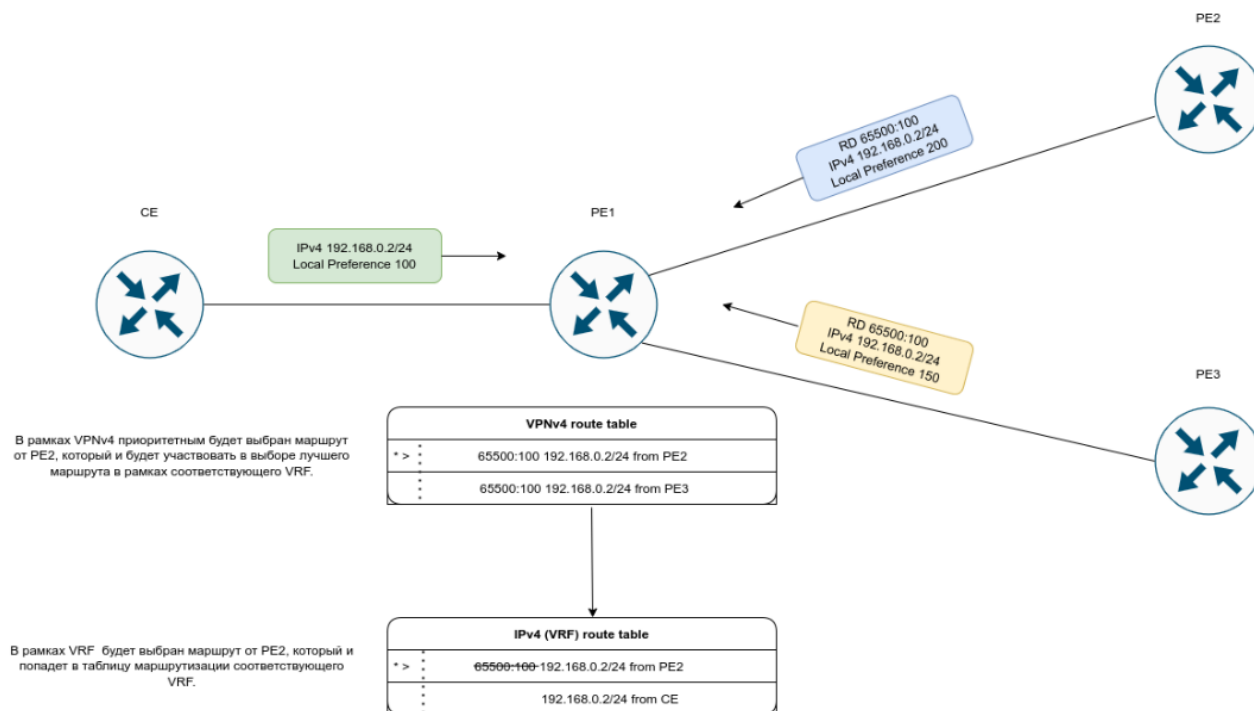
При анонсировании BGP пирам будут использоваться атрибуты лучшего маршрута.

Ниже приведен алгоритм выбора лучшего маршрута в протоколе BGP:

Алгоритм применяется для следующих address family: unicast IPv4, unicast IPv6, VPNv4 unicast, VPLS.

Для VPNv4 маршрутов выбор лучшего маршрута происходит следующим образом:

Сначала выбор лучшего маршрута происходит в рамках **своего RD**, затем в рамках VRF, куда он попадет в соответствии со своим RT.



Прежде всего проверяется доступность next-hop-a у маршрута. Next-hop считается доступным, если до него можно определить connected-маршрут.

1. Маршрут, помеченный как «stale», является менее приоритетным, чем маршрут без таковой метки. Маршрут помечается как «stale» в процессе работы технологии LLGR;
2. Сравнивается значение атрибута Weight – лучшим становится маршрут, имеющий большее значение;
3. Сравнивается значение атрибута Local preferences – лучшим становится маршрут, имеющий большее значение;
4. Сравнивается длина AS-path – маршрут с меньшим количеством hop-ов становится лучшим;
5. Сравнивается значение атрибута Origin – IGP является самым приоритетным. EGP приоритетнее, чем Incomplete;
6. Для маршрутов, принятых от одной и той же автономной системы, сравнивается значение атрибута multiple exit discriminator (MED) – наименьшее значение атрибута имеет больший приоритет;
7. Маршрут, полученный от EBGP-пира, имеет больший приоритет по сравнению с маршрутом, полученным от IBGP-пира;
8. Сравнивается IGP-метрика сети, через который доступен маршрут – наименьшее значение имеет больший приоритет;

8.1 Если включен ESMР, то дальнейших сравнений не производится и маршрут (multipath) попадет в таблицу маршрутизации;

9. Сравнивается параметр Router-Id – маршрут, полученный от BGP-соседа с наименьшим Router-Id, является приоритетным. При наличии атрибута Originator ID будет учитываться Router-id источника маршрута;

10. Сравнивается количество адресов в Cluster list – маршрут, имеющий наименьшее количество адресов, становится лучшим;

11. Сравниваются адреса BGP-пиров – маршрут, полученный от BGP-пира с наименьшим из адресов, является приоритетным.

В выводе маршрутной информации для определенного префикса лучший маршрут будет отмечен как «Best»:

```
RTT# show bgp ipv4 unicast 192.0.2.0/24
192.0.2.0/24 via 100.64.28.1 on gil/0/1.2800 [bgp65514 2022-05-22] (65041i)
  Administrative Distance: 170
  Type: unicast
  Origin: IGP
  AS PATH: 65054 65055 65056 65077 65098 65059
  Next Hop: 100.64.28.1
  Local Preference: 100
  Community: (3356:2) (3356:22) (3356:86) (3356:501) (3356:666)
              (3356:903) (3356:2065)
              (12389:6) (65000:64990)
  Weight: 0
  Valid
192.0.2.0/24 via 101.7.0.1 on gil/0/1.2800 [bgp65514 2022-05-22] (65041i)
  Administrative Distance: 170
  Type: unicast
  Origin: IGP
  AS PATH: 65020 65030
  Next Hop: 101.7.0.1
  Local Preference: 200
  Community: (3356:2) (3356:22) (3356:86) (3356:501) (3356:666)
              (3356:903) (3356:2065)
              (12389:6) (65000:64990)
  Weight: 0
  Valid,Best
```

12.7.4. Условное анонсирование маршрутной информации (Conditional Advertisement)

В обычных сценариях BGP анонсирует все лучшие маршруты из своей BGP RIB. Иногда необходимо более гибкое управление анонсируемой маршрутной информацией. В этом случае рекомендуется использование функции Conditional advertisement, которая позволяет описать условия, при совпадении которых будет анонсироваться (или наоборот отзываться) необходимая маршрутная информация.



В текущей реализации функционал поддерживан для IPv4 (AFI-1, SAFI-1), IPv6 (AFI-2, SAFI-1) маршрутов.



Реализована поддержка как для GRT, так и в VRF.

Для работы Conditional advertisement необходимо выполнить следующие шаги:

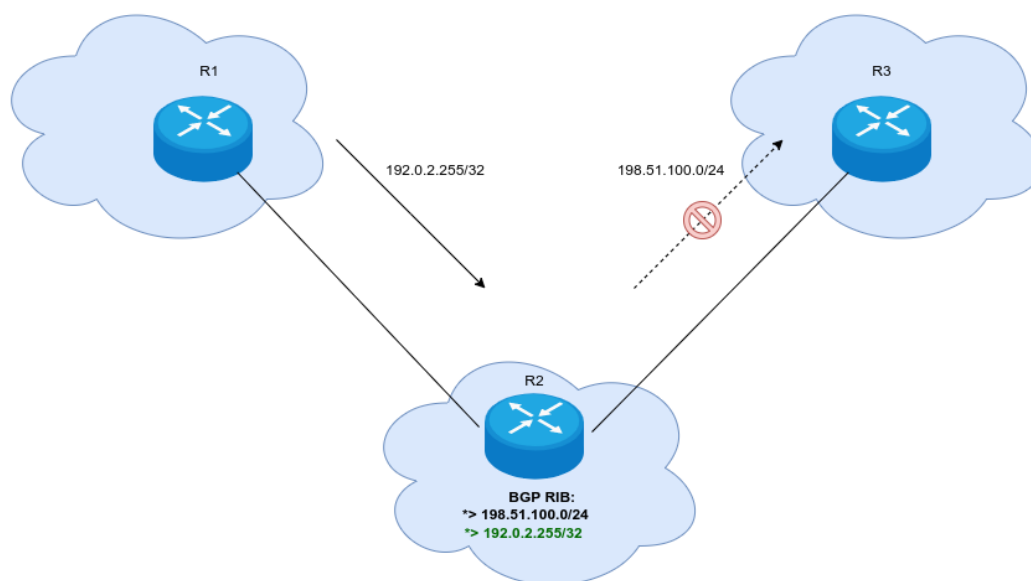
1. Описать маршрутные карты condition-map и advertise-map:

- Condition-map – это карта, в которой необходимо описать маршрутную информацию для проверки. Планировщик будет запускаться каждые 60 секунд для проверки наличия в BGP RIB маршрутной информации, описанной в этой карте;
- Advertise-map – это карта, в который необходимо описать маршрутную информацию, которая будет анонсироваться при выполнении условий, описанных в condition-map.

2. В контексте настройки BGP-соседа необходимо задать условие, при котором будет анонсироваться маршрутная информация, описанная в advertise-map. Рассмотрим этот пункт на примере ниже:

Условие EXIST-MAP:

- Если R2 содержит в BGP RIB маршрут 192.0.2.255/32, то R2 анонсирует в сторону R3 маршрут 198.51.100.0/24 (пример на рисунке ниже);
- Если R2 не содержит в BGP RIB маршрут 192.0.2.255/32, то анонсирование маршрута 198.51.100.0/24 соседу R3 не происходит.

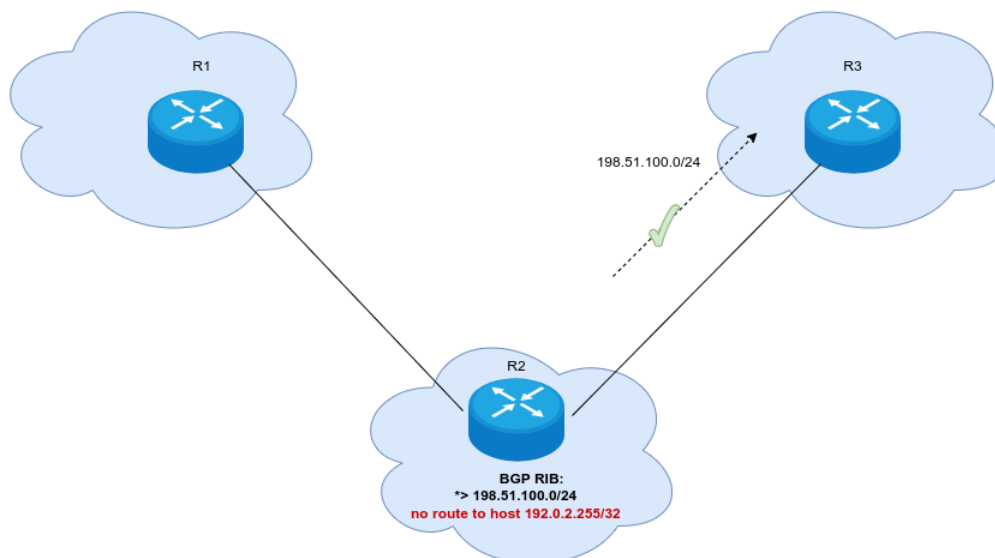


```
route-map CONDITION
  rule 1
    match ip address 192.0.2.255/32
  exit
exit
route-map ADVERTISE
```

```
rule 1
  match ip address 198.51.100.0/24
exit
router bgp 65540
  neighbor R3
    description "To R3"
    address-family ipv4 unicast
      advertise-map ADVERTISE exist-map CONDITION
    enable
  exit
```

Условие not EXIST-MAP:

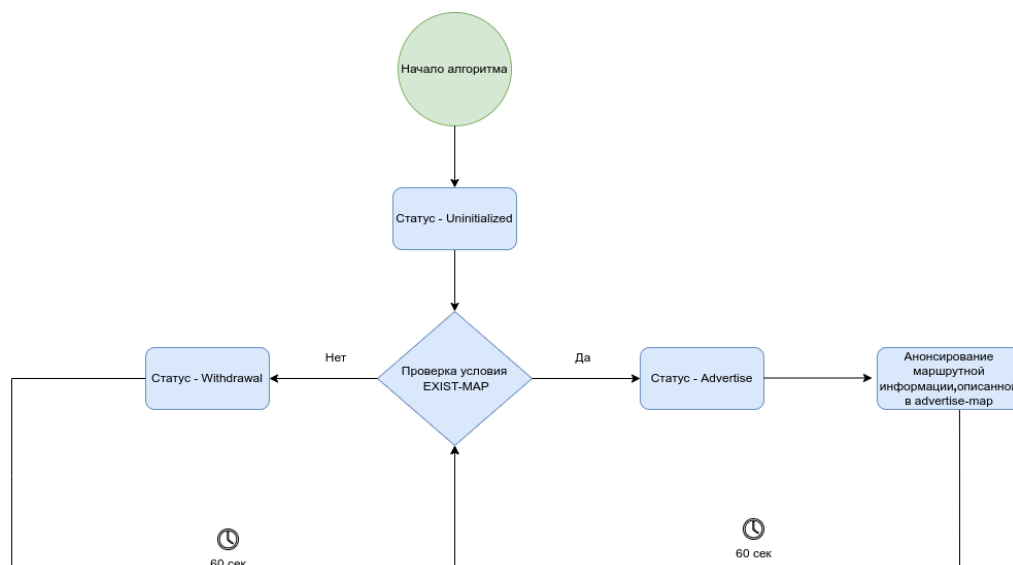
- Если R2 содержит в BGP RIB маршрут 192.0.2.255/32, то анонсирование маршрута 198.51.100.0/24 соседу R3 не происходит;
- Если R2 не содержит в BGP RIB маршрут 192.0.2.255/32, то R2 анонсирует в сторону R3 маршрут 198.51.100.0/24 (пример на рисунке ниже).



```
route-map CONDITION
  rule 1
    match ip address 192.0.2.255/32
  exit
exit
route-map ADVERTISE
  rule 1
    match ip address 198.51.100.0/24
  exit
exit
router bgp 65540
  neighbor R3
    description "To R3"
    address-family ipv4 unicast
      advertise-map ADVERTISE not exist-map CONDITION
    enable
```

exit

Ниже приведена диаграмма состояний для условия EXIST-MAP:



После активации функции Conditional advertisement находится в состоянии «Uninitialized». На этом этапе анонсируется вся разрешенная маршрутная информация, происходит инициализация планировщика для дальнейшей работы. Время нахождения в этом состоянии – 60 секунд:

```

vRTT# sh bgp neighbors
BGP neighbor is 192.0.2.2
  BGP state:                Established
  Type:                     Static neighbor
  Neighbor address:         192.0.2.2
  Neighbor AS:              202766
  Neighbor ID:              192.0.2.2
  Neighbor caps:             refresh enhanced-refresh restart-aware
AS4
  Session:                  external AS4
  Source address:           192.0.2.1
  Weight:                   0
  Hold timer:               136/180
  Keepalive timer:          35/60
  Address family ipv4 unicast:
    Send-label:              No
    Default originate:       No
    Default information originate: No
    Incoming route-map:      IN
    Outgoing route-map:      OUT
    Advertise-map:            ADVERTISE
    Condition-map:            CONDITION
    Conditional advertisement status: Uninitialized <-----
  Uptime:                   12 s
  
```

Далее планировщик проверяет условие EXIST-MAP для соответствующей condition-map. Если условие истинно, происходит анонсирование (обновление) маршрутной информации в соответствии с правилами, заданными в advertise-map. Состояние статуса меняется на «Advertise». Время нахождения в этом состоянии – 60 секунд:

```
vrtt# sh bgp neighbors
BGP neighbor is 192.0.2.2
  BGP state:                Established
  Type:                     Static neighbor
  Neighbor address:         192.0.2.2
  Neighbor AS:              202766
  Neighbor ID:              192.0.2.2
  Neighbor caps:            refresh enhanced-refresh restart-aware
AS4
  Session:                  external AS4
  Source address:           192.0.2.1
  Weight:                   0
  Hold timer:               136/180
  Keepalive timer:         41/60
  Address family ipv4 unicast:
    Send-label:             No
    Default originate:      No
    Default information originate: No
    Incoming route-map:     IN
    Outgoing route-map:     OUT
    Advertise-map:          ADVERTISE
    Condition-map:          CONDITION
    Conditional advertisement status: Advertise    <----
  Uptime:                   1119 s
```

Если условие EXIST-MAP для соответствующей condition-map не выполняется, происходит отзыв маршрутной информации, описанной в соответствующей advertise-map. Состояние статуса меняется на «Withdrawal». Время нахождения в этой стадии – 60 секунд:

```
vRTT# sh bgp neighbors
BGP neighbor is 192.0.2.2
  BGP state:                Established
  Type:                     Static neighbor
  Neighbor address:         192.0.2.2
  Neighbor AS:              202766
  Neighbor ID:              192.0.2.2
  Neighbor caps:            refresh enhanced-refresh restart-aware
AS4
  Session:                  external AS4
  Source address:           192.0.2.1
  Weight:                   0
  Hold timer:               136/180
  Keepalive timer:         41/60
  Address family ipv4 unicast:
    Send-label:             No
    Default originate:      No
    Default information originate: No
    Incoming route-map:     IN
    Outgoing route-map:     OUT
    Advertise-map:          ADVERTISE
    Condition-map:          CONDITION
```

```
Conditional advertisement status: Withdrawal    <----  
Uptime:                                     1119 s
```

Порядок выполнения политик фильтрации маршрутной информации:

1. Выполняется политика, заданная при редистрибуции маршрутов (AF_POLICY_OUT);
2. Применяется advertise-map, описанная в Conditional advertisement (advertise-map ADVERTISE...);
3. Обрабатывается политика фильтрации исходящей маршрутной информации (route-map OUT out).

```
route-map ADVERTISE  
  rule 1  
    match ip address 10.100.0.255/32  
    action set local-preference 101  
    action set metric bgp 78  
  exit  
exit  
route-map OUT  
  rule 1  
    action set local-preference 200  
  exit  
exit  
route-map CONDITION  
  rule 1  
    match ip address 10.100.0.255/32  
  exit  
exit  
route-map AF_POLICY_OUT  
  rule 1  
    match ip address 10.100.0.255/32  
    action set community 65:65  
  exit  
exit  
router bgp 64512  
  neighbor 192.0.2.2  
  remote-as 64512  
  address-family ipv4 unicast  
    route-map OUT out <----- 3  
    advertise-map ADVERTISE exist-map CONDITION <----- 2  
  enable  
  exit  
  enable  
exit  
address-family ipv4 unicast  
  redistribute static route-map AF_POLICY_OUT <----- 1  
exit  
enable  
exit
```

// Вывод атрибутов BGP маршрута после прохождения всех политик:

```
show bgp ipv4 unicast 10.100.0.255/32  
Administrative Distance: 170  
Type: unicast  
Origin: Incomplete
```

```
AS path: --
Next Hop: 192.168.1.1
Output Label: --
Input Label: imp-null
Local Preference: 200
MED: 78
Cluster List: --
Community: 65:65
EXT Community: --
Weight: --
```

12.7.4.1. Алгоритм настройки

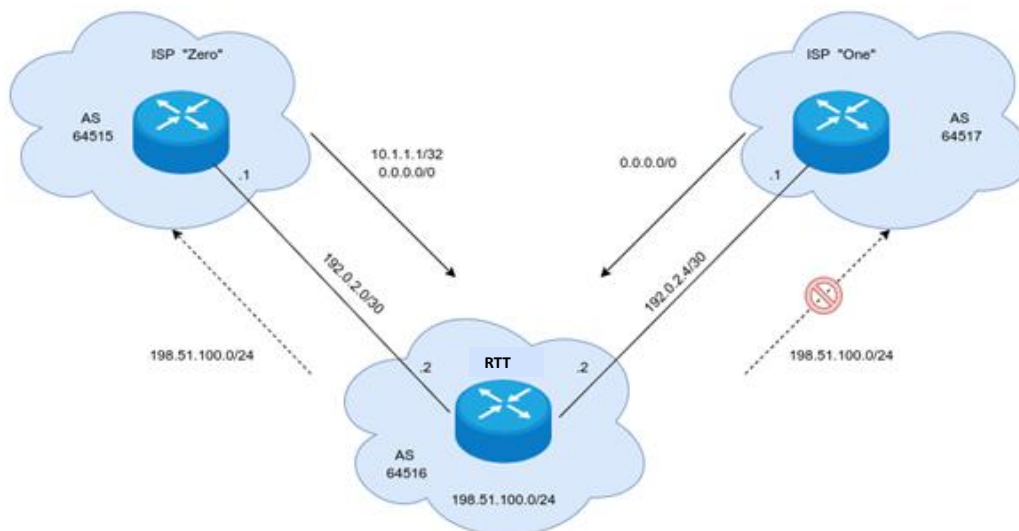
Шаг	Описание	Команда	Ключ
1	Настроить протокол BGP (см. раздел Настройка BGP).		
2	Создать advertise-map, описав в нем список подсетей для дальнейшего анонсирования.	<code>rtt(config) # route-map <ADVERTISE></code>	<ADVERTISE> – имя конфигурируемых правил маршрутизации, задаётся строкой до 31 символа.
3	Создать condition-map, описав в нем список подсетей по которым будет осуществляться проверка.	<code>rtt(config) # route-map <CONDITION></code>	<CONDITION> – имя конфигурируемых правил маршрутизации, задаётся строкой до 31 символа.
4	В контексте настройки address-family заданного BGP соседа указать условие и созданные ранее маршрутные карты.	<code>rtt(config-bgp-neighbor-af) # advertise-map <ADVERTISE> {EXIST-MAP NOT-EXIST-MAP} <CONDITION></code>	<p><ADVERTISE> – имя конфигурируемых правил маршрутизации, задаётся строкой до 31 символа.</p> <p><CONDITION> – имя конфигурируемых правил маршрутизации, задаётся строкой до 31 символа.</p> <p><EXIST-MAP> – условие проверки: если маршруты, описанные в condition-map, присутствуют в BGP RIB, то происходит анонсирование маршрутов, описанных в advertise-map.</p> <p><NOT-EXIST-MAP> – условие проверки: если маршруты, описанные в condition-map, отсутствуют в BGP RIB, то происходит анонсирование маршрутов, описанных в advertise-map.</p>

12.7.4.2. Пример настройки

Задача:

RTT получает маршрут по умолчанию от двух провайдеров – ISP «Zero» и «ISP One». Дополнительно ISP «Zero» анонсирует маршрут 10.1.1.1/32, наличие которого в BGP RIB в дальнейшем и будет отслеживаться.

Необходимо в случае присутствия маршрута 10.1.1.1/32 в BGP RIB анонсировать маршрут 198.51.100.0/24 провайдеру ISP «Zero», если маршрут 10.1.1.1/32 отсутствует в BGP RIB – анонсировать 198.51.100.0/24 провайдеру ISP «One».



Решение:

Сконфигурируем необходимые сетевые интерфейсы на каждом устройстве в сети:

```
ISP-ZERO(config)# interface gigabitethernet 1/0/1
ISP-ZERO(config-if-gi)# ip firewall disable
ISP-ZERO(config-if-gi)# ip address 192.0.2.1/30
ISP-ZERO(config-if-gi)# do commit
ISP-ZERO(config-if-gi)# do confirm
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# description "FROM ISP-ZERO"
RTT(config-if-gi)# ip address 192.0.2.2/30
RTT(config-if-gi)# exit
RTT(config)# interface gigabitethernet 1/0/2
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# ip address 192.0.2.5/30
RTT(config-if-gi)# description "TO ISP-ONE"
RTT(config-if-gi)# exit
RTT(config)# do commit
RTT(config)# do confirm
```

```
ISP-ONE(config)# interface gigabitethernet 1/0/1
ISP-ONE(config-if-gi)# ip firewall disable
ISP-ONE(config-if-gi)# ip address 192.0.2.6/30
ISP-ONE(config-if-gi)# do commit
ISP-ONE(config-if-gi)# do confirm
```

Произведем настройку BGP:

```
ISP-ZERO(config)# ip route 10.1.1.1/32 blackhole
```

```
ISP-ZERO(config)# ip route 0.0.0.0/0 blackhole
ISP-ZERO(config)# route-map OUT
ISP-ZERO(config-route-map)# rule 1
ISP-ZERO(config-route-map-rule)# exit
ISP-ZERO(config-route-map)# exit
ISP-ZERO(config)# router bgp 64515
ISP-ZERO(config-bgp)# neighbor 192.0.2.2
ISP-ZERO(config-bgp-neighbor)# remote-as 64516
ISP-ZERO(config-bgp-neighbor)# enable
ISP-ZERO(config-bgp-neighbor)# address-family ipv4 unicast
ISP-ZERO(config-bgp-neighbor-af)# route-map OUT out
ISP-ZERO(config-bgp-neighbor-af)# enable
ISP-ZERO(config-bgp-neighbor-af)# exit
ISP-ZERO(config-bgp-neighbor)# exit
ISP-ZERO(config-bgp)# enable
ISP-ZERO(config-bgp)# address-family ipv4 unicast
ISP-ZERO(config-bgp-af)# redistribute static
ISP-ZERO(config-bgp-af)# do commit
ISP-ZERO(config-bgp-af)# do confirm
RTT(config)# route-map OUT
RTT(config-route-map)# rule 1
RTT(config-route-map-rule)# match ip address 198.51.100.0/24
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
RTT(config)# router bgp 64516
RTT(config-bgp)# neighbor 192.0.2.1
RTT(config-bgp-neighbor)# remote-as 64515
RTT(config-bgp-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af)# route-map OUT out
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# neighbor 192.0.2.6
RTT(config-bgp-neighbor)# remote-as 64517
RTT(config-bgp-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# route-map OUT out
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# enable
RTT(config-bgp)# do commit
RTT(config-bgp)# do confirm
ISP-ONE(config)# ip route 0.0.0.0/0 blackhole
ISP-ONE(config)# route-map OUT
ISP-ONE(config-route-map)# rule 1
ISP-ONE(config-route-map-rule)# exit
ISP-ONE(config-route-map)# exit
ISP-ONE(config)# router bgp 64517
ISP-ONE(config-bgp)# neighbor 192.0.2.5
ISP-ONE(config-bgp-neighbor)# address-family ipv4 unicast
ISP-ONE(config-bgp-neighbor-af)# route-map OUT out
ISP-ONE(config-bgp-neighbor-af)# enable
ISP-ONE(config-bgp-neighbor-af)# exit
ISP-ONE(config-bgp-neighbor)# remote-as 64516
ISP-ONE(config-bgp-neighbor)# enable
ISP-ONE(config-bgp-neighbor)# exit
ISP-ONE(config-bgp)# enable
```



```
ISP-ONE(config-bgp)# address-family ipv4 unicast
ISP-ONE(config-bgp-af)# redistribute static
ISP-ONE(config-bgp-af)# do commit
ISP-ONE(config-bgp-af)# do confirm
```

Опишем advertise и condition maps на RTT:

```
RTT(config)# ip route 198.51.100.0/24 blackhole
RTT(config)# route-map CONDITION
RTT(config-route-map)# rule 1
RTT(config-route-map-rule)# match ip address 10.1.1.1/32
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
RTT(config)# route-map ADVERTISE
RTT(config-route-map)# rule 1
RTT(config-route-map-rule)# match ip address 198.51.100.0/24
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
RTT(config)# router bgp 64516
RTT(config-bgp)# address-family ipv4 unicast
RTT(config-bgp-af)# network 198.51.100.0/24
RTT(config-bgp-af)# do commit
RTT(config-bgp-af)# do confirm
```

Активируем функцию Conditional advertisement, применив ранее созданные маршрутные карты:

```
RTT(config)# router bgp 64516
RTT(config-bgp)# neighbor 192.0.2.6
RTT(config-bgp-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af)# advertise-map ADVERTISE not exist-map CONDITION
RTT(config-bgp-neighbor-af)# do commit
RTT(config-bgp-neighbor-af)# do confirm
```

Проверим корректность настройки:

//Проверяем наличие маршрута 10.1.1.1/32 в BGP RIB

```
RTT# sh bgp ipv4 unicast 10.1.1.1/32
10.1.1.1/32          via 192.0.2.1 on gil/0/1          [bgp64516 07:07:59]
(64515?)
```

```
Administrative Distance: 170
Type:                    unicast
Origin:                  Incomplete
AS path:                 64515
Next Hop:                192.0.2.1
Output Label:            --
Input Label:             --
Local Preference:       100
MED:                    --
Cluster List:            --
Community:               --
EXT Community:           --
Weight:                  0
Valid, Best
```

// Проверяем статус conditional advertisement и отсутствие анонса 198.51.100.0/24 провайдеру ISP "One"

```
RTT# sh bgp ipv4 unicast neighbor 192.0.2.6 advertise-routes
```

```
RTT# sh bgp neighbors 192.0.2.6
```

```
BGP neighbor is 192.0.2.6
```

```
BGP state:                Established
Type:                     Static neighbor
Neighbor address:         192.0.2.6
Neighbor AS:              64517
Neighbor ID:              192.0.2.6
Neighbor caps:            refresh enhanced-refresh restart-aware
```

```
AS4
```

```
Session:                  external AS4
Source address:           192.0.2.5
Weight:                   0
Hold timer:               99/180
Keepalive timer:         4/60
RR client:                No
```

```
Address family ipv4 unicast:
```

```
Send-label:               No
Default originate:        No
Default information originate: No
Outgoing route-map:       OUT
Advertise-map:            ADVERTISE
Condition-map:            CONDITION
Conditional advertisement status: Withdrawal
Preference:               170
Remove private AS:        No
Next-hop self:            No
Next-hop unchanged:       No
Uptime:                   1300 s
```

Настройка завершена.

12.7.5. Быстрая деактивация пиринговых сессий

В случае, когда возникновения проблем между соседями BGP, по умолчанию BGP ожидает 180 секунд (3 таймера кеераливе) для того, чтобы разорвать соседство и отозвать все маршруты, полученные от неактивного соседа. Для обхода данной проблемы существуют методы, которые помогают быстрее обнаружить проблемы в работе сети и произвести отключение соседа, улучшая время реакции на изменения смежности с соседями BGP. Рассмотрим существующие реализации этих методов.

12.7.5.1. Метод на основе протокола BFD

BFD (Bidirectional Forwarding Detection) — протокол для быстрого обнаружения проблем на канальном уровне. В текущей реализации для его работы необходима настройка с обеих сторон (на каждом BGP-пире).

По умолчанию BFD-сессия устанавливается в следующем режиме:

Протокол	Режим
eBGP	single-hop

Протокол	Режим
eBGP multi-hop	multi-hop
iBGP	multi-hop

Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Настроить протокол BGP (см. раздел Настройка BGP).		
2	Включить поддержку протокола BFD в контексте настройки пира или пир-группы	<code>rtt(config-bgp-neighbor)# fall-over bfd</code>	

Пример настройки

Задача:

Необходимо настроить eBGP между маршрутизаторами R1, R2 и включить протокол BFD.



Решение:

На R1 предварительно необходимо настроить интерфейс Gi1/0/1:

```
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# ip firewall disable
rtt(config-if-gi)# ip address 10.0.0.1/24
```

Следующим шагом на R1 настроим eBGP и включим BFD:

```
rtt(config)# router bgp 100
rtt(config-bgp)# neighbor 10.0.0.2
rtt(config-bgp-neighbor)# remote-as 200
rtt(config-bgp-neighbor)# update-source 10.0.0.1
rtt(config-bgp-neighbor)# fall-over bfd
rtt(config-bgp-neighbor)# enable
rtt(config-bgp-neighbor)# exit
rtt(config-bgp)# enable
rtt(config-bgp)# exit
```

На R2 предварительно необходимо настроить интерфейс Gi1/0/1:

```
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# ip firewall disable
rtt(config-if-gi)# ip address 10.0.0.2/24
```

Далее на R2 настроим eBGP и включим BFD:

```
rtt(config)# router bgp 200
rtt(config-bgp)# neighbor 10.0.0.1
rtt(config-bgp-neighbor)# remote-as 100
rtt(config-bgp-neighbor)# update-source 10.0.0.2
rtt(config-bgp-neighbor)# fall-over bfd
rtt(config-bgp-neighbor)# enable
rtt(config-bgp-neighbor)# exit
rtt(config-bgp)# enable
rtt(config-bgp)# exit
```

Настройка завершена. Для просмотра оперативной информации можно использовать следующие команды:

```
rtt# sh bgp neighbors
BGP neighbor is 10.0.0.2
  BGP state:                               Established    <---- BGP сессия
установлена
  Type:                                     Static neighbor
  Neighbor address:                        10.0.0.2
  Neighbor AS:                             200
  Neighbor ID:                             10.0.0.2
  Neighbor caps:                           refresh enhanced-refresh restart-aware
AS4
  Session:                                 external AS4
  Source address:                          10.0.0.1
  Weight:                                  0
  Hold timer:                              144/180
  Keepalive timer:                         29/60
  Uptime (d,h:m:s):                        00,00:00:20
  BFD address:                             10.0.0.2
  BFD state:                               Up              <---- Статус протокола BFD
  BFD interval:                            0.300 s
  BFD timeout:                             1.500 s

rttv# sh bfd neighbors 10.0.0.2
Neighbor address:                          10.0.0.2
Local address:                             10.0.0.1
Interface:                                gil/0/1
Remote discriminator:                      889907056
Local discriminator:                      924658435
State:                                     Up
Session type:                             Control
Session mode:                             Single-hop
Local diagnostic code:                    No Diagnostic
Remote diagnostic code:                   No Diagnostic
Minimal Tx Interval:                      300 ms
Minimal Rx Interval:                      300 ms
Multiplier:                              5
Actual Tx Interval:                       300 ms
Actual Detection Interval:                 1500 ms
Number of transmitted packets:             257
Number of received packets:               156
Uptime (d,h:m:s):                         00,00:00:38
Client:                                   BGP
Last received packet:
  Desired Min Tx Interval:                 200 ms
```

Required Min Rx Interval: 200 ms
Multiplier: 5

12.7.5.2. Метод на основе Fast Peer Deactivation (Fall-over)

BGP Fast Peer Deactivation – это метод оптимизации конвергенции BGP, при котором соседство BGP разрывается, как только указанный маршрут (или более/менее специфичный) к соседу удаляется из таблицы маршрутизации. Механизм реализован с совместным использованием маршрутных карт (route-map).



Если правило route-map будет пустым, то под правило будет попадать любой доступный маршрут до соседа в таблице маршрутизации.

Функционал поддержан для IPv4 (AFI-1, SAFI-1), IPv6 (AFI-2, SAFI-1) маршрутов. В route-map поддерживаются все значения команды **match**. Команды **action set** игнорируются. Реализована поддержка как для GRT, так и в VRF.

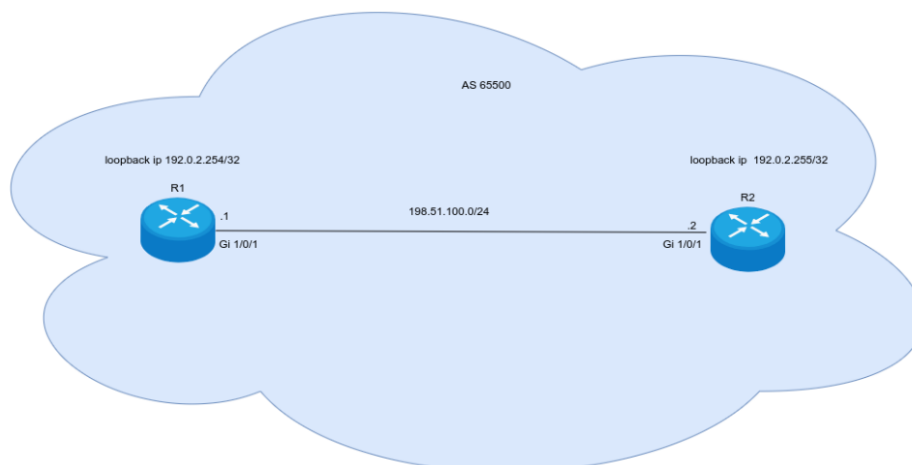
Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Настроить протокол BGP (см. раздел Настройка BGP).		
2	Описать в маршрутной карте подсеть, наличие которой будет отслеживаться в таблице маршрутизации (см. в документе «Справочник команд CLI»).		
3	Активировать функционал, привязав маршрутную карту в соответствующему пиру или пир-группе.	<code>rtt(config-bgp-neighbor)# fall-over route-map <NAME></code>	<NAME> – имя маршрутной карты, задается строкой до 31 символа.

Пример настройки

Задача:

Настроить механизм Fast Peer Deactivation между iBGP пирами R1 и R2.



Решение:

Предварительно настроим связность между маршрутизаторами в схеме:

R1

```
R1(config)# interface gigabitethernet 1/0/1
R1(config-if-gi)# ip firewall disable
R1(config-if-gi)# ip address 198.51.100.1/24
```

R2

```
R2(config)# interface gigabitethernet 1/0/1
R2(config-if-gi)# ip firewall disable
R2(config-if-gi)# ip address 198.51.100.2/24
```

На каждом устройстве настроим протокол OSPF и анонсируем адреса loopback-интерфейсов:

R1

```
R1(config)# router ospf 1
R1(config-ospf)# area 0.0.0.0
R1(config-ospf-area)# enable
R1(config-ospf-area)# exit
R1(config-ospf)# enable
R1(config-ospf)# exit
R1(config)# interface loopback 1
R1(config-if-loopback)# ip ospf instance 1
R1(config-if-loopback)# ip ospf
R1(config-if-loopback)# exit
R1(config)# interface gigabitethernet 1/0/1
R1(config-if-gi)# ip ospf instance 1
R1(config-if-gi)# ip ospf
```

R2

```
R2(config)# router ospf 1
R2(config-ospf)# area 0.0.0.0
R2(config-ospf-area)# enable
R2(config-ospf-area)# exit
R2(config-ospf)# enable
R2(config-ospf)# exit
R2(config)# interface loopback 1
R2(config-if-loopback)# ip address 192.0.2.255/32
R2(config-if-loopback)# ip ospf instance 1
R2(config-if-loopback)# ip ospf
R2(config-if-loopback)# exit
R2(config)# interface gigabitethernet 1/0/1
R2(config-if-gi)# ip ospf instance 1
R2(config-if-gi)# ip ospf
```

Настроим протокол BGP на обоих маршрутизаторах:

R1

```
R1(config)# router bgp 65500
R1(config-bgp)# neighbor 192.0.2.255
R1(config-bgp-neighbor)# remote-as 65500
R1(config-bgp-neighbor)# update-source loopback 1
R1(config-bgp-neighbor)# enable
R1(config-bgp-neighbor)# exit
R1(config-bgp)# enable
R2(config)# router bgp 65500
R2(config-bgp)# neighbor 192.0.2.254
R2(config-bgp-neighbor)# remote-as 65500
R2(config-bgp-neighbor)# update-source loopback 1
R2(config-bgp-neighbor)# enable
R2(config-bgp-neighbor)# exit
R2(config-bgp)# enable
```

Создадим маршрутную карту, в которой опишем адрес BGP-пира для дальнейшего отслеживания с помощью функционала Fast Peer Deactivation:

R1

```
R1(config)# route-map Failover
R1(config-route-map)# rule 1
R1(config-route-map-rule)# match ip address 192.0.2.255/32
R1(config-route-map-rule)# exit
R1(config-route-map)# exit
```

R2

```
R2(config)# route-map Failover
R2(config-route-map)# rule 1
R2(config-route-map-rule)# match ip address 192.0.2.254/32
```

Привяжем созданные маршрутные карты в контексте настройки BGP-пира:

R1

```
R1(config)# router bgp 65500
R1(config-bgp)# neighbor 192.0.2.255
R1(config-bgp-neighbor)# fall-over route-map Failover
```

R2

```
R2(config)# router bgp 65500
R2(config-bgp)# neighbor 192.0.2.254
R2(config-bgp-neighbor)# fall-over route-map Failover
```

Для просмотра оперативного состояния можно воспользоваться следующей командой:

```
R2# sh bgp neighbors
BGP neighbor is 192.0.2.254
  BGP state:                Established
  Type:                     Static neighbor
  Neighbor address:         192.0.2.254
  Neighbor AS:              65500
  Neighbor ID:              192.0.2.254
  Neighbor caps:            refresh enhanced-refresh restart-aware
AS4
  Session:                  internal multihop AS4
  Source address:           192.0.2.255
  Weight:                   0
  Hold timer:               164/180
  Keepalive timer:          23/60
  Uptime:                   437 s
  Fall-over route-map:      Failover      <---- Функционал
активирован

R2# sh bgp neighbors 192.0.2.254
BGP neighbor is 192.0.2.254
  BGP state:                Down
  Type:                     Static neighbor
  Neighbor address:         192.0.2.254
  Neighbor AS:              65500
  Fall-over route-map:      Failover
  Last error:               Error: Fall over route-map  <----
Сессия BGP была разорвана из-за отработавшего механизма Fast Peer Deactivation
```

Настройка завершена.

12.7.6. Настройка политик маршрутизации Route-map

Route-map – это механизм, позволяющий применять условия (условные фильтры) и действия к маршрутам и, соответственно, к трафику. Он используется для фильтрации, изменения и управления атрибутами протокола BGP, обеспечивая расширенные возможности по сравнению с Prefix-list. Подробная логика работы описана в разделе **Политика фильтрации маршрутной информации**.

Функциональные возможности Route-map позволяют работать со следующими атрибутами:

Название	Семейство адресов (AF)	Манипуляции		Поддержка регулярных выражений в классификации
		Тип действия	Поддержка трекинга	
As-path	IPv4, IPv6, VPNv4, L2VPN, Flowspec	Prepend, Replace	Да	Да
Community	Pv4, IPv6, VPNv4, L2VPN, Flowspec	no-advertise, no-export, добавление в атрибута (add), создание списка (set)	Да	Да
Extended community	VPNv4, L2VPN	Добавление в список (add), создание списка (set)	Да	Да
Local preference	IPv4, IPv6, VPNv4, L2VPN, Flowspec	Установка атрибута, увеличение, уменьшение	Да	Да
MED (metric)	IPv4, IPv6, VPNv4, L2VPN, Flowspec	Установка атрибута, увеличение, уменьшение	Да	Да
Next-hop	IPv4, IPv6, VPNv4, L2VPN, Flowspec	Установка атрибута	Нет	Нет
Origin	IPv4, IPv6, VPNv4, L2VPN, Flowspec	Установка атрибута	Да	Нет
Weight	IPv4, IPv6, VPNv4, L2VPN, Flowspec	Установка атрибута, увеличение, уменьшение	Нет	Да

12.7.6.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать маршрутную карту для фильтрации и модификации IP-маршрутов.	<code>rtt(config)# route-map <NAME></code>	<NAME> – имя маршрутной карты, задается строкой до 31 символа.
2	Создать правило маршрутной карты.	<code>rtt(config-route-map)# rule <ORDER></code>	<ORDER> – номер правила, принимает значения [1 .. 10000].
3	Указать действие, которое должно быть применено для маршрутной информации.	<code>rtt(config-route-map-rule)# action <ACT></code>	<ACT> – назначаемое действие: <ul style="list-style-type: none"> • permit – прием или анонсирование маршрутной информации разрешено; • deny – запрещено.

Шаг	Описание	Команда	Ключи
4	Задать значение атрибута BGP AS-Path в маршруте, для которого должно срабатывать правило (необязательно).	<pre>rtt(config-route-map-rule)# match as-path { [begin contain end] <AS-PATH> empty regex <REGEX>}</pre>	<p><AS-PATH> – список номеров автономных систем, задается в виде AS,AS,AS, принимает значения [1..4294967295]. Опциональные параметры:</p> <p><REGEX> – регулярное выражение, задаётся по стандарту POSIX-Extended Regular Expressions.</p> <ul style="list-style-type: none"> • begin – значение атрибута начинается с указанных номеров AS; • contain – значение атрибута содержит указанные номера AS; • empty – значение атрибута пусто; • end – значение атрибута заканчивается указанными номерами AS; • regex – значение атрибута соответствует регулярному выражению.
5	Задать значение атрибута BGP Community, для которого должно срабатывать правило (необязательно).	<pre>rtt(config-route-map-rule)# match community {<COMMUNITY-LIST> regex <REGEX>}</pre>	<p><COMMUNITY-LIST> – список community, задается в виде AS:N,AS:N, принимает значения [1..4294967295]. Можно указать до 64 community.</p> <ul style="list-style-type: none"> • regex – значение атрибута соответствует регулярному выражению. <p><REGEX> – регулярное выражение, задаётся по стандарту POSIX-Extended Regular Expressions.</p>

Шаг	Описание	Команда	Ключи
6	Задать значение атрибута BGP Extended Community, для которого должно срабатывать правило (необязательно).	<pre>rtt(config-route-map-rule)# match extcommunity {<EXTCOMMUNITY-LIST> regex <REGEX>}</pre>	<p><EXTCOMMUNITY-LIST> – список extcommunity, задается в виде KIND:AS:N, KIND:AS:N, где</p> <p>KIND – тип extcommunity:</p> <ul style="list-style-type: none"> • rt (Route Target); • ro (Route Origin). <p>N – номер extcommunity, принимает значения [1..65535].</p> <ul style="list-style-type: none"> • regex – значение атрибута соответствует регулярному выражению. <p><REGEX> – регулярное выражение, задаётся по стандарту POSIX-Extended Regular Expressions.</p>
7	Задать профиль IP-адресов, содержащий значения подсетей назначения в маршруте (необязательно).	<pre>rtt(config-route-map-rule)# match ip address object-group <OBJ-GROUP-NETWORK-NAME></pre> <pre>rtt(config-route-map-rule)# match ipv6 address object-group <OBJ-GROUP-NETWORK-NAME></pre>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, содержащего префиксы подсетей назначения, задаётся строкой до 31 символа.
8	Задать профиль IP-адресов, содержащий значения атрибута BGP Next-Хоп в маршруте для которого должно срабатывать правило (необязательно).	<pre>rtt(config-route-map-rule)# match ip bgp next-hop object-group <OBJ-GROUP-NETWORK-NAME></pre> <pre>rtt(config-route-map-rule)# match ipv6 bgp next-hop object-group <OBJ-GROUP-NETWORK-NAME></pre>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, содержащего префиксы подсетей назначения, задается строкой до 31 символа.
9	Задать профиль, содержащий IP-адреса маршрутизатора, анонсировавшего маршрут, для которого должно срабатывать правило (необязательно).	<pre>rtt(config-route-map-rule)# match ip route-source object-group <OBJ-GROUP-NETWORK-NAME></pre> <pre>rtt(config-route-map-rule)# match ipv6 route-source object-group <OBJ-GROUP-NETWORK-NAME></pre>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, содержащего префиксы подсетей назначения, задается строкой до 31 символа.
10	Задать ACL-группу, для которой должно срабатывать правило (необязательно).	<pre>rtt(config-route-map-rule)# match access-group <NAME></pre>	<NAME> – имя списка контроля доступа, задается строкой до 31 символа.

Шаг	Описание	Команда	Ключи
11	Задать значение атрибута BGP MED в маршруте для которого должно срабатывать правило (необязательно).	<code>rtt(config-route-map-rule)# match metric bgp <METRIC></code>	<METRIC> – значение атрибута BGP MED, принимает значения [0..4294967295].
12	Задать значение атрибута BGP AS-Path, которое будет добавляться в начало списка AS-Path (необязательно).	<code>rtt(config-route-map-rule)# action set as-path prepend <AS-PATH> {track <TRACK-ID>}</code>	<p><AS-PATH> – список номеров автономных систем, который будет добавлен к текущему значению в маршруте. Задаётся в виде AS,AS,AS, принимает значения [1..4294967295].</p> <p><TRACK-ID> – идентификатор vrrp-tracking, при котором будет исполняться указанное действие. Изменяется в диапазоне [1..60].</p>
13	Заменять номер или последовательность номеров AS в атрибуте BGP AS-Path на номер локальной AS (необязательно).	<code>rtt(config-route-map-rule)# action set as-path replace { any <AS-PATH> }</code>	<p><AS-PATH> – список номеров автономных систем, который будет заменён на локальный номер AS. Задаётся в виде AS,AS,AS, принимает значения [1..4294967295].</p> <ul style="list-style-type: none"> • any – заменять любой номер AS.
14	Задать значение атрибута BGP Community, которое будет установлено в маршруте (необязательно).	<code>rtt(config-route-map-rule)# action set community {COMMUNITY-LIST} no-advertise no-export }</code>	<p><COMMUNITY-LIST> – список community, задается в виде AS:N,AS:N, где каждая часть принимает значения [1..65535];</p> <ul style="list-style-type: none"> • no - advertise – маршруты, передаваемые с данным community, не должны анонсироваться другим BGP-соседям; • no - export – маршруты, передаваемые с таким community, не должны анонсироваться eBGP-соседям, но анонсируются внешним соседям в конфедерации.

Шаг	Описание	Команда	Ключи
15	Задать значение атрибута BGP ExtCommunity, которое будет установлено в маршруте (необязательно).	<code>rtt(config-route-map-rule)# action set extcommunity <EXTCOMMUNITY-LIST></code>	<p><EXTCOMMUNITY-LIST> – список extcommunity, задается в виде KIND:AS:N, KIND:AS:N, где</p> <p>KIND – тип extcommunity:</p> <ul style="list-style-type: none"> • rt (Route Target); • ro (Route Origin). <p>N – номер extcommunity, принимает значения [1..65535].</p>
16	Задать атрибут BGP Next-Hop, который будет установлен в маршруте при анонсировании (необязательно).	<code>rtt(config-route-map-rule)# action set ip bgp-next-hop <ADDR></code>	<ADDR> – IP-адрес шлюза, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		<code>rtt(config-route-map-rule)# action set ipv6 bgp-next-hop <IPV6-ADDR></code>	<IPV6-ADDR> – IPv6-адрес шлюза, задается в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].
17	Задать значение атрибута BGP Local Preference, который будет установлен в маршруте (необязательно).	<code>rtt(config-route-map-rule)# action set local-preference <PREFERENCE></code>	<PREFERENCE> – значение атрибута BGP Local Preference, принимает значения [0..255].
18	Задать значение атрибута BGP Origin, которое будет установлено в маршруте (необязательно).	<code>rtt(config-route-map-rule)# action set origin <ORIGIN></code>	<p><ORIGIN> – значение атрибута BGP Origin:</p> <ul style="list-style-type: none"> • egp – маршрут выучен по протоколу EGP; • igp – маршрут получен внутри исходной AS; • incomplete – маршрут выучен другим образом.
19	Задать значение BGP MED, которое будет установлено в маршруте (необязательно).	<code>rtt(config-route-map-rule)# action set metric bgp <METRIC></code>	<METRIC> – значение атрибута BGP MED, принимает значения [0..4294967295].
20	Добавить фильтрацию и модификацию маршрутов во	<code>rtt(config-bgp-neighbor)# route-map <NAME><DIRECTION></code>	

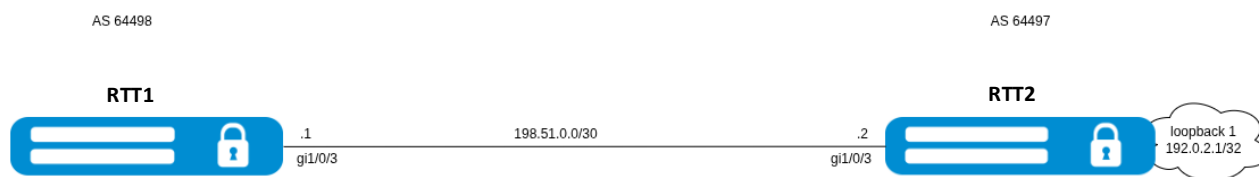
Шаг	Описание	Команда	Ключи
	входящих или исходящих направлениях.	<code>rtt (config-ipv6-bgp-neighbor) # route-map <NAME><DIRECTION></code>	<p><NAME> – имя сконфигурированной маршрутной карты;</p> <p><DIRECTION> – направление:</p> <ul style="list-style-type: none"> in – фильтрация и модификация получаемых маршрутов; out – фильтрация и модификация анонсируемых маршрутов.

12.7.6.2. Пример настройки 1

Задача:

Назначить community для маршрутной информации, приходящей из AS 64498.

Схема:



Базовая конфигурация:

RTT1

```

security zone Untrusted
exit
router bgp 64498
  neighbor 198.51.100.2
    remote-as 64497
  address-family ipv4 unicast
    enable
  exit
  enable
exit
enable
exit

interface gigabitethernet 1/0/3
  security-zone Untrusted
  ip address 198.51.100.1/30
exit
  
```

```
security zone-pair Untrusted self
  rule 1
    action permit
    match protocol tcp
    match destination-port port-range 179
    enable
  exit
  rule 2
    action deny
    enable
  exit
exit
```

RTT2

```
security zone Untrusted
exit
```

```
route-map BGP
  rule 1
    match ip address 192.0.2.1/32
  exit
exit
```

```
router bgp 64497
  neighbor 198.51.100.1
    remote-as 64498
    address-family ipv4 unicast
      route-map BGP out
      enable
    exit
  enable
exit
address-family ipv4 unicast
  network 192.0.2.1/32
exit
enable
exit
```

```
interface gigabitethernet 1/0/3
  security-zone Untrusted
  ip address 198.51.100.2/30
exit
interface loopback 1
  ip address 192.0.2.1/32
exit
```

```
security zone-pair Untrusted self
  rule 1
    action permit
    match protocol tcp
    match destination-port port-range 179
    enable
  exit
  rule 2
    action deny
```

```
enable
exit
exit
```

Решение:

Создаем политику на RTT1:

```
RTT1# config
RTT1(config)# route-map set_community
```

Создаем правило 1:

```
RTT1(config-route-map)# rule 1
```

Если AS PATH содержит AS 64497, то назначаем ему community 64497:100, выходим и применяем конфигурацию:

```
RTT1(config-route-map-rule)# match as-path contain 64497
RTT1(config-route-map-rule)# action set community 64497:100
RTT1(config-route-map-rule)# exit
RTT1(config-route-map)# exit
RTT1(config)# do com
Configuration has been successfully applied and saved to flash. Commit timer
started, changes will be reverted in 600 seconds.
RTT1(config)# do conf
Configuration has been confirmed. Commit timer canceled.
```

Проверяем, что политика была создана:

RTT1

```
RTT1# sh ip route-map set_community
Order:                               1
Description:                         --
Matching pattern:
  Access group                       --
  AS path                            contains 64497
  Community                          --
  Extcommunity                       --
  BGP local-preference:              --
  BGP metric (MED):                  --
  BGP weight:                        --
  Address (object-group):            --
  Next hop (object-group):           --
  Route source (object-group):       --
  RIP metric                         --
  RIP tag                            --
  OSPF metric type                   --
  OSPF metric                        --
  OSPF tag                           --
Actions:
  Decision:                          Permit
  Route next hop:                     --
  Route IPv6 next hop:               --
```



```

IP address: --
IPv6 address: --
AS path (prepend): --
Community: 64497:100
Extcommunity: --
Local preference: --
BGP next hop address: --
BGP IPv6 next hop address: --
BGP metric (MED): --
BGP weight: --
Origin: --
RIP metric --
RIP tag --
OSPF metric type --
OSPF metric --
OSPF tag --

```

В контексте настройки BGP-инстанса заходим в настройки параметров соседа:

```

RTT1(config)# router bgp 64498
RTT1(config-bgp)# neighbor 198.51.100.2
RTT1(config-bgp-neighbor)# address-family ipv4 unicast

```

Привязываем политику к принимаемой маршрутной информации:

```

RTT1(config-bgp-neighbor-af)# route-map set_community in
RTT1(config-bgp-neighbor-af)# do com
RTT1(config-bgp-neighbor-af)# do conf

```

Проверяем, что для полученного префикса установлена необходимая community:

RTT1

```

RTT1# show bgp ipv4 unicast 192.0.2.1/32
192.0.2.1/32      via 198.51.100.2 on gil/0/3      [bgp64498 08:44:32]
(64497i)
  Administrative Distance: 170
  Type:                    unicast
  Origin:                  IGP
  AS path:                 64497
  Next Hop:               198.51.100.2
  Output Label:           --
  Input Label:            --
  Local Preference:       100
  MED:                    --
  Cluster List:           --
  Community:              64497:100
  EXT Community:          --
  Weight:                 0
  Valid, Best

```

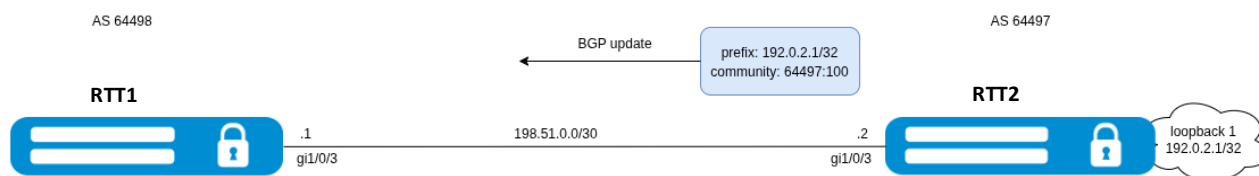
Настройка завершена.

Задача:

Для всей полученной маршрутной информации (с community 64497:100) от RTT2 установить следующие BGP-атрибуты:

- MED – 240;
- Origin – EGP.

Схема:



Базовая конфигурация:

RTT1

```
security zone Untrusted
exit

route-map set_community
  rule 1
    match as-path contain 64497
    action set community 64497:100
  exit
exit

router bgp 64498
  neighbor 198.51.100.2
    remote-as 64497
    address-family ipv4 unicast
      enable
    exit
  enable
exit
enable

interface gigabitethernet 1/0/3
  security-zone Untrusted
  ip address 198.51.100.1/30
exit

security zone-pair Untrusted self
  rule 1
    action permit
    match protocol tcp
    match destination-port port-range 179
```

```
    enable
  exit
  rule 2
    action deny
    enable
  exit
exit
```

RTT2

```
security zone Untrusted
exit

route-map BGP
  rule 1
    match ip address 192.0.2.1/32
    action set community 64497:100
  exit
exit

router bgp 64497
  neighbor 198.51.100.1
    remote-as 64498
    address-family ipv4 unicast
      route-map BGP out
      enable
    exit
  enable
exit
address-family ipv4 unicast
  network 192.0.2.1/32
exit
enable
exit

interface gigabitethernet 1/0/3
  security-zone Untrusted
  ip address 198.51.100.2/30
exit
interface loopback 1
  ip address 192.0.2.1/32
exit

security zone-pair Untrusted self
  rule 1
    action permit
    match protocol tcp
    match destination-port port-range 179
    enable
  exit
  rule 2
    action deny
    enable
  exit
exit
```

Решение:

Для решения задачи настройка будет производиться на RTT1. Первым шагом создаем политику:

```
RTT1(config)# route-map community_in
```

Далее правило:

```
RTT1(config-route-map)# rule 1
```

Если community содержит 64497:100, то назначаем ему MED 240 и Origin EGP:

```
RTT1(config)# route-map community_in
RTT1(config-route-map)# rule 1
RTT1(config-route-map-rule)#
RTT1(config-route-map-rule)# match community 64497:100
RTT1(config-route-map-rule)# action set metric bgp 240
RTT1(config-route-map-rule)# action set origin egp
RTT1(config-route-map-rule)# do com
Configuration has been successfully applied and saved to flash. Commit timer
started, changes will be reverted in 600 seconds.
RTT1(config-route-map-rule)# do conf
```

Проверим, что политика создана корректно:

RTT1

```
RTT1# sh ip route-map community_in
Order:                               1
Description:                         --
Matching pattern:
    Access group                      --
    AS path                          --
    Community                         64497:100
    Extcommunity                     --
    BGP local-preference:             --
    BGP metric (MED):                 --
    BGP weight:                      --
    Address (object-group):           --
    Next hop (object-group):          --
    Route source (object-group):      --
    RIP metric                       --
    RIP tag                          --
    OSPF metric type                  --
    OSPF metric                      --
    OSPF tag                         --
Actions:
    Decision:                        Permit
    Route next hop:                  --
    Route IPv6 next hop:             --
    IP address:                      --
    IPv6 address:                    --
    AS path (prepand):               --
    Community:                       --
    Extcommunity:                   --
```

```

Local preference:                --
BGP next hop address:            --
BGP IPv6 next hop address:       --
BGP metric (MED):                set 240
BGP weight:                      --
Origin:                          EGP
RIP metric                      --
RIP tag                          --
OSPF metric type                 --
OSPF metric                      --
OSPF tag                         --

```

В контексте настройки BGP-инстанса заходим в настройки параметров соседа:

```

RTT1(config)# router bgp 64498
RTT1(config-bgp)# neighbor 198.51.100.2
RTT1(config-bgp-neighbor)# address-family ipv4 unicast

```

Привязываем политику для получаемой маршрутной информации:

```

RTT1(config-bgp-neighbor-af)# route-map community_in in
RTT1(config-bgp-neighbor-af)# do com
RTT1(config-bgp-neighbor-af)# do conf

```

Проверим, что соответствующие атрибуты были изменены:

RTT1

```

RTT1# sh bgp ipv4 unicast 192.0.2.1/32
192.0.2.1/32          via 198.51.100.2 on gil/0/3          [bgp64498 09:19:24]
(64497e)
  Administrative Distance: 170
  Type:                    unicast
  Origin:                  EGP
  AS path:                 64497
  Next Hop:                198.51.100.2
  Output Label:            --
  Input Label:             --
  Local Preference:       100
  MED:                    240
  Cluster List:            --
  Community:               64497:100
  EXT Community:           --
  Weight:                  0
  Valid, Best

```

Настройка завершена.

12.7.6.4. Использование регулярных выражений

Начиная с версии 1.23 доступно использование регулярных выражений в Route-map для контроля распространения маршрутной информации по протоколу BGP. Контроль можно производить

по трём атрибутам BGP: AS-path, community, extcommunity. Синтаксис регулярных выражений соответствует стандарту POSIX ERE. В таблице ниже представлены некоторые примеры регулярных выражений.

Условие совпадения	Регулярное выражение
Маршруты с любым содержимым AS-path	<code>.*</code>
Маршруты с пустым AS-path	<code>^\$</code>
Маршруты с одной любой AS в AS-path	<code>^[0-9]+\$</code>
Маршруты с двумя любыми AS в AS-path	<code>^[0-9]+ [0-9]+\$</code>
Маршруты, зарожённые в AS 15	<code>(^ .*)15\$</code>
Маршруты, полученные из AS 20	<code>^20(.* \$)</code>
Маршруты, проходящие через AS 22	<code>.* 22 .*</code>
Маршруты, проходящие через AS 30, а затем через AS 22	<code>.* 22 30 .*</code>
Маршруты, проходящие через AS 30 или AS 43	<code>.* (30 43) .*</code>
Маршруты, зарожённые в AS 66 и проходящие через AS 60	<code>.* 60 (.*)*66\$</code>
Маршруты, зарожённые в AS 70 или проходящие через неё	<code>.* 70(.* \$)</code>
Маршруты, содержащие приватные AS в AS-path	<code>(^ .*)((6451[2-9]) (645[2-9][0-9]) (64[6-9][0-9]{2}) (65[0-4][0-9]{2}) (655[0-2][0-9]) (6553[0-4]))(.* \$)</code>
Номер AS 100, номер community 200	<code>^100:200\$</code>
Номера AS 112 или 232, любой номер community	<code>^(112 232):[0-9]*\$</code>
Номер AS 277, номер community начинается с 3	<code>^277:3[0-9]*\$</code>
Любой номер AS, номер community в диапазоне 150-1230	<code>^([0-9]*):((1[5-9][0-9]) ([2-9][0-9]{2}) (1[0-2][0-2][0-9]) (1230))\$</code>
Тип route target, IP-адрес 10.10.10.1, номер extcommunity 653 и 654	<code>^rt:10\.10\.10\.1:65[34]\$</code>

Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать маршрутную карту для фильтрации и модификации IP-маршрутов.	<code>rtt(config)# route-map <NAME></code>	<NAME> – имя маршрутной карты, задается строкой до 31 символа.
2	Создать правило маршрутной карты.	<code>rtt(config-route-map)# rule <ORDER></code>	<ORDER> – номер правила, принимает значения [1 .. 10000].

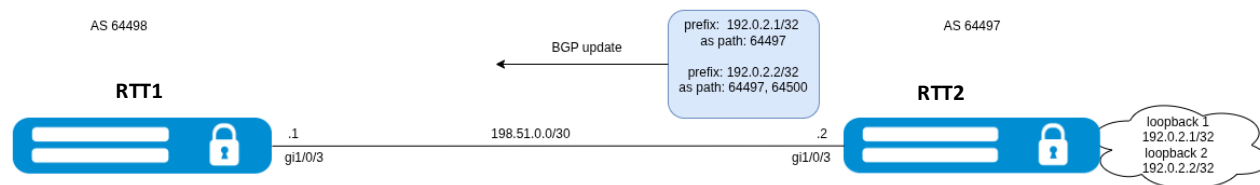
Шаг	Описание	Команда	Ключи
3	Указать действие, которое должно быть применено для маршрутной информации.	<code>rtt(config-route-map-rule) # action <ACT></code>	<p><ACT> – назначаемое действие:</p> <ul style="list-style-type: none"> permit – прием или анонсирование маршрутной информации разрешено; deny – запрещено.
4	Задать значение BGP AS-Path, Community, Extended Community в маршруте, для которого должно срабатывать правило (необязательно).	<code>rtt(config-route-map-rule) # match as-path { [begin contain end] <AS-PATH> empty regex <REGEX> } rtt(config-route-map-rule) # match community { <COMMUNITY-LIST> regex <REGEX> } rtt(config-route-map-rule) # match extcommunity { <EXTCOMMUNITY-LIST> regex <REGEX> }</code>	<p>regex – значение атрибута соответствует регулярному выражению.</p> <p><REGEX> – регулярное выражение, задаётся по стандарту POSIX-Extended Regular Expressions.</p>
5	Описать дополнительные условия для выбора и действие (см. раздел Настройка политик маршрутизации Route-map).		
6	Применить созданный Route-map в контексте настройки BGP peer, peer-group, address-family.	<code>rtt(config-bgp-neighbor) # route-map <NAME><DIRECTION> rtt(config-ipv6-bgp-neighbor) # route-map <NAME><DIRECTION></code>	<p><NAME> – имя сконфигурированной маршрутной карты;</p> <ul style="list-style-type: none"> in – фильтрация и модификация получаемых маршрутов; out – фильтрация и модификация анонсируемых маршрутов.

Пример настройки

Задача:

Запретить прием маршрутной информации по BGP, содержащей в атрибуте AS-path номер AS 64500.

Схема:



Базовая конфигурация:

RTT1

```
security zone Untrusted
exit

interface gigabitethernet 1/0/1
  security-zone Untrusted
  ip address 198.51.100.1/30
exit
interface loopback 1
  ip address 192.0.2.1/32
exit

security zone-pair Untrusted self
  rule 1
    action permit
    match protocol tcp
    match destination-port port-range 179
    enable
  exit
  rule 2
    action deny
    enable
  exit
exit
```

RTT2

```
security zone Untrusted
exit
security zone Trusted
exit

interface gigabitethernet 1/0/1
  security-zone Untrusted
  ip address 198.51.100.2/30
exit
interface gigabitethernet 1/0/2
  security-zone Trusted
  ip address 203.0.113.1/30
exit

security zone-pair Untrusted self
  rule 1
    action permit
    match protocol tcp
    match destination-port port-range 179
```



```
    enable
  exit
  rule 2
    action deny
    enable
  exit
exit
security zone-pair Trusted self
  rule 1
    action permit
    match protocol tcp
    match destination-port port-range 179
    enable
  exit
  rule 2
    action deny
    enable
  exit
exit
```

RTT3

```
security zone Trusted
exit

interface gigabitethernet 1/0/2
  security-zone Trusted
  ip address 203.0.113.2/30
exit
interface loopback 1
  ip address 192.0.2.2/32
exit

security zone-pair Trusted self
  rule 1
    action permit
    match protocol tcp
    match destination-port port-range 179
    enable
  exit
  rule 2
    action deny
    enable
  exit
exit
```

Решение:

Первым шагом необходимо создать Route-map на RTT1, в котором с помощью регулярных выражений опишем интересующий AS-path. В случае совпадения укажем – запретить:

RTT1

```
route-map AS
  rule 1
```

```
match as-path regex '(64500)'  
action deny  
exit  
rule 2  
exit  
exit
```

Проверим корректность ранее созданного Route-map:

RTT1

```
RTT1# sh ip route-map AS  
Order: 1  
Description: --  
Matching pattern:  
  Access group --  
  AS path regex "(64500)"  
  Community --  
  Extcommunity --  
  BGP local-preference: --  
  BGP metric (MED): --  
  BGP weight: --  
  Address (object-group): --  
  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: --  
  Route IPv6 next hop: --  
  IP address: --  
  IPv6 address: --  
  AS path (prepend): --  
  Community: --  
  Extcommunity: --  
  Local preference: --  
  BGP next hop address: --  
  BGP IPv6 next hop address: --  
  BGP metric (MED): --  
  BGP weight: --  
  Origin: --  
  RIP metric --  
  RIP tag --  
  OSPF metric type --  
  OSPF metric --  
  OSPF tag --  
-----  
Order: 2  
Description: --  
Matching pattern:  
  Access group --  
  AS path --  
  Community --  
  Extcommunity --
```

```

BGP local-preference:      --
BGP metric (MED):         --
BGP weight:               --
Address (object-group):   --
Next hop (object-group):  --
Route source (object-group): --
RIP metric                --
RIP tag                   --
OSPF metric type          --
OSPF metric               --
OSPF tag                  --
Actions:
  Decision:               Permit
  Route next hop:         --
  Route IPv6 next hop:    --
  IP address:             --
  IPv6 address:           --
  AS path (prepand):      --
  Community:              --
  Extcommunity:           --
  Local preference:       --
  BGP next hop address:   --
  BGP IPv6 next hop address: --
  BGP metric (MED):       --
  BGP weight:             --
  Origin:                 --
  RIP metric              --
  RIP tag                 --
  OSPF metric type        --
  OSPF metric             --
  OSPF tag                --

```

В контексте настройки пира применим созданный Route-map для фильтрации входящих маршрутов:

RTT1

```

RTT1(config)# router bgp 64498
RTT1(config-bgp)# neighbor 198.51.100.2
RTT1(config-bgp-neighbor)# address-family ipv4 unicast
RTT1(config-bgp-neighbor-af)# route-map AS in
RTT1(config-bgp-neighbor-af)# do com
RTT1(config-bgp-neighbor-af)# do conf

```

Проверим, что BGP RIB не содержит маршрут, в котором AS-path 64500:

RTT1

```

RTT1# sh bgp ipv4 unicast
Status codes: u - unicast, b - broadcast, m - multicast, a - anycast
               * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete

```

Network	Next Hop	Metric	LocPrf	Weight	Path
*> u 192.0.2.1/32	198.51.100.2	--	100	0	64497 i

12.7.7. Конфедерация

Механизм позволяет разделить одну автономную систему на множество под-AS, функционирующих как отдельные административные единицы, но представляющих единую AS для внешних автономных систем. Взаимодействие между под-AS осуществляется посредством межконфедерационных BGP-сессий, использующих расширенную семантику AS_PATH. Для реализации механизма в рамках RFC 5065 введены атрибуты AS_CONFED_SEQUENCE и AS_CONFED_SET, которые применяются исключительно внутри конфедерации и подлежат удалению перед передачей маршрутов за её пределы.

Ограничения:

1. При работе с атрибутом AS-PATH в route-map будет использован AS_SEQUENCE/AS_SET;
2. Для корректной работы с динамическими соседями (listen-range) необходимо, чтобы все AS, описанные в as-range, входили в диапазон confederation peers.

Стандарт и реализация включают в себя следующие изменения в поведении:

- Значение атрибута MED распространяется между eBGP-пирами;
- Значение атрибута next-hop не изменяется при eBGP-пиринге (поведение можно переопределить с помощью команды **next-hop-self**);
- Внутри конфедерации политика анонсирования маршрутной информации аналогична политике анонсирования для iBGP-пиринга;
- Длина списка AS_CONFED_SEQUENCE или AS_CONFED_SET не участвует в политике выбора лучшего маршрута.

12.7.7.1. Алгоритм настройки

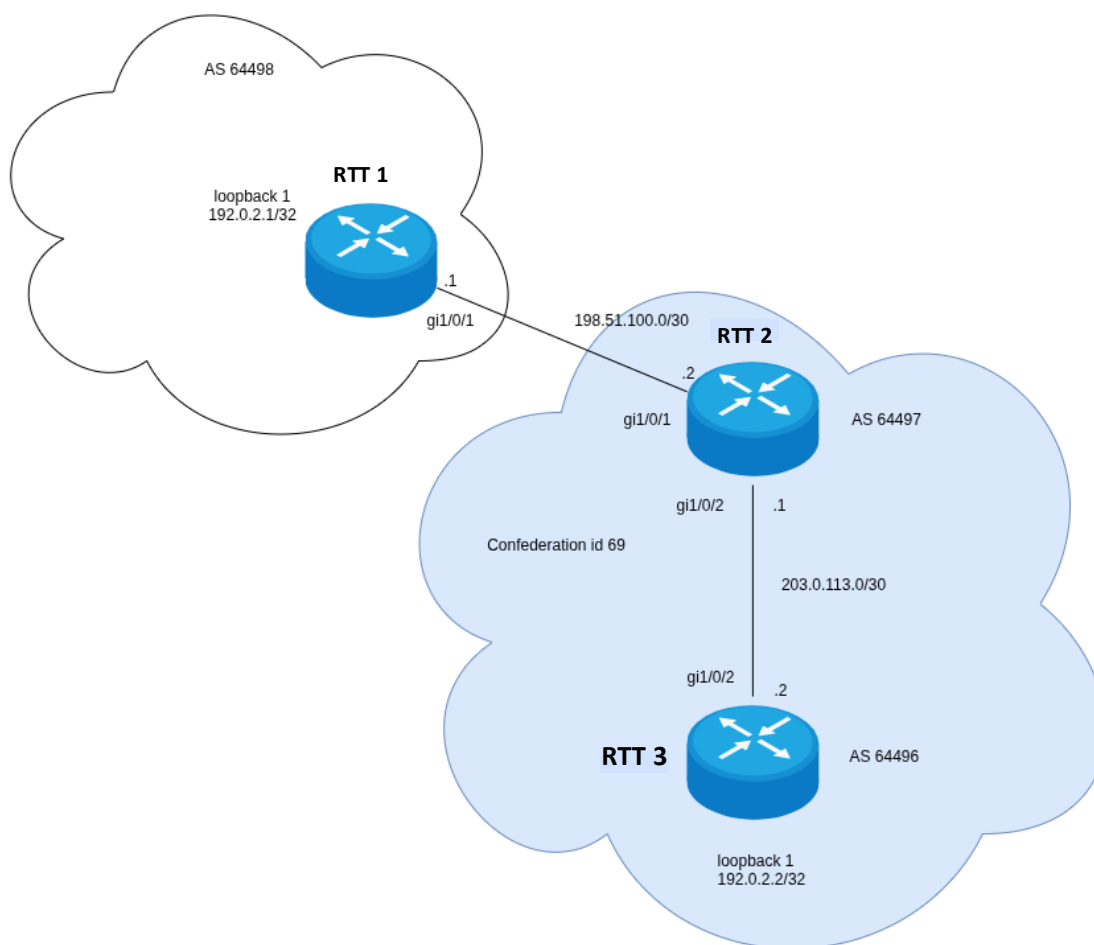
Шаг	Описание	Команда	Ключи
1	В контексте настройки BGP указать идентификатор конфедерации	<code>rtt(config-bgp) # confederation identifier <ID></code>	<ID> – идентификатор конфедерации, принимает значение [1..4294967295].
2	Сконфигурировать члены конфедерации	<code>rtt(config-bgp) # confederation peer <AS></code>	<AS> – список номеров автономных систем, задается в виде AS-AS,AS,AS-AS, принимает значения [1..4294967295].

12.7.7.2. Пример настройки

Задача:

Необходимо настроить конфедерацию между RTT2 и RTT3. На RTT2 настроить eBGP-пиринг с RTT1, проанонсировать подсети в соответствии со схемой.

Схема:



Базовая конфигурация:

RTT1

```
security zone Untrusted
exit

interface gigabitethernet 1/0/1
  security-zone Untrusted
  ip address 198.51.100.1/30
exit
interface loopback 1
  ip address 192.0.2.1/32
exit

security zone-pair Untrusted self
rule 1
  action permit
  match protocol tcp
  match destination-port port-range 179
enable
```

```
exit
rule 2
    action deny
    enable
exit
exit
```

RTT2

```
security zone Untrusted
exit
security zone Trusted
exit

interface gigabitethernet 1/0/1
    security-zone Untrusted
    ip address 198.51.100.2/30
exit
interface gigabitethernet 1/0/2
    security-zone Trusted
    ip address 203.0.113.1/30
exit

security zone-pair Untrusted self
    rule 1
        action permit
        match protocol tcp
        match destination-port port-range 179
        enable
    exit
    rule 2
        action deny
        enable
    exit
exit
security zone-pair Trusted self
    rule 1
        action permit
        match protocol tcp
        match destination-port port-range 179
        enable
    exit
    rule 2
        action deny
        enable
    exit
exit
```

RTT3

```
security zone Trusted
exit

interface gigabitethernet 1/0/2
    security-zone Trusted
    ip address 203.0.113.2/30
exit
```

```
interface loopback 1
  ip address 192.0.2.2/32
exit

security zone-pair Trusted self
  rule 1
    action permit
    match protocol tcp
    match destination-port port-range 179
    enable
  exit
  rule 2
    action deny
    enable
  exit
exit
```

Решение:

Первым шагом настроим BGP внутри конфедерации: установим пиринг, зададим идентификатор, определим члены конфедерации. На RTT3 проанонсируем соответствующий loopback.

RTT2

```
RTT2(config)# router bgp 64497
RTT2(config-bgp)# confederation id 69          <--- Назначение
идентификатора конфедерации
RTT2(config-bgp)# confederation peer 64496      <--- AS с номер 64496
является членом конфедерации
RTT2(config-bgp)# neighbor 203.0.113.2
RTT2(config-bgp-neighbor)# remote-as 64496
RTT2(config-bgp-neighbor)# address-family ipv4 unicast
RTT2(config-bgp-neighbor-af)# enable
RTT2(config-bgp-neighbor-af)# exit
RTT2(config-bgp-neighbor)# enable
RTT2(config-bgp-neighbor)# exit
RTT2(config-bgp)# enable
RTT2(config-bgp)# exit
RTT2(config)#
RTT2(config)# do com
RTT2(config)# do conf
```

RTT3

```
RTT3(config)# router bgp 64496
RTT3(config-bgp)# confederation id 69
RTT3(config-bgp)# confederation peer 64497
RTT3(config-bgp)# neighbor 203.0.113.1
RTT3(config-bgp-neighbor)# remote-as 64497
RTT3(config-bgp-neighbor)# address-family ipv4 unicast
RTT3(config-bgp-neighbor-af)# enable
RTT3(config-bgp-neighbor-af)# exit
RTT3(config-bgp-neighbor)# enable
RTT3(config-bgp-neighbor)# exit
```

```

RTT3(config-bgp)# address-family ipv4 unicast
RTT3(config-bgp-af)# network 192.0.2.2/32
RTT3(config-bgp-af)# exit
RTT3(config-bgp)# enable
RTT3(config-bgp)# exit
RTT3(config)# do com
RTT3(config)# do conf

```

Проверяем, что конфедерация успешно сконфигурирована с обеих сторон:

```

RTT2# sh bgp summary
2025-11-26 06:37:55
  BGP router identifier 198.51.100.2, local AS number 64497, AS confederation identifier 69
<----
  BGP activity 1/0 prefixes
  Neighbor          AS              MsgRcvd    MsgSent    Up/Down      St/PfxRcd
  -----
  203.0.113.2       64496          9          10        00,00:03:07    1

RTT3# sh bgp summary
2025-11-26 06:38:52
  BGP router identifier 192.0.2.2, local AS number 64496, AS confederation identifier 69
<----
  BGP activity 0/1 prefixes
  Neighbor          AS              MsgRcvd    MsgSent    Up/Down      St/PfxRcd
  -----
  203.0.113.1       64497          7          10        00,00:04:03    0

```

BGP-сессия перешла в состояние "Established". Необходимо убедиться, что тип сессии соответствует "confed-external". Для eBGP-сессии автоматически увеличится multi-hop до 255:

```

RTT3# sh bgp neighbors
BGP neighbor is 203.0.113.1
  BGP state:                Established
  Type:                     Static neighbor
  Neighbor address:         203.0.113.1
  Neighbor AS:              64497
  Neighbor ID:              198.51.100.2
  Neighbor caps:            refresh enhanced-refresh restart-aware
AS4
  Session:                  confed-external multihop AS4
<-----
  Source address:           203.0.113.2
  Weight:                   0
  Hold timer:               120/180
  Keepalive timer:          18/60
  EBGP multi-hop:           255  <-----
  RR client:                No
  Address family ipv4 unicast:
    Send-label:             No
    Default originate:      No
    Default information originate: No
    Preference:             170
    Remove private AS:      No
    Next-hop self:          No
    Next-hop unchanged:     No
  Uptime (d,h:m:s):         00,00:02:09

```



```

RTT2# sh bgp neighbors
BGP neighbor is 203.0.113.2
  BGP state:                Established
  Type:                     Static neighbor
  Neighbor address:         203.0.113.2
  Neighbor AS:              64496
  Neighbor ID:              192.0.2.2
  Neighbor caps:            refresh enhanced-refresh restart-aware
AS4
  Session:                  confed-external multihop AS4
<-----
  Source address:           203.0.113.1
  Weight:                   0
  Hold timer:               140/180
  Keepalive timer:         38/60
  EBGP multi-hop:           255    <-----
  RR client:                No
  Address family ipv4 unicast:
    Send-label:             No
    Default originate:      No
    Default information originate: No
    Preference:             170
    Remove private AS:      No
    Next-hop self:          No
    Next-hop unchanged:     No
  Uptime (d,h:m:s):         00,00:08:05

```

Проверим, что RTT2 принимает анонсируемый RTT3 маршрут 192.0.2.2/32:



Для обозначения использования AS_CONFED_SEQUENCE AS Path указывается в круглых скобках.

RTT2

```

RTT2# sh bgp ipv4 unicast 192.0.2.2/32
192.0.2.2/32      via 203.0.113.2 on gil0/2      [bgp64497 06:34:48]
(64496i)
  Administrative Distance: 170
  Type:                   unicast
  Origin:                 IGP
  Aggregator:             --
  AS path:                (64496)    <-----
  Next Hop:               203.0.113.2
  Output Label:           --
  Input Label:            --
  Local Preference:       100
  MED:                   --
  Cluster List:           --
  Community:              --
  EXT Community:          --
  Weight:                 0

```

Следующим шагом настроим взаимодействие конфедерации с внешней автономной системой. При настройке пиринга необходимо помнить, что в качестве внешней AS будет использоваться идентификатор конфедерации.

RTT1

```
RTT1(config)# route-map OUT
RTT1(config-route-map)# rule 1
RTT1(config-route-map-rule)# match ip address 192.0.2.1/32
RTT1(config-route-map-rule)# exit
RTT1(config-route-map)# rule 2
RTT1(config-route-map-rule)# action deny
RTT1(config-route-map-rule)# exit
RTT1(config-route-map)# exit
RTT1(config)# router bgp 64498
RTT1(config-bgp)# neighbor 198.51.100.2
RTT1(config-bgp-neighbor)# remote-as 69
RTT1(config-bgp-neighbor)# address-family ipv4 unicast
RTT1(config-bgp-neighbor-af)# route-map OUT out
RTT1(config-bgp-neighbor-af)# enable
RTT1(config-bgp-neighbor-af)# exit
RTT1(config-bgp-neighbor)# enable
RTT1(config-bgp-neighbor)# exit
RTT1(config-bgp)# address-family ipv4 unicast
RTT1(config-bgp-af)# network 192.0.2.1/32
RTT1(config-bgp-af)# exit
RTT1(config-bgp)# enable
RTT1(config-bgp)# exit
RTT1(config)#
RTT1(config)# do com
RTT1(config)# do conf
```

RTT2

```
RTT2(config)# route-map OUT
RTT2(config-route-map)# rule 1
RTT2(config-route-map-rule)# match ip address 192.0.2.2/32
RTT2(config-route-map-rule)# exit
RTT2(config-route-map)# rule 2
RTT2(config-route-map-rule)# action deny
RTT2(config-route-map-rule)# exit
RTT2(config-route-map)# exit
RTT2(config)#
RTT2(config)# router bgp 64497
RTT2(config-bgp)# neighbor 198.51.100.1
RTT2(config-bgp-neighbor)# remote-as 64498
RTT2(config-bgp-neighbor)# address-family ipv4 unicast
RTT2(config-bgp-neighbor-af)# route-map OUT out
RTT2(config-bgp-neighbor-af)# enable
RTT2(config-bgp-neighbor-af)# exit
RTT2(config-bgp-neighbor)# enable
RTT2(config-bgp-neighbor)# exit
RTT2(config-bgp)# neighbor 203.0.113.2
RTT2(config-bgp-neighbor)# address-family ipv4 unicast
RTT2(config-bgp-neighbor-af)# next-hop-self
```

```
RTT2(config-bgp-neighbor-af)# exit
RTT2(config-bgp-neighbor)# do com
RTT2(config-bgp)# do com
RTT2(config-bgp)# do conf
```

Проверяем, что пиринг поднялся:

```
RTT2# sh bgp neighbors 198.51.100.1
BGP neighbor is 198.51.100.1
  BGP state:                Established
  Type:                     Static neighbor
  Neighbor address:         198.51.100.1
  Neighbor AS:              64498
  Neighbor ID:              192.0.2.1
  Neighbor caps:            refresh enhanced-refresh restart-aware
AS4
  Session:                  external AS4
  Source address:           198.51.100.2
  Weight:                   0
  Hold timer:               136/180
  Keepalive timer:         36/60
  RR client:                No
  Address family ipv4 unicast:
    Send-label:             No
    Default originate:      No
    Default information originate: No
    Outgoing route-map:     OUT
    Preference:             170
    Remove private AS:      No
    Next-hop self:          No
    Next-hop unchanged:     No
  Uptime (d,h:m:s):        00,00:16:04

RTT1# sh bgp neighbors 198.51.100.2
BGP neighbor is 198.51.100.2
  BGP state:                Established
  Type:                     Static neighbor
  Neighbor address:         198.51.100.2
  Neighbor AS:              69
  Neighbor ID:              198.51.100.2
  Neighbor caps:            refresh enhanced-refresh restart-aware
AS4
  Session:                  external AS4
  Source address:           198.51.100.1
  Weight:                   0
  Hold timer:               135/180
  Keepalive timer:         39/60
  RR client:                No
  Address family ipv4 unicast:
    Send-label:             No
    Default originate:      No
    Default information originate: No
    Outgoing route-map:     OUT
    Preference:             170
    Remove private AS:      No
    Next-hop self:          No
    Next-hop unchanged:     No
```

Проверяем корректность анонсируемой информации:

RTT1

```
RTT1# sh bgp ipv4 unicast 192.0.2.2/32
192.0.2.2/32 via 198.51.100.2 on gil/0/1 [bgp64498 07:49:49] (69i)
Administrative Distance: 170
Type: unicast
Origin: IGP
Aggregator: --
AS path: 69
Next Hop: 198.51.100.2
Output Label: --
Input Label: --
Local Preference: 100
MED: --
Cluster List: --
Community: --
EXT Community: --
Weight: 0
Valid, best, external
```

RTT2

```
RTT3# sh bgp ipv4 unicast 192.0.2.1/32
192.0.2.1/32 via 203.0.113.1 on gil/0/2 [bgp64496 07:49:51]
(64498i)
Administrative Distance: 170
Type: unicast
Origin: IGP
Aggregator: --
AS path: (64497) 64498
Next Hop: 203.0.113.1
Output Label: --
Input Label: --
Local Preference: 100
MED: --
Cluster List: --
Community: --
EXT Community: --
Weight: 0
Valid, best, confed-external
```

Настройка завершена.

12.8. Настройка Policy-Based Routing

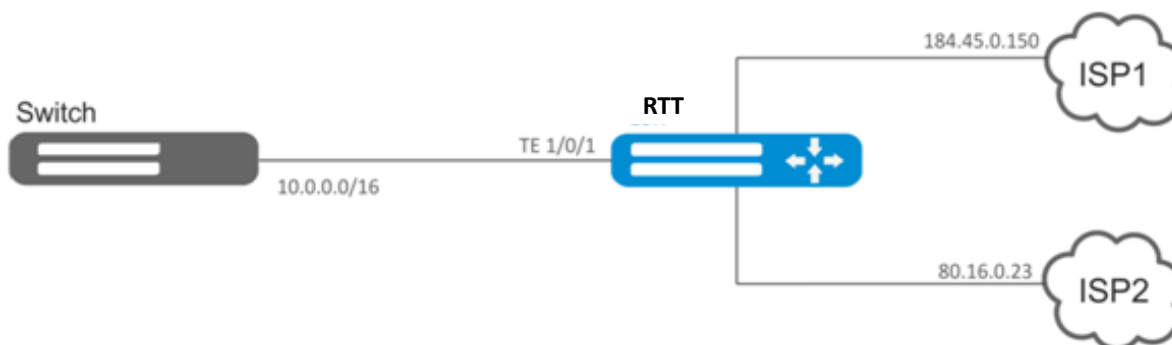
Policy-Based Routing (PBR) — это механизм маршрутизации, который позволяет принимать решения о форвардинге трафика на основе заданных политик, а не основываясь на таблице маршрутизации. В отличие от традиционной маршрутизации, которая опирается исключительно на наилучший путь по метрике (например, кратчайший маршрут), PBR предоставляет администраторам

гибкий инструмент для управления трафиком с учётом дополнительных параметров: источника трафика, типа протокола, VLAN, уровня приоритета и других.

12.8.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать маршрутную карту для фильтрации и модификации IP-маршрутов.	rtt(config)# route-map <NAME>	<NAME> – имя маршрутной карты, задаётся строкой до 31 символа.
2	Создать правило маршрутной карты.	rtt(config-route-map)# rule <ORDER>	<ORDER> – номер правила, принимает значения [1..10000].
3	Указать действие, которое должно быть применено для маршрутной информации.	rtt(config-route-map-rule)# action <ACT>	<ACT> – назначаемое действие: <ul style="list-style-type: none"> • permit – прием или анонсирование маршрутной информации разрешено; • deny – запрещено.
4	Задать ACL, для которого должно срабатывать правило (необязательно).	rtt(config-route-map-rule)# match ip access-group <NAME>	<NAME> – имя списка контроля доступа, задаётся строкой до 31 символа.
5	Задать Next-Hop для пакетов, которые попадают под критерии в указанном списке доступа (ACL) (необязательно).	rtt(config-route-map-rule)# action set ip next-hop verify-availability <NEXTHOP><METRIC>	<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].
6	Назначить политику маршрутизации на основе списков доступа (ACL).	rtt(config-if-gi)# ip policy route-map <NAME>	<NAME> – имя сконфигурированной политики маршрутизации, строка до 31 символа.
7	Разрешить фильтрацию и модификацию локального трафика на основе политики маршрутизации.	rtt(config)# ip local policy [vrf <VRF>] route-map <NAME>	<NAME> – имя сконфигурированной политики маршрутизации, строка до 31 символа.

12.8.2. Пример настройки



Задача:

Распределить трафик между Интернет-провайдерами на основе подсетей пользователей.

Предварительно нужно назначить IP-адреса на интерфейсы.

Требуется направлять трафик с адресов 10.0.20.0/24 через ISP1 (184.45.0.150), а трафик с адресов 10.0.30.0/24 – через ISP2 (80.16.0.23). Требуется контролировать доступность адресов провайдеров (работоспособность подключений к ISP), и при неработоспособности одного из подключений переводить с него на рабочее подключение весь трафик.

Решение:

Создаем ACL:

```

rtt# configure
rtt(config)# ip access-list extended sub20
rtt(config-acl)# rule 1
rtt(config-acl-rule)# match source-address 10.0.20.0 255.255.255.0
rtt(config-acl-rule)# match destination-address any
rtt(config-acl-rule)# match protocol any
rtt(config-acl-rule)# action permit
rtt(config-acl-rule)# enable
rtt(config-acl-rule)# exit
rtt(config-acl)# exit
rtt(config)# ip access-list extended sub30
rtt(config-acl)# rule 1
rtt(config-acl-rule)# match source-address 10.0.30.0 255.255.255.0
rtt(config-acl-rule)# match destination-address any
rtt(config-acl-rule)# match protocol any
rtt(config-acl-rule)# action permit
rtt(config-acl-rule)# enable
rtt(config-acl-rule)# exit
rtt(config-acl)# exit
  
```

Создаем политику:

```

rtt(config)# route-map PBR
  
```

Создаем правило 1:

```
rtt(config-route-map)# rule 1
```

Указываем список доступа (ACL) в качестве фильтра:

```
rtt(config-route-map-rule)# match ip access-group sub20
```

Указываем next-hop для sub20:

```
rtt(config-route-map-rule)# action set ip next-hop verify-availability  
184.45.0.150 10  
rtt(config-route-map-rule)# action set ip next-hop verify-availability  
80.16.0.23 30  
rtt(config-route-map-rule)# exit  
rtt(config-route-map)# exit
```

Правилом 1 будет обеспечена маршрутизация трафика из сети 10.0.20.0/24 на адрес 184.45.0.150, а при его недоступности – на адрес 80.16.0.23. Приоритетность шлюзов задается значениями метрик – 10 и 30.

Создаем правило 2:

```
rtt(config-route-map)# rule 2
```

Указываем список доступа (ACL) в качестве фильтра:

```
rtt(config-route-map-rule)# match ip access-group sub30
```

Указываем nexthop для sub30 и выходим:

```
rtt(config-route-map-rule)# action set ip next-hop verify-availability  
80.16.0.23 10  
rtt(config-route-map-rule)# action set ip next-hop verify-availability  
184.45.0.150 30  
rtt(config-route-map-rule)# exit  
rtt(config-route-map)# exit
```

Правилом 2 будет обеспечена маршрутизация трафика из сети 10.0.30.0/24 на адрес 80.16.0.23, а при его недоступности – на адрес 184.45.0.150. Приоритетность задается значениями метрик.

Заходим на интерфейс TE 1/0/1:

```
rtt(config)# interface tengigabitethernet 1/0/1
```

Привязываем политику на соответствующий интерфейс:

```
rtt(config-if-te)# ip policy route-map PBR
```

12.9. Настройка BFD

BFD (Bidirectional Forwarding Detection) – это протокол, работающий поверх других протоколов, и позволяющий сократить время обнаружения проблемы до 50 мс. BFD является двусторонним протоколом, т. е. требует настройки обоих маршрутизаторов (оба маршрутизатора генерируют BFD-пакеты и отвечают друг другу).

По умолчанию сессия устанавливается в следующем режиме:

Протокол	Режим
iBGP	multi-hop
eBGP	single-hop
eBGP multi-hop	multi-hop
OSPF	single-hop
IS-IS	single-hop
Static route	single-hop
RIP	single-hop

Для изменения поведения (режима) необходимо вручную переопределить параметры сессии, указав необходимый режим. Рассмотрим на примере.

Допустим, мы установили eBGP-соседство и включили для него BFD:

```
RTT# show running-config routing bgp
router bgp 65516
  neighbor 10.100.0.2
    remote-as 65515
    update-source 10.100.0.1
    fall-over bfd
    enable
  exit
enable
exit
```

```
RTT# show bfd neighbors 10.100.0.2
Neighbor address:      10.100.0.2
Local address:         10.100.0.1
Interface:             --
Remote discriminator:  3751534121
Local discriminator:   1670865501
State:                 Up
Session type:          Control
Session mode:          Single-hop
Local diagnostic code:  No Diagnostic
Remote diagnostic code: No Diagnostic
Minimal Tx Interval:   300 ms
Minimal Rx Interval:   300 ms
Multiplier:            5
```



```
Actual Tx Interval:          300 ms
Actual Detection Interval:    1500 ms
Number of transmitted packets: 1149
Number of received packets:  1153
Uptime:                      2m
Client:                      BGP
Last received packet:
  Desired Min Tx Interval:    300 ms
  Required Min Rx Interval:    300 ms
  Multiplier:                 5
```

Как видно, по умолчанию BFD установился в режиме single-hop. Переключим режим в multi-hop:

```
RTT(config)# ip bfd neighbor 10.100.0.2 local-address 10.100.0.1 multihop
RTT(config)# do commit
RTT(config)# do confirm
```

Конфигурацию необходимо производить на обоих устройствах. После переустановки сессии ее режим сменится на multi-hop:

```
rtt# sh bfd neighbors 10.100.0.2
Neighbor address:          10.100.0.2
Local address:             10.100.0.1
Interface:                 --
Remote discriminator:      3751534121
Local discriminator:       1670865501
State:                     Up
Session type:              Control
Session mode:              Multi-hop
Local diagnostic code:     No Diagnostic
Remote diagnostic code:    No Diagnostic
Minimal Tx Interval:       300 ms
Minimal Rx Interval:       300 ms
Multiplier:                5
Actual Tx Interval:        300 ms
Actual Detection Interval:  1500 ms
Number of transmitted packets: 9
Number of received packets: 11
Uptime:                    2m
Client:                    BGP
Last received packet:
  Desired Min Tx Interval:  300 ms
  Required Min Rx Interval:  300 ms
  Multiplier:               5
```

12.9.1. Настройка таймеров



Значение таймеров индивидуально для каждой сети и во многом зависит от ее параметров. В случае частого флапинга BFD рекомендуется увеличить значение таймеров.

Таймеры, вне зависимости от режима работы протокола (single или multi-hop mode), могут быть настроены в контексте глобальной конфигурации или на определенных интерфейсах. Настройка на интерфейсах имеет наибольший приоритет.

```
RTT(config)# ip bfd min-tx-interval 1000
RTT(config)# ip bfd min-rx-interval 1000
RTT(config)# do commit
```

```
RTT# sh ip bfd
Minimum RX interval: 1000 ms
Minimum TX interval: 1000 ms
Idle TX interval:    1000 ms
Multiplier:         5 packets
Passive:             No
```

После того как BFD-сессия установлена, каждая сторона индивидуально вычисляет свои Tx Interval и Detection Interval. Tx Interval выбирается как наибольшее значение из локального Tx Interval и удаленного RX Interval. Detection Interval вычисляется по следующей формуле: $\text{Detection Interval} = \text{remoteMultiplier} * \text{MAX}(\text{RxLocal} || \text{TxRemote})$, где remoteMultiplier – значение Multiplier удаленной стороны, RxLocal – локальный Tx Interval, TxRemote – Tx Interval удаленной стороны.

Локально настроенные таймеры, таймеры удаленной стороны, а также вычисленные таймеры можно посмотреть следующим образом:

```
rtt# sh bfd neighbors 10.100.0.2
Neighbor address:      10.100.0.2
Local address:         10.100.0.1
Interface:             --
Remote discriminator:  3751534121
Local discriminator:   1670865501
State:                 Up
Session type:          Control
Session mode:          Multi-hop
Local diagnostic code:  No Diagnostic
Remote diagnostic code: No Diagnostic
Minimal Tx Interval:   300 ms      <---- Локальный Tx Interval
Minimal Rx Interval:   300 ms      <---- Локальный Rx Interval
Multiplier:           5           <---- Локальный Multiplier
Actual Tx Interval:    300 ms      <---- Вычисленный Tx Interval
Actual Detection Interval: 1500 ms <---- Вычисленный Detection
Interval
Number of transmitted packets: 21781
Number of received packets: 21804
Uptime:                1d21h54m
Client:                 BGP
Last received packet:
  Desired Min Tx Interval: 300 ms      <----
  Required Min Rx Interval: 300 ms      <---- Таймеры удаленной
стороны
  Multiplier:            5           <----
```

12.9.2. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Активировать BFD для протокола OSPF на интерфейсе.	<code>rtt(config-if-gi)# ip ospf bfd-enable</code>	
2	Активировать BFD для neighbor протокола BGP.	<code>rtt(config-bgp-neighbor)# fall-over bfd</code>	
3	Активировать BFD для протокола RIP на интерфейсе.	<code>rtt(config-if-gi)# ip rip bfd-enable</code>	
4	Задать интервал, по истечении которого происходит отправка BFD-сообщения соседу. Глобально (необязательно).	<code>rtt(config)# ip bfd idle-tx-interval <TIMEOUT></code>	<p><TIMEOUT> – интервал, по истечении которого происходит отправка BFD-пакета, принимает значение в миллисекундах в диапазоне:</p> <ul style="list-style-type: none"> • для R800 – [200..65535]; • для R100/200 – [300..65535]. <p>По умолчанию: 1 секунда.</p>
5	Включить логирование изменений состояния BFD-протокола (необязательно).	<code>rtt(config)# ip bfd log-adjacency-changes</code>	
6	Задать минимальный интервал, по истечении которого сосед должен сгенерировать BFD-сообщение. Глобально (необязательно).	<code>rtt(config)# ip bfd min-rx-interval <TIMEOUT></code>	<p><TIMEOUT> – интервал, по истечении которого должна происходить отправка BFD-сообщения соседом, принимает значение в миллисекундах в диапазоне:</p> <ul style="list-style-type: none"> • для R800 – [200..65535]; • для R100/200 – [300..65535]. <p>По умолчанию:</p> <ul style="list-style-type: none"> • на R800: 200 миллисекунд; • на R100/200: 300 миллисекунд.

Шаг	Описание	Команда	Ключи
7	Задать минимальный интервал, по истечении которого происходит отправка BFD-сообщения соседу. Глобально (необязательно).	<code>rtt(config)# ip bfd min-tx-interval <TIMEOUT></code>	<p><TIMEOUT> – интервал, по истечении которого должна происходить отправка BFD-сообщения соседом, принимает значение в миллисекундах в диапазоне:</p> <ul style="list-style-type: none"> • для R800 – [200..65535]; • для R100/200 – [300..65535]. <p>По умолчанию:</p> <ul style="list-style-type: none"> • на R800 – 200 миллисекунд; • на R100/200 – 300 миллисекунд.
8	Задать число пропущенных пакетов, после достижения которого BFD-сосед считается недоступным. Глобально.	<code>rtt(config)# ip bfd multiplier <COUNT></code>	<p><COUNT> – число пропущенных пакетов, после достижения которого сосед считается недоступным, принимает значение в диапазоне [1..100].</p> <p>По умолчанию: 5.</p>
9	Запустить работу механизма BFD с определенным IP-адресом.	<code>rtt(config)# ip bfd neighbor <ADDR> [{ interface <IF> tunnel <TUN> }] [local-address <ADDR> [multihop]] [vrf <VRF>]</code>	<p><ADDR> – IP-адрес шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><IF> – интерфейс или группы интерфейсов;</p> <p><TUN> – тип и номер туннеля;</p> <p><VRF> – имя экземпляра VRF, задается строкой до 31 символа;</p> <p>multihop – ключ для установки TTL=255, для работы механизма BFD через маршрутизируемую сеть.</p>
10	Перевести BFD-сессию в пассивный режим, то есть BFD-сообщения не будут отправляться до тех пор, пока не будут получены сообщения от BFD-соседа. Глобально (необязательно).	<code>rtt(config)# ip bfd passive</code>	

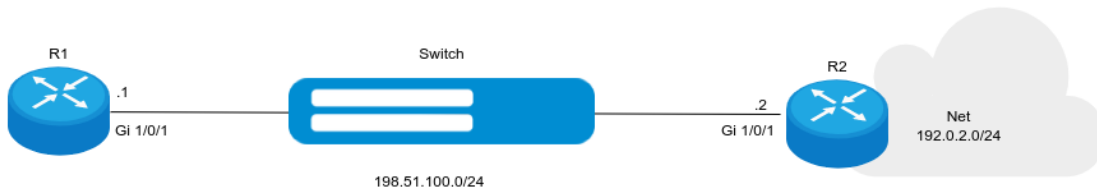
Шаг	Описание	Команда	Ключи
11	Задать интервал, по истечении которого происходит отправка BFD-сообщения соседу. На интерфейсе (необязательно).	<code>rtt(config-if-gi)# ip bfd idle-tx-interval <TIMEOUT></code>	<p><TIMEOUT> – интервал, по истечении которого происходит отправка BFD-пакета, принимает значение в миллисекундах в диапазоне:</p> <ul style="list-style-type: none"> • для R800 – [200..65535]; • для R100/200 – [300..65535]. <p>По умолчанию: 1 секунда.</p>
12	Задать минимальный интервал, по истечении которого сосед должен сгенерировать BFD-сообщение. На интерфейсе (необязательно).	<code>rtt(config-if-gi)# ip bfd min-rx-interval <TIMEOUT></code>	<p><TIMEOUT> – интервал, по истечении которого должна происходить отправка BFD-сообщения соседом, принимает значение в миллисекундах в диапазоне:</p> <ul style="list-style-type: none"> • для R800 – [200..65535]; • для R100/200 – [300..65535]. <p>По умолчанию:</p> <ul style="list-style-type: none"> • на R800 – 200 миллисекунд; • на R100/200 – 300 миллисекунд.
13	Задать минимальный интервал, по истечении которого происходит отправка BFD-сообщения соседу. На интерфейсе (необязательно).	<code>rtt(config-if-gi)# ip bfd min-tx-interval <TIMEOUT></code>	<p><TIMEOUT> – интервал, по истечении которого должна происходить отправка BFD-сообщения соседом, принимает значение в миллисекундах в диапазоне:</p> <ul style="list-style-type: none"> • для R800 – [200..65535]; • для R100/200 – [300..65535]. <p>По умолчанию:</p> <ul style="list-style-type: none"> • на R800 – 200 миллисекунд; • на R100/200 – 300 миллисекунд.
14	Задать число пропущенных пакетов, после достижения которого BFD-сосед считается недоступным. На интерфейсе (необязательно).	<code>rtt(config-if-gi)# ip bfd multiplier <COUNT></code>	<p><COUNT> – число пропущенных пакетов, после достижения которого сосед считается недоступным, принимает значение в диапазоне [1..100].</p> <p>По умолчанию: 5.</p>

Шаг	Описание	Команда	Ключи
15	Перевести BFD-сессию в пассивный режим, то есть BFD-сообщения не будут отправляться до тех пор, пока не будут получены сообщения от BFD-соседа. На интерфейсе (необязательно).	<code>rtt(config-if-gi)# ip bfd passive</code>	
16	При активизации работы протока BFD на интерфейсе с включенным firewall, необходимо разрешить работу протокола UDP порт назначения – 3784 из зоны, сконфигурированной на интерфейсе в зону self. Как создать необходимое правило описано в разделе Конфигурирование Firewall.		

12.9.3. Пример настройки

Задача:

Необходимо настроить протокол BFD для статического маршрута на маршрутизаторе R1.



Решение:

Предварительно необходимо настроить интерфейс Gi1/0/1 на R1 и R2:

R1

```
R1(config)# interface gigabitethernet 1/0/1
R1(config-if-gi)# ip firewall disable
R1(config-if-gi)# ip address 198.51.100.1/24
```

R2

```
R2(config)# interface gigabitethernet 1/0/1
R2(config-if-gi)# ip firewall disable
R2(config-if-gi)# ip address 198.51.100.2/24
```

На R1 настроим статический маршрут и привяжем к нему функционал BFD:

R1

```
R1(config)# ip route 192.0.2.0/24 198.51.100.2 bfd
```

Для установки BFD-сессии на R2 также необходимо настроить соседа:

R2

```
R2(config)# ip bfd neighbor 198.51.100.1
```

Для вывода оперативной информации возможно использование следующих команд:

```
R1# sh bfd neighbors
Neighbor                               Discriminator State      Interface
-----
198.51.100.2                          2907010617    Up        gil/0/1
R1# sh bfd neighbors 198.51.100.2
Neighbor address: 198.51.100.2
Local address:    198.51.100.1
Interface:        gil/0/1
Remote discriminator: 2907010617
Local discriminator: 2856477782
State:            Up
Session type:     Control
Session mode:     Single-hop
Local diagnostic code: No Diagnostic
Remote diagnostic code: No Diagnostic
Minimal Tx Interval: 300 ms
Minimal Rx Interval: 300 ms
Multiplier:       5
Actual Tx Interval: 300 ms
Actual Detection Interval: 1500 ms
Number of transmitted packets: 1444
Number of received packets: 1402
Uptime (d,h:m:s): 00,00:03:39
Client:           STATIC
подписан на отслеживание изменения состояния <---- состояние протокола
<---- сервис, который
```

```
R1# sh ip route 192.0.2.0/24
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, V - VRRP route
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       H - NHRP, * - FIB route

S      * 192.0.2.0/24      [1/0]          via 198.51.100.2 on gil/0/1      [static 16:22:27]
R1# sh bfd neighbors
Neighbor                               Discriminator State      Interface
-----
198.51.100.2                          2907010617    Up        gil/0/1
R1# sh ip route 192.0.2.0/24
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, V - VRRP route
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       H - NHRP, * - FIB route

S      * 192.0.2.0/24      [1/0]          via 198.51.100.2 on gil/0/1      [static 16:22:27]
<---- маршрут присутствует в FIB
```

// После того как BFD-сессия разрушилась, отслеживаемый маршрут удалился из FIB:

```
R1# sh bfd neighbors
Neighbor                               Discriminator State      Interface
-----
198.51.100.2                          2907010617    Down      gi1/0/1
```

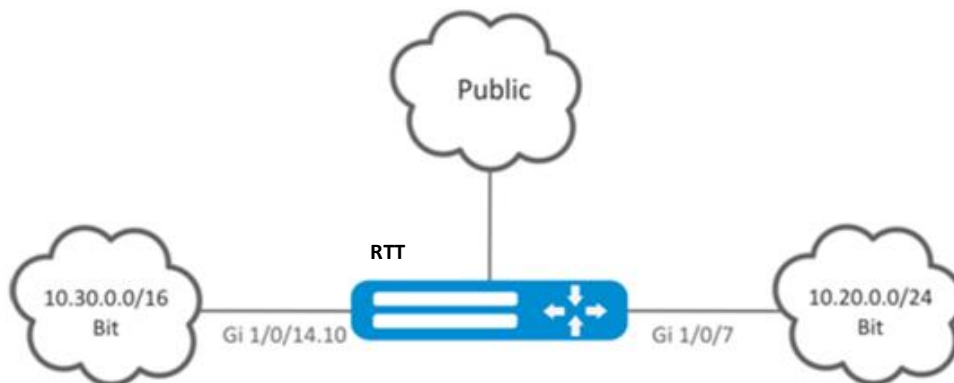
```
R1# sh ip route 192.0.2.0/24
Codes: C - connected, S - static, R - RIP derived,
       O - OSPF derived, IA - OSPF inter area route,
       E1 - OSPF external type 1 route, E2 - OSPF external type 2 route
       B - BGP derived, D - DHCP derived, K - kernel route, V - VRRP route
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       H - NHRP, * - FIB route
```

R1#

Настройка завершена.

12.10. Настройка VRF

VRF (Virtual Routing and Forwarding) – технология, которая позволяет изолировать маршрутную информацию, принадлежащую различным классам (например, маршруты одного клиента).



12.10.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать экземпляр VRF и перейти в режим настройки параметров экземпляра VRF.	rtt(config)# ip vrf <VRF>	<VRF> – имя экземпляра VRF, задается строкой до 31 символа.
2	Назначить описание конфигурируемого экземпляра VRF.	rtt(config-vrf)# description <DESCRIPTION>	<DESCRIPTION> – описание экземпляра VRF, задается строкой до 255 символов.
3	Настроить емкость таблиц маршрутизации в конфигурируемом VRF для	rtt(config-vrf)# ip protocols <PROTOCOL> max-routes <VALUE>	

Шаг	Описание	Команда	Ключи
	IPv4/IPv6 протоколов маршрутизации (необязательно).	rtt(config-vrf)#ipv6 protocols <PROTOCOL> max-routes <VALUE>	<p><PROTOCOL> – вид протокола, принимает значения: ospf, bgp;</p> <p><VALUE> – количество маршрутов в маршрутной таблице, принимает значения в диапазоне:</p> <ul style="list-style-type: none"> • OSPF R800 – [1..500000], R100/200 – [1..300000], • BGP R800 – [1..5000000], R100/200 – [1..2500000]. <p>Значение по умолчанию: 0.</p>
4	Включить и настроить протоколы динамической маршрутизации трафика (Static/OSPF/BGP/IS-IS) в экземпляре VRF (необязательно). См. разделы Конфигурирование статических маршрутов , Настройка OSPF и Настройка BGP .		
5	В режиме конфигурирования физического/логического интерфейса, туннеля, правила DNAT/SNAT, DAS-сервера или SNMPv3 пользователя указать имя экземпляра VRF для которого будет использоваться (при необходимости).	rtt(config-snat-ruleset)# ip vrf forwarding <VRF>	<VRF> – имя экземпляра VRF, задается строкой до 31 символа.
6	Настроить LT-туннель для передачи трафика в глобальный режим или другие VRF (при необходимости).		См. раздел Настройка LT-туннелей .

12.10.2. Пример настройки

Задача:

К маршрутизатору RTT подключены 2 сети, которые необходимо изолировать от остальных сетей.

Решение:

Создадим VRF:

```
rtt(config)# ip vrf bit
rtt(config-vrf)# exit
```

Создадим зону безопасности:

```
rtt(config)# security zone vrf-sec
rtt(config-zone)# ip vrf forwarding bit
rtt(config-zone)# exit
```

Создадим правило для пары зон и разрешим любой TCP/UDP-трафик:

```
rtt(config)# security zone-pair vrf-sec vrf-sec
rtt(config-zone-pair)# rule 1
rtt(config-zone-rule)# match protocol udp
rtt(config-zone-rule)# action permit
rtt(config-zone-rule)# enable
rtt(config-zone-rule)# exit
rtt(config-zone-pair)# rule 2
rtt(config-zone-rule)# match protocol tcp
rtt(config-zone-rule)# action permit
rtt(config-zone-rule)# enable
rtt(config-zone-rule)# exit
```

Создадим привязку интерфейсов, назначим IP-адреса, укажем принадлежность к зоне:

```
rtt(config)# interface gigabitethernet 1/0/7
rtt(config-if-gi)# ip vrf forwarding bit
rtt(config-if-gi)# ip address 10.20.0.1/24
rtt(config-if-gi)# security-zone vrf-sec
rtt(config-if-gi)# exit
rtt(config)# interface gigabitethernet 1/0/14.10
rtt(config-if-sub)# ip vrf forwarding bit
rtt(config-if-sub)# ip address 10.30.0.1/16
rtt(config-if-sub)# security-zone vrf-sec
rtt(config-if-sub)# exit
rtt(config)# exit
```

Информацию об интерфейсах, привязанных к VRF, можно посмотреть командой:

```
rtt# show ip vrf
```

Таблицу маршрутов VRF можно просмотреть с помощью команды:

```
rtt# show ip route vrf bit
```

12.11. Настройка MultiWAN

Технология MultiWAN позволяет организовать отказоустойчивое соединение с резервированием линков от нескольких провайдеров, а также решает проблему балансировки трафика между резервными линками.

12.11.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Сконфигурировать интерфейсы, по которым будет работать MultiWAN: установить IP-адреса и указать security-zone.		
2	Прописать статические маршруты через WAN (если необходимо).	rtt(config)# ip route <SUBNET> wan load- balance rule <ID> [<METRIC>]	<ID> – идентификатор создаваемого правила из п.2. [METRIC] – метрика маршрута, принимает значения [0..255].
3	Создать правило WAN и перейти в режим настройки параметров правила.	rtt(config)# wan load- balance rule <ID>	<ID> – идентификатор создаваемого правила, принимает значения [1..50].
4	Задать интерфейсы или туннели, которые являются шлюзами в маршруте, создаваемом службой MultiWAN.	rtt(config-wan-rule)# outbound { interface <IF> tunnel <TUN> } [WEIGHT]	<IF> – имя интерфейса; <TUN> – имя туннеля; [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 со значением по умолчанию. В режиме резервирования активным будет маршрут с наибольшим весом. Значение по умолчанию 1.
5	Описать правила (необязательно).	rtt(config-wan-rule)# description <DESCRIPTION>	<DESCRIPTION> – описание правила wan, задаётся строкой до 255 символов.
6	Данной командой осуществляется переключение из режима балансировки в режим резервирования (если необходимо).	rtt(config-wan-rule)# failover	
7	Данной командой включается отправка ответных пакетов сессии через тот же интерфейс, через который получены входящие пакеты сессии (если необходимо).	rtt(config-wan- rule)# stickiness	
8	Включить wan-правило.	rtt(config-wan-rule)# enable	

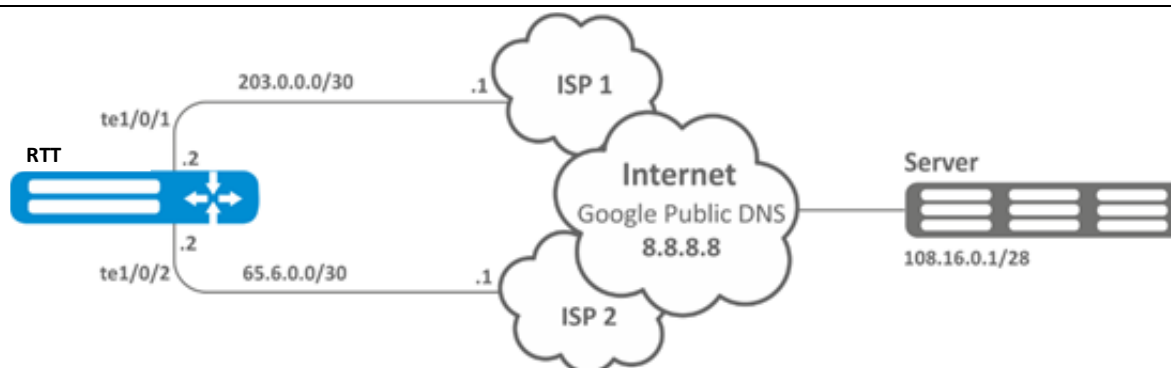
Шаг	Описание	Команда	Ключи
9	Создать список IP-адресов для проверки целостности соединения и осуществить переход в режим настройки параметров списка.	rtt(config)# wan load-balance target-list <NAME>	<NAME> – название списка, задается строкой до 31 символа.
10	Задать цель проверки и перейти в режим настройки параметров цели.	rtt(config-target-list)# target <ID>	<ID> – идентификатор цели, задается в пределах [1..50]. Если при удалении используется значение параметра «all», то будут удалены все цели для конфигурируемого списка целей.
11	Описать target (необязательно).	rtt(config-wan-target)# description <DESCRIPTION>	<DESCRIPTION> – описание target, задается строкой до 255 символов.
12	Указать время ожидания ответа на запрос по протоколу ICMP (необязательно).	rtt(config-wan-target)# resp-time <TIME>	<TIME> – время ожидания, определяется в секундах [1..30].
13	Указать IP-адрес проверки.	rtt(config-wan-target)# ip address <ADDR>	<ADDR> – IP-адрес назначения, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		rtt(config-wan-target)# ipv6 address <IPV6-ADDR>	<IPV6-ADDR> – IPv6-адрес назначения, задается в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].
14	Включить проверку цели.	rtt(config-wan-target)# enable	
Команды для пунктов 14–17 необходимо применить на интерфейсах/туннелях в MultiWAN.			
15	Включить WAN-режим на интерфейсе для IPv4/IPv6-стека.	rtt(config-if-gi)# wan load-balance enable	
		rtt(config-if-gi)# ipv6 wan load-balance enable	
16	Задать количество неудачных попыток проверки соединения, после которых, при отсутствии ответа от встречной стороны, соединение будет считаться неактивным (необязательно).	rtt(config-if-gi)# wan load-balance failure-count <VALUE>	<VALUE> – количество попыток, определяется в диапазоне [1..10].
		rtt(config-if-gi)# ipv6 wan load-balance failure-count <VALUE>	Значение по умолчанию: 1.
17	Задать количество успешных попыток проверки соединения, после которых, в случае успеха, соединение считается вновь активным (необязательно).	rtt(config-if-gi)# wan load-balance success-count <VALUE>	<VALUE> – количество попыток, определяется в диапазоне [1..10].
		rtt(config-if-gi)# ipv6 wan load-balance success-count <VALUE>	Значение по умолчанию: 1.

Шаг	Описание	Команда	Ключи
18	Задать IP-адрес соседа, который будет указан в качестве одного из шлюзов в статическом маршруте, создаваемом службой MultiWAN.	rtt(config-if-gi)# wan load-balance nexthop { <IP> dhcp enable tunnel enable }	<p><IP> – IP-адрес назначения (шлюз), задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p> <p>dhcp enable – если на интерфейсе IP-адрес получен через DHCP-клиента, используется шлюз с DHCP-сервера.</p> <p>tunnel enable – использовать в качестве nexthop – p-t-p адрес назначения. Применимо для подключаемых интерфейсов, работающих через ppp.</p>
		rtt(config-if-gi)# ipv6 wan load-balance nexthop { <IPV6> }	<p><IPV6> – IPv6-адрес назначения (шлюз), задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].</p>
19	Данной командой будут проверяться IP-адреса из списка проверки целостности. В случае недоступности всех (по умолчанию)/хотя бы одной (с использованием ключа check-all) из проверяемых узлов, шлюз будет считаться недоступным.	rtt(config-if-gi)# wan load-balance target-list { check-all <NAME> }	<p><NAME> – проверку производить на основании конкретного target листа (заданного в п.7).</p>
		rtt(config-if-gi)# ipv6 wan load-balance target-list { check-all <NAME> }	<p>check-all – проверку производить на основании всех target листа.</p>
20	Прописать статические маршруты через WAN.	rtt(config)# ip route <SUBNET> wan load-balance rule <ID> [<METRIC>]	<p><ID> – идентификатор создаваемого правила из п.2.</p>
		rtt(config)# ipv6 route <SUBNET> wan load-balance rule <ID> [<METRIC>]	<p>[METRIC] – метрика маршрута, принимает значения [0..255].</p>

12.11.2. Пример настройки

Задача:

Настроить маршрут к серверу (108.16.0.1/28) с возможностью балансировки нагрузки.



Решение:

Предварительно нужно выполнить следующие действия:

- настроить зоны для интерфейсов te1/0/1 и te1/0/2;
- указать IP-адреса для интерфейсов te1/0/1 и te1/0/2.

Основной этап конфигурирования:

Настроим маршрутизацию:

```
rtt(config)# ip route 108.16.0.0/28 wan load-balance rule 1
```

Создадим правило WAN:

```
rtt(config)# wan load-balance rule 1
```

Укажем участвующие интерфейсы:

```
rtt(config-wan-rule)# outbound interface tengigabitethernet 1/0/2
rtt(config-wan-rule)# outbound interface tengigabitethernet 1/0/1
```

Включим созданное правило балансировки и выйдем из режима конфигурирования правила:

```
rtt(config-wan-rule)# enable
rtt(config-wan-rule)# exit
```

Создадим список для проверки целостности соединения:

```
rtt(config)# wan load-balance target-list google
```

Создадим цель проверки целостности:

```
rtt(config-target-list)# target 1
```

Зададим адрес для проверки, включим проверку указанного адреса и выйдем:

```
rtt(config-wan-target)# ip address 8.8.8.8
rtt(config-wan-target)# enable
```

```
rtt(config-wan-target)# exit
```

Настроим интерфейсы. В режиме конфигурирования интерфейса te1/0/1 указываем nexthop:

```
rtt(config)# interface tengigabitethernet 1/0/1
rtt(config-if)# wan load-balance nexthop 203.0.0.1
```

В режиме конфигурирования интерфейса te1/0/1 указываем список целей для проверки соединения:

```
rtt(config-if)# wan load-balance target-list google
```

В режиме конфигурирования интерфейса te1/0/1 включаем WAN-режим и выходим:

```
rtt(config-if)# wan load-balance enable
rtt(config-if)# exit
```

В режиме конфигурирования интерфейса te1/0/2 указываем nexthop:

```
rtt(config)# interface tengigabitethernet 1/0/2
rtt(config-if)# wan load-balance nexthop 65.6.0.1
```

В режиме конфигурирования интерфейса te1/0/2 указываем список целей для проверки соединения:

```
rtt(config-if)# wan load-balance target-list google
```

В режиме конфигурирования интерфейса te1/0/2 включаем WAN-режим и выходим:

```
rtt(config-if)# wan load-balance enable
rtt(config-if)# exit
```

Для переключения в режим резервирования настроим следующее:

Заходим в режим настройки правила WAN:

```
rtt(config)# wan load-balance rule 1
```

Функция MultiWAN также может работать в режиме резервирования, в котором трафик будет направляться в активный интерфейс с наибольшим весом. Включить данный режим можно следующей командой:

```
rtt(config-wan-rule)# failover
```

12.12. Настройка IS-IS

IS-IS — протокол динамической маршрутизации, стандартизированный ISO, основанный на состояниях линков (link-state). Он обеспечивает быструю сходимость и отличную масштабируемость, экономно использует пропускную способность сетей, использует Алгоритм Дейкстры для просчёта

наилучших маршрутов. Отличительной особенностью протокола IS-IS является работа поверх канального уровня модели OSI, поэтому он не привязан к конкретному протоколу сетевого уровня.

12.12.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать IS-IS процесс и перейти в режим настройки параметров этого процесса.	rtt(config)# router isis <ID> [vrf <VRF>]	<ID> – номер процесса, принимает значения [1..65535]; <VRF> – имя экземпляра VRF, задается строкой до 31 символа.
2	Установить NET-адрес.	rtt(config-isis)# net {<NET>}	<NET> – NET-адрес, формат: ff[.ffff.ffff.ffff.ffff.ffff].ffff.ffff.ffff.00.
3	Включить IS-IS процесс.	rtt(config-isis)# enable	
4	Установить алгоритм аутентификации для L2-уровня (необязательно).	rtt(config-isis)# authentication domain algorithm <ALGORITHM>	<ALGORITHM> – алгоритм аутентификации: <ul style="list-style-type: none"> • cleartext – пароль, передается открытым текстом; • md5 – пароль, хешируется по алгоритму md5.
5	Установить пароль аутентификации для L2-уровня (необязательно).	rtt(config-isis)# authentication domain key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }	<CLEAR-TEXT> – пароль, задается строкой 8 символов; <ENCRYPTED-TEXT> – зашифрованный пароль размером 8 байт (16 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).
6	Установить список ключей для аутентификации (необязательно).	rtt(config-isis)# authentication domain key chain <KEYCHAIN>	<KEYCHAIN> – идентификатор списка ключей, задается строкой до 16 символов.
7	Выбрать алгоритм аутентификации для L1-уровня (необязательно).	rtt(config-isis)# authentication area algorithm <ALGORITHM>	<ALGORITHM> – алгоритм аутентификации: <ul style="list-style-type: none"> • cleartext – пароль, передается открытым текстом; • md5 – пароль, хешируется по алгоритму md5.
8	Установить пароль аутентификации для L1-уровня (необязательно).	rtt(config-isis)# authentication area key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }	<CLEAR-TEXT> – пароль, задается строкой 8 символов; <ENCRYPTED-TEXT> – зашифрованный пароль размером 8 байт (16 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

Шаг	Описание	Команда	Ключи
9	Установить список ключей для аутентификации (необязательно).	rtt(config-isis)# authentication area key chain <KEYCHAIN>	<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.
10	Включить передачу имени маршрутизатора в LSP (необязательно).	rtt(config-isis)# hostname dynamic	
11	Установить уровень работы IS-IS процесса (необязательно).	rtt(config-isis)# is-type {<LEVEL>}	<p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-1-2 – работа производится и на 1, и на 2 уровне; level-2 – работа производится только на 2 уровне.
12	Установить тип метрики, который будет использоваться в работе IS-IS процесса (необязательно).	rtt(config-isis)# metric-style { narrow wide transition } [<LEVEL>]	<p>narrow – принимает и генерирует TLV (о достижимости сетей) старого типа;</p> <p>wide – принимает и генерирует TLV (о достижимости сетей) нового типа;</p> <p>transition – принимает и генерирует TLV (о достижимости сетей) нового и старого типа;</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
13	Установить приоритетность маршрутов для данного IS-IS процесса (необязательно).	rtt(config-isis)# preference {<VALUE>}	<VALUE> – принимает значения [1..255].
14	Включить работу IS-IS с IPv4 и/или IPv6 адресами (необязательно).	rtt(config-isis)# address-family { ipv4 ipv6 }	<p>ipv4 – семейство адресов IPv4;</p> <p>ipv6 – семейство адресов IPv6.</p>

Шаг	Описание	Команда	Ключи
15	Установить интервал обновления собственных LSP (необязательно).	rtt(config-isis)# lsp-refresh-interval { min max } <TIME> [<LEVEL>]	<p>min – минимальный интервал обновления/генерации;</p> <p>max – максимальный интервал обновления/генерации;</p> <p><TIME> – время в секундах, принимает значения [1..65535];</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
16	Установить время жизни собственных LSP (необязательно).	rtt(config-isis)# max-lsp-lifetime <TIME> [<LEVEL>]	<p><TIME> – время в секундах, принимает значения [1..65535];</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
17	Установить таймаут перед следующим расчётом SPF (необязательно).	rtt(config-isis)# spf-timeout <TIME> [<LEVEL>]	<p><TIME> – время в миллисекундах, принимает значения [1..10000];</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
18	Включить анонсирование маршрутов, полученных	rtt(config-isis)# redistribute bgp <AS> [route-map <NAME>] [is-type <LEVEL>]	

Шаг	Описание	Команда	Ключи
	альтернативным способом (необязательно).	rtt(config-isis)# redistribute ipv6 bgp <AS> [route-map <NAME>] [is-type <LEVEL>]	<p><AS> – номер автономной системы, может принимать значения [1..4294967295];</p> <p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых маршрутов, задаётся строкой до 31 символа;</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
		rtt(config-isis)# redistribute ospf <ID> <ROUTE-TYPE> [route-map <NAME>] [is-type <LEVEL>]	<p><ID> – номер процесса, может принимать значение [1..65535];</p> <p><ROUTE-TYPE> – тип маршрута:</p> <ul style="list-style-type: none"> 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;
		rtt(config-isis)# redistribute ipv6 ospf <ID> <ROUTE-TYPE> [route-map <NAME>] [is-type <LEVEL>]	<p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых OSPF-маршрутов, задаётся строкой до 31 символа;</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.

Шаг	Описание	Команда	Ключи
		rtt(config-isis)# redistribute isis <ID> <ROUTE-TYPE> [route- map <NAME>] [is-type <LEVEL>]	<p><ID> – номер процесса, может принимать значение [1..65535];</p> <p><ROUTE-TYPE> – тип маршрута:</p> <ul style="list-style-type: none"> level-1 – анонсирование маршрутов 1 уровня; level-2 – анонсирование маршрутов 1 уровня; inter-area – анонсирование маршрутов IS-IS-процесса между зонами; <p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых IS-IS-маршрутов, задаётся строкой до 31 символа;</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
		rtt(config-isis)# redistribute rip [route-map <NAME>] [is- type <LEVEL>]	<p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых RIP-маршрутов, задаётся строкой до 31 символа;</p>
		rtt(config-isis)# redistribute ipv6 rip [route-map <NAME>] [is-type <LEVEL>]	<p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
		rtt(config-isis)# redistribute static [route-map <NAME>] [is-type <LEVEL>]	<p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых статических маршрутов, задаётся строкой до 31 символа;</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.

Шаг	Описание	Команда	Ключи
		rtt(config-isis)# redistribute connected [route-map <NAME>] [is-type <LEVEL>]	<p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых подключённых маршрутов, задаётся строкой до 31 символа;</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
19	Добавить фильтрацию подсетей во входящих или исходящих обновлениях (необязательно).	rtt(config-isis)# prefix-list { ipv6 <LIST_NAME> <LIST_NAME> } {in out}	<p><LIST-NAME> – имя сконфигурированного списка подсетей, задаётся строкой до 31 символа.</p> <p>in – фильтрация входящих маршрутов;</p> <p>out – фильтрация анонсируемых маршрутов.</p>
20	Добавить фильтрацию подсетей во входящих или исходящих обновлениях (необязательно).	rtt(config-isis)# route-map <NAME> {in out}	<p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых маршрутов, задаётся строкой до 31 символа.</p>
21	Установить принадлежность интерфейса к определённому IS-IS процессу.	rtt(config-if-gi)# isis instance <ID>	<p><ID> – номер процесса, принимает значения [1..65535].</p>
22	Включить работу протокола IS-IS на интерфейсе.	rtt(config-if-gi)# isis enable	
24	Включить использование TLV#8 в hello-пакетах (необязательно).	rtt(config-if-gi)# isis hello-padding	
25	Установить приоритет при выборе DIS (необязательно).	rtt(config-if-gi)# isis priority <VALUE> [<LEVEL>]	<p><VALUE> – число, принимающее значения [0..127];</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.

Шаг	Описание	Команда	Ключи
26	Установить значение метрики для интерфейса (необязательно).	rtt(config-if-gi)# isis metric <VALUE> [<LEVEL>]	<p><VALUE> – число, принимающее значения [1..16777215];</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
27	Установить на каком уровне маршрутизации будет работать текущий процесс IS-IS на конкретном интерфейсе (необязательно).	rtt(config-if-gi)# isis circuit-type {<LEVEL>}	<p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-1-2 – работа производится и на 1, и на 2 уровне; level-2 – работа производится только на 2 уровне.
28	Установить интервал отправки hello-пакетов (необязательно).	rtt(config-if-gi)# isis hello-interval <TIME> [<LEVEL>]	<p><TIME> – время в секундах, принимает значения [1..65535];</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
29	Установить множитель для вычисления и отправки Hold Time (необязательно).	rtt(config-if-gi)# isis hello-multiplier <VALUE> [<LEVEL>]	<p><VALUE> – число, принимающее значения [3..1000];</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
30	Перевести интерфейс в режим работы point-to-point протокола IS-IS (необязательно).	rtt(config-if-gi)# isis network point-to-point	

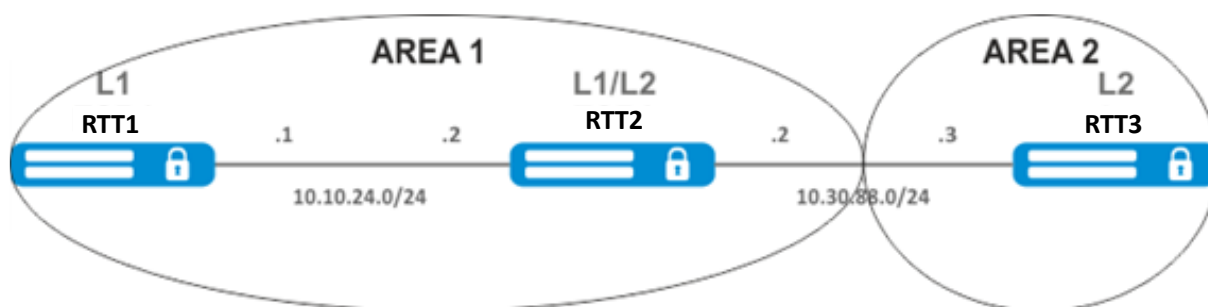
Шаг	Описание	Команда	Ключи
31	Установить интервал генерации и отправки CSNP (необязательно).	rtt(config-if-gi)# isis csnp-interval <TIME> [<LEVEL>]	<p><TIME> – время в секундах, принимает значения [1..65535];</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
32	Установить интервал генерации и отправки PSNP (необязательно).	rtt(config-if-gi)# isis psnp-interval <TIME> [<LEVEL>]	<p><TIME> – время в секундах, принимает значения [1..65535];</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
33	Установить интервал между передачами LSP в Broadcast-сети (необязательно).	rtt(config-if-gi)# isis lsp-interval <TIME> [<LEVEL>]	<p><TIME> – время в миллисекундах, принимает значения [1-10000];</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
34	Установить интервал повторного распространения LSP в PtP-сети (необязательно).	rtt(config-if-gi)# isis lsp-retransmit-interval <TIME> [<LEVEL>]	<p><TIME> – время в секундах, принимает значения [1..65535];</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.

Шаг	Описание	Команда	Ключи
35	Установить алгоритм аутентификации для hello-пакетов (необязательно).	rtt(config-if-gi)# isis authentication algorithm <ALGORITHM> [<LEVEL>]	<p><ALGORITHM> – алгоритм аутентификации:</p> <ul style="list-style-type: none"> cleartext – пароль, передается открытым текстом; md5 – пароль, хешируется по алгоритму md5; <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
36	Установить пароль для аутентификации hello-пакетов (необязательно).	rtt(config-if-gi)# isis authentication key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> } [<LEVEL>]	<p><CLEAR-TEXT> – пароль, задаётся строкой 8 символов;</p> <p><ENCRYPTED-TEXT> – зашифрованный пароль размером 8 байт (16 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...);</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
37	Установить список ключей для аутентификации hello-пакетов (необязательно).	rtt(config-if-gi)# isis authentication key chain <KEYCHAIN> [<LEVEL>]	<p><KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов;</p> <p><LEVEL> – уровень работы протокола IS-IS:</p> <ul style="list-style-type: none"> level-1 – работа производится только на 1 уровне; level-2 – работа производится только на 2 уровне.
38	Включить протокол BFD для протокола IS-IS (необязательно).	rtt(config-if-gi)# isis bfd-enable rtt(config-if-gi)# isis ipv6-bfd-enable	

12.12.2. Пример настройки

Задача:

Настроить протокол IS-IS на маршрутизаторах для обмена маршрутной информацией с соседями. Маршрутизатор RTT1 будет L1-only, RTT2 – L1/L2, RTT3 – L2-only, который также будет находиться в другой area.



Решение:

Предварительно нужно настроить IP-адреса на интерфейсах согласно схеме, приведенной на рисунке выше.

Перейдём к настройке маршрутизатора RTT1. Создадим IS-IS процесс с идентификатором 1 и перейдём в режим конфигурирования протокола:

```
RTT1(config)# router isis 1
```

Зададим номер зоны, в которой будет работать маршрутизатор и его системный идентификатор:

```
RTT1(config-isis)# net 49.0001.1111.1111.1111.00
```

Настроим работу маршрутизатора только на первом уровне протокола IS-IS:

```
RTT1(config-isis)# is-type level-1
```

Зададим работу маршрутизатора с узкой метрикой на первом уровне:

```
RTT1(config-isis)# metric-style narrow level-1
```

Включим работу процесса IS-IS на маршрутизаторе:

```
RTT1(config-isis)# enable
```

Перейдём к конфигурированию интерфейсов. Нужно задать номер процесса IS-IS, который будет работать на интерфейсе и включить работу самого протокола на нём:

```
RTT1(config-if-gi)# isis instance 1
RTT1(config-if-gi)# isis enable
```

Перейдём к настройке маршрутизатора RTT2:

```
RTT2(config)# router isis 2
```

Зададим номер зоны такой же, как на RTT1, а также уникальный системный идентификатор:

```
RTT2(config-isis)# net 49.0001.2222.2222.2222.00
```

Зададим работу маршрутизатора с узкой метрикой на первом уровне и с широкой метрикой на втором и включим работу данного процесса IS-IS:

```
RTT2(config-isis)# metric-style narrow level-1
RTT2(config-isis)# metric-style wide level-2
RTT2(config-isis)# enable
```

Настроим работу интерфейсов на маршрутизаторе. На обоих интерфейсах настройка будет одинаковая:

```
RTT2(config-if-gi)# isis instance 2
RTT2(config-if-gi)# isis enable
```

Перейдём к настройке маршрутизатора RTT3:

```
RTT3(config)# router isis 3
RTT3(config-isis)# net 49.0002.3333.3333.3333.00
RTT3(config-isis)# is-type level-2
RTT3(config-isis)# metric-style wide level-2
RTT3(config-isis)# enable
RTT3(config-if-gi)# isis instance 3
RTT3(config-if-gi)# isis enable
```

Установление соседства можно посмотреть командой **show isis neighbors**. Выполним её на RTT2:

```
RTT2# show isis neighbors
IS-IS 2
IS-IS Level 1 Neighbors
System ID          Hostname      Interface    State    Holdtime  SNPA
1111.1111.1111     RTT1          gi1/0/2      Up        25         a8f9.4baa.1d42
IS-IS Level 2 Neighbors
System ID          Hostname      Interface    State    Holdtime  SNPA
3333.3333.3333     RTT3          gi1/0/1      Up         8         a8f9.4bab.813a
```

13. УПРАВЛЕНИЕ ТЕХНОЛОГИЕЙ MPLS

13.1. Настройка протокола LDP

LDP — протокол распределения меток. Для нахождения соседей используется рассылка hello-сообщений на мультикастный адрес 224.0.0.2. При обмене hello-сообщениями маршрутизаторы узнают транспортные адреса друг друга. Маршрутизатор с большим адресом инициализирует TCP-сессию. После проверки параметров LDP-сессия считается установленной.

В маршрутизаторах RTT поддерживаются следующие режимы работы LDP:

- Режим обмена информации о метках — Downstream Unsolicited;
- Механизм контроля за распространением меток — Independent Label Distribution Control;
- Режим сохранения меток — Liberal Label Retention.



На интерфейсах, где включены протокол LDP и MPLS-коммутация, firewall должен быть отключен.



В текущей реализации протокол LDP работает только с IPv4-адресами.

13.1.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	В контексте настройки параметров MPLS указать интерфейсы, участвующие в процессе MPLS-коммутации.	<pre>rtt(config-mpls)# forwarding interface { <IF> <TUN> }</pre>	<p><IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;</p> <p><TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.</p>

Шаг	Описание	Команда	Ключи
2	Задать router-id для LDP (необязательно, если указан transport-address).	<code>rtt(config-ldp)# router-id { <ID> <IF> <TUN> }</code>	<p><ID> – идентификатор маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p> <p><IF> – интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.</p> <p><TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.</p>
3	В контексте настройки address family ipv4 указать transport-address (необязательно, если указан router-id).	<code>rtt(config-ldp-af- ipv4)# transport- address <ADDR></code>	<ADDR> – задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
4	В контексте настройки address family ipv4 указать интерфейсы для включения на них процесса LDP.	<code>rtt(config-ldp-af- ipv4)# interface { <IF> <TUN> }</code>	<p><IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;</p> <p><TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.</p>
5	Включить процесс LDP.	<code>rtt(config-ldp)# enable</code>	
6	Включить функционал explicit-null (необязательно).	<code>rtt(config-ldp)# egress-label-type explicit-null</code>	
7	В режиме конфигурирования соседа LDP задать пароль командой password (необязательно).	<code>rtt(config-ldp-neig)# password {<TEXT> ENCRYPTED-TEXT}</code>	<p><CLEAR-TEXT> – пароль, задаётся строкой длиной [8..16] символов;</p> <p><ENCRYPTED-TEXT> – зашифрованный пароль размером [8..16] байт ([16..32] символа) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).</p>
<p>В рамках настройки протокола LDP также доступен следующий функционал:</p> <ul style="list-style-type: none"> • Настройка фильтрации LDP-меток (см. Настройка фильтрации LDP-меток); • Настройка параметров LDP-сессии (см. Конфигурирование параметров сессии в протоколе LDP); • Настройка параметров tLDP-сессии (см. Конфигурирование параметров сессии в протоколе targeted-LDP). 			



Если изменить значение router-id, то новое значение будет применено только после рестарта данного протокола. Для рестарта mppls ldp используется команда clear mppls ldp.

13.1.2. Пример настройки

Задача:

Настроить взаимодействие по протоколу LDP между пирами.



Решение:

Предварительная конфигурация маршрутизаторов:

На интерфейсы должны быть назначены IP-адреса, отключен межсетевой экран и настроен один из протоколов внутренней маршрутизации.

Предварительная конфигурация RTT:

```
hostname RTT
router ospf 1
  area 0.0.0.0
  enable
exit
interface gigabitethernet 1/0/1
  ip firewall disable
  ip address 10.10.10.1/30
  ip ospf instance 1
  ip ospf
exit

interface loopback 1
  ip address 1.1.1.1/32
  ip ospf instance 1
  ip ospf
exit
```

Предварительная конфигурация RTT1:

```
hostname RTT1
```

```
router ospf 1
  area 0.0.0.0
  enable
exit

interface gigabitethernet 1/0/1
  ip firewall disable
  ip address 10.10.10.2/30
  ip ospf instance 1
  ip ospf
exit

interface loopback 1
  ip address 4.4.4.4/32
  ip ospf instance 1
  ip ospf
exit
```

Настройка на RTT:

RTT

```
RTT# config
RTT(config)# mpls
RTT(config-mpls)# forwarding interface gigabitethernet 1/0/1
RTT(config-mpls)# ldp
RTT(config-ldp)# router-id 1.1.1.1
RTT(config-ldp)# enable
RTT(config-ldp)# address-family ipv4
RTT(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
RTT(config-ldp-af-ipv4-if)# end
RTT#
```

Настройка на RTT1:

RTT1

```
RTT1# configure
RTT1(config)# mpls
RTT1(config-mpls)# forwarding interface gigabitethernet 1/0/1
RTT1(config-mpls)# ldp
RTT1(config-ldp)# router-id 4.4.4.4
RTT1(config-ldp)# enable
RTT1(config-ldp)# address-family ipv4
RTT1(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
RTT1(config-ldp-af-ipv4-if)# end
RTT1#
```

Проверка:

На одном из пиров ввести следующие команды:

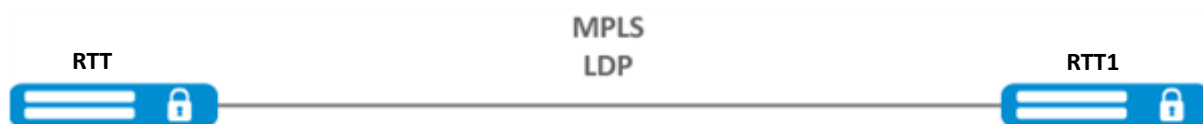
```
RTT# show mpls ldp discovery detailed
Local LDP ID: 1.1.1.1
Discovery sources:
  Interfaces:
    gigabitethernet 1/0/1:
      Hello interval: 5 seconds
      Transport IP address: 1.1.1.1
      LDP ID: 4.4.4.4
      Source IP address: 10.10.10.2
      Transport IP address: 4.4.4.4
      Hold time: 15 seconds
      Proposed hold time: 90/15 (local/peer) seconds
```

Вывод покажет параметры соседнего пира, полученные из мультикастовых hello-сообщений.

Сессия LDP должна находиться в статусе «Operational».

```
RTT1# show mpls ldp neighbor
Peer LDP ID: 4.4.4.4; Local LDP ID 1.1.1.1
State: Operational
TCP connection: 4.4.4.4:40245 - 1.1.1.1:646
Messages sent/received: 10/11
Uptime: 00:00:58
LDP discovery sources:
  gigabitethernet 1/0/1
```

13.2. Конфигурирование параметров сессии в протоколе LDP



По умолчанию в рассылаемых hello-сообщениях установлены следующие значения:

Параметр	LDP
Hello interval	5 секунд
Hold timer	15 секунд
Keepalive holdtime	180 секунд

Hold timer является согласуемым параметром – выбирается наименьший. В данном примере показано, что на RTT после согласования Hold timer равен 10 секундам.

```
RTT# sh mpls ldp discovery detailed
Local LDP ID: 4.4.4.4
Discovery sources:
  Interfaces:
    gigabitethernet 1/0/4:
```

```
Hello interval: 5 seconds
Transport IP address: 4.4.4.4
LDP ID: 1.1.1.1
  Source IP address: 10.10.10.1
  Transport IP address: 1.1.1.1
    Hold time: 10 seconds
    Proposed hold time: 15/10 (local/peer)
```

seconds

Если после согласования Hello interval стал больше, чем Hold timer, то Hello interval будет равным Hold timer/3.

На маршрутизаторах RTT реализована возможность гибкой настройки параметров Hello holdtime, Hello interval и Keepalive holdtime. Рассмотрим пример настройки Hello holdtime для LDP-сессии:

```
RTT# show run mpls
mpls
  ldp
    router-id 4.4.4.4
    discovery hello holdtime 40
    address-family ipv4
      interface gigabitethernet 1/0/4
        discovery hello holdtime 60
    exit
  exit
enable
exit
```

Если параметры Hello Holdtime и Hello Interval не указаны, то используются значения по умолчанию. Если параметры указаны, то приоритет значений для address-family будет выше, чем для значений, сконфигурированных глобально.

```
RTT# show mpls ldp discovery detailed
Local LDP ID: 4.4.4.4
Discovery sources:
  Interfaces:
    gigabitethernet 1/0/4:
      Hello interval: 5 seconds
      Transport IP address: 4.4.4.4
      LDP ID: 1.1.1.1
      Source IP address: 10.10.10.1
      Transport IP address: 1.1.1.1
      Hold time: 15 seconds
      Proposed hold time: 60 /15 (local/peer)
```

seconds

Параметры, сконфигурированные в address-family, могут быть настроены на каждый отдельный интерфейс, участвующий в процессе LDP.

```
RTT# show running-config mpls
mpls
  ldp
    router-id 4.4.4.4
    discovery hello holdtime 50
    discovery hello interval 10
```



```

address-family ipv4
    interface gigabitethernet 1/0/1
        discovery hello holdtime 60
        discovery hello interval 20
    exit
    interface gigabitethernet 1/0/4
        discovery hello holdtime 30
        discovery hello interval 10
    exit
exit
enable
exit

```

Для TCP-сессии Keepalive holdtime является также согласуемым параметром по аналогии с Hold timer. Keepalive interval рассчитывается автоматически и равен Keepalive holdtime/3. Keepalive holdtime можно задать как глобально, так и для каждого соседа. Таймер, заданный для определенного соседа, является более приоритетным.

```

RTT# show running-config mpls
mpls
 ldp
  router-id 4.4.4.4
    keepalive 30 // установлен в глобальной конфигурации LDP
  neighbor 1.1.1.1
    keepalive 55// установлен в соседа с адресом 1.1.1.1
  exit
exit
RTT# sh mpls ldp neighbor 1.1.1.1
Peer LDP ID: 1.1.1.1; Local LDP ID 4.4.4.4
State: Operational
TCP connection: 1.1.1.1:646 - 4.4.4.4:56668
Messages sent/received: 401/401
Uptime: 02:00:24
Peer holdtime: 55
Keepalive interval: 18
LDP discovery sources:

```

13.2.1. Алгоритм настройки параметров Hello holdtime и Hello interval в глобальной конфигурации LDP

Шаг	Описание	Команда	Ключи
1	Настроить протокол LDP (см. раздел Настройка протокола LDP).		
2	В режиме конфигурации протокола LDP задать Hello holdtime.	<code>rtt(config-ldp)# discovery hello holdtime <TIME></code>	<TIME> – время в секундах в интервале [3..65535]. Значение по умолчанию: 15.
3	В режиме конфигурации протокола LDP задать Hello interval.	<code>rtt(config-ldp)# discovery hello interval <TIME></code>	<TIME> – время в секундах в интервале [3..65535]. Значение по умолчанию: 5.

13.2.2. Алгоритм настройки параметров Hello holdtime и Hello interval для address

family

Шаг	Описание	Команда	Ключи
1	Настроить протокол LDP (см. раздел Настройка протокола LDP).		
2	В режиме конфигурации address family протокола LDP установить Hello holdtime на нужном интерфейсе.	<code>rtt(config-ldp-af-ipv4-if) # discovery hello holdtime <TIME></code>	<TIME> – время в секундах в интервале [3..65535]. Значение по умолчанию: 15.
3	В режиме конфигурации address family протокола LDP установить Hello interval на нужном интерфейсе.	<code>rtt(config-ldp-af-ipv4-if) # discovery hello interval <TIME></code>	<TIME> – время в секундах в интервале [3..65535]. Значение по умолчанию: 5.

13.2.3. Алгоритм настройки параметра Keepalive holdtime в глобальной конфигурации LDP

Шаг	Описание	Команда	Ключи
1	Настроить протокол LDP (см. раздел Настройка протокола LDP).		
2	В режиме конфигурации LDP задать параметр Keepalive.	<code>rtt(config-ldp) # keepalive <TIME></code>	<TIME> – время в секундах в интервале [3..65535]. Значение по умолчанию: 180.

13.2.4. Алгоритм настройки параметра Keepalive holdtime для определенного соседа

Шаг	Описание	Команда	Ключи
1	Настроить протокол LDP (см. раздел Настройка протокола LDP).		
2	В режиме конфигурации соседа задать параметр Keepalive holdtime.	<code>rtt(config-ldp-neig) # keepalive <TIME></code>	<TIME> – время в секундах в интервале [3..65535]. Значение по умолчанию: 180.

13.2.5. Пример настройки

Задача:

Переопределить параметры Hello holdtime (40 секунд) и Hello interval (10 секунд) для всего процесса LDP. Для соседа с адресом 1.1.1.1 установить Keepalive holdtime равным 150 секунд.

Решение:

RTT

```
RTT(config)# mpls
RTT(config-mpls)# ldp
RTT(config-ldp)# discovery hello holdtime 40
RTT(config-ldp)# discovery hello interval 10
RTT(config-ldp)# neighbor 1.1.1.1
RTT(config-ldp-neig)# keepalive 150
```

Проверка:

Для просмотра hello-параметров:

RTT

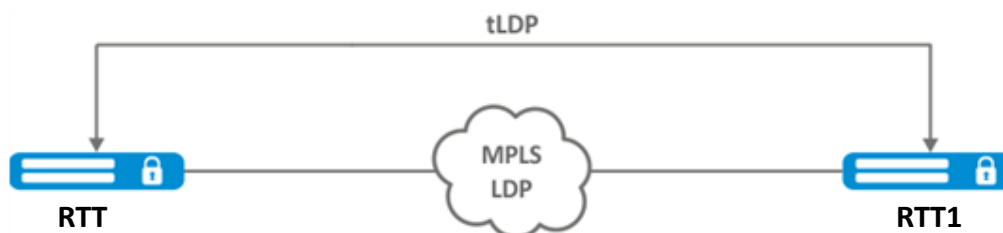
```
RTT# sh mpls ldp discovery detailed
Local LDP ID: 4.4.4.4
Discovery sources:
  Interfaces:
    gigabitethernet 1/0/4:
      Hello interval:      10 seconds
      Transport IP address: 4.4.4.4
      LDP ID:              1.1.1.1
      Source IP address:   10.10.10.1
      Transport IP address: 1.1.1.1
      Hold time:           15 seconds
      Proposed hold time:  40/15 (local/peer) seconds
```

Для просмотра параметров установленной TCP-сессии:

RTT

```
RTT# sh mpls ldp neighbor 1.1.1.1
Peer LDP ID: 1.1.1.1; Local LDP ID 4.4.4.4
State: Operational
TCP connection: 1.1.1.1:646 - 4.4.4.4:45414
Messages sent/received: 15/15
Uptime: 00:06:31
Peer holdtime: 150
Keepalive interval: 50
LDP discovery sources:
```

13.3. Конфигурирование параметров сессии в протоколе targeted-LDP



По умолчанию для targeted LDP-сессии установлены следующие значения:

Параметр	targeted-LDP
Hello interval	5 секунд
Hold timer	45 секунд
Keepalive holdtime	180 секунд

Hold timer является согласуемым параметром – выбирается наименьший. В данном примере показано, что RTT после согласования установил 30 секунд:

```
RTT1# sh mpls ldp discovery detailed
```

```
...
Targeted hellos:
1.1.1.1 -> 4.4.4.4:
Hello interval: 2 seconds
Transport IP address: 1.1.1.1
LDP ID: 4.4.4.4
Source IP address: 4.4.4.4
Transport IP address: 4.4.4.4
Hold time: 30 seconds
Proposed hold time: 30/45 (local/peer) seconds
```

Если после согласования Hello interval стал больше, чем Hold timer, то Hello interval будет равным Hold timer/3.

На маршрутизаторах RTT реализована возможность гибкой настройки параметров Hello holdtime, Hello interval и Keepalive holdtime: параметры можно задать как для всего процесса LDP, так и на соответствующего соседа.

Пример вывода для процесса LDP:

```
RTT# sh running-config mpls
mpls
 ldp
  router-id 1.1.1.1
  keepalive 160
  discovery targeted-hello holdtime 30
  discovery targeted-hello interval 10
exit
exit
```

Пример вывода для targeted-LDP-сессии для определенного соседа:

```
RTT# sh running-config mpls
mpls
 ldp
  router-id 1.1.1.1
  neighbor 4.4.4.4
    keepalive 160
    targeted
    discovery targeted-hello holdtime 30
```

```

        discovery targeted-hello interval 45
    exit
exit
exit

```

Если параметры установлены и для процесса LDP, и на определенного соседа, приоритетом будут считаться настройки, установленные для соседа.

```

RTT# sh running-config mpls
mpls
    ldp
        router-id 1.1.1.1
        keepalive 160
        discovery hello holdtime 90
        discovery targeted-hello interval 30
        neighbor 4.4.4.4
            keepalive 140
            targeted
            discovery targeted-hello holdtime 45
            discovery targeted-hello interval 15
        exit
    exit
exit
RTT# show mpls ldp discovery detailed
...
Targeted hellos:
1.1.1.1 -> 4.4.4.4:
Hello interval: 15 seconds
Transport IP address: 1.1.1.1
LDP ID: 4.4.4.4
Source IP address: 4.4.4.4
Transport IP address: 4.4.4.4
Hold time: 45 seconds
Proposed hold time: 45/45 (local/peer) seconds

```

```

RTT# show mpls ldp neighbor 4.4.4.4
Peer LDP ID: 4.4.4.4; Local LDP ID 1.1.1.1
State: Operational
TCP connection: 4.4.4.4:51861 - 1.1.1.1:646
Messages sent/received: 10/10
Uptime: 00:00:09
Peer holdtime: 140
Keepalive interval: 46
LDP discovery sources:
    1.1.1.1 -> 4.4.4.4:

```

13.3.1. Алгоритм настройки параметров Hello holdtime, Hello interval и Keepalive holdtime для процесса LDP

Шаг	Описание	Команда	Ключи
1	Настроить протокол LDP (см. раздел Настройка протокола LDP).		

Шаг	Описание	Команда	Ключи
2	В режиме конфигурации протокола LDP задать Hello holdtime.	<code>rtt(config-ldp) # discovery targeted-hello holdtime <TIME></code>	<TIME> – время в секундах в интервале [3..65535]. Значение по умолчанию: 45.
3	В режиме конфигурации протокола LDP задать Hello interval.	<code>rtt(config-ldp) # discovery targeted- hello interval <TIME></code>	<TIME> – время в секундах в интервале [1..65535]. Значение по умолчанию: 5.
4	В режиме конфигурации протокола LDP задать Keepalive holdtime.	<code>rtt(config-ldp) # keepalive <TIME></code>	<TIME> – время в секундах в интервале [3..65535]. Значение по умолчанию: 180.

13.3.2. Алгоритм настройки параметров Hello holdtime, Hello interval и Keepalive holdtime для определенного соседа

Шаг	Описание	Команда	Ключи
1	Настроить протокол LDP (см. раздел Настройка протокола LDP).		
2	В режиме конфигурации LDP-соседа задать Hello holdtime.	<code>rtt(config-ldp-neig) # discovery targeted-hello holdtime <TIME></code>	<TIME> – время в секундах в интервале [3..65535]. Значение по умолчанию: 45.
3	В режиме конфигурации LDP-соседа задать Hello interval.	<code>rtt(config-ldp-neig) # discovery targeted- hello interval <TIME></code>	<TIME> – время в секундах в интервале [1..65535]. Значение по умолчанию: 5.
4	В режиме конфигурации LDP-соседа задать Keepalive holdtime.	<code>rtt(config-ldp-neig) # keepalive <TIME></code>	<TIME> – время в секундах в интервале [3..65535]. Значение по умолчанию: 180.

13.3.3. Пример настройки

Задача:

Переопределить параметры Hello holdtime (120 секунд) и Hello interval (30 секунд) для всего процесса targeted-LDP. Для соседа с адресом 4.4.4.4 установить Keepalive holdtime равным 150 секунд.

Решение:

RTT

```
RTT(config) # mpls
```

```
RTT(config-mpls)# ldp
RTT(config-ldp)# discovery targeted-hello holdtime 40
RTT(config-ldp)# discovery targeted-hello interval 10
RTT(config-ldp)# neighbor 4.4.4.4
RTT(config-ldp-neig)# keepalive 150
```

Проверка:

Для просмотра hello-параметров targeted LDP-сессии:

RTT

```
RTT1# sh mpls ldp discovery detailed
...
Targeted hellos:
  1.1.1.1 -> 4.4.4.4:
    Hello interval:      10 seconds
    Transport IP address: 1.1.1.1
    LDP ID:              4.4.4.4
    Source IP address:   4.4.4.4
    Transport IP address: 4.4.4.4
    Hold time:           40 seconds
    Proposed hold time:  40/45 (local/peer) seconds
```

Для просмотра параметров установленной TCP-сессии:

RTT

```
RTT# sh mpls ldp neighbor 4.4.4.4
Peer LDP ID: 4.4.4.4; Local LDP ID 1.1.1.1
State: Operational
TCP connection: 4.4.4.4:34879 - 1.1.1.1:646
Messages sent/received: 11/11
Uptime: 00:01:05
Peer holdtime: 150
Keepalive interval: 50
LDP discovery sources:
  1.1.1.1 -> 4.4.4.4:
    Hello interval: 10 seconds
    Holdtime: 40 seconds
...
```

13.4. Настройка фильтрации LDP-меток

По умолчанию маршрутизаторы выделяют на каждый FEC отдельную метку. Существуют сценарии, когда необходимо выделять MPLS-метки только для определенных FEC. Ниже рассмотрены существующие возможности для реализации фильтрации LDP-меток.

13.4.1. Метод на основе Advertise-labels

Данный метод позволяет аллоцировать метки протоколом LDP только на префиксы, описанные в соответствующей object-group. Отличительной особенностью данного метода является то, что префиксы должны иметь точное совпадение с маршрутом из FIB.

13.4.1.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Настроить протокол LDP (см. раздел Настройка протокола LDP).		
2	Создать object-group типа network.	<code>rtt(config)# object-group network <NAME></code>	<NAME> – имя конфигурируемого списка подсетей, задаётся строкой до 31 символа.
3	Описать префиксы, для которых будут назначаться метки.	<code>rtt(config-object-group-network)# ip prefix <ADDR/LEN> [unit <ID>]</code>	<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]. <ID> – номер юнита, принимает значения [1..4].
4	В контексте настройки LDP применить созданную object-group.	<code>rtt(config-ldp)# advertise-labels <NAME></code>	<NAME> – имя конфигурируемого списка подсетей, задаётся строкой до 31 символа.

Метки будут выделяться ТОЛЬКО на описанные в object-group подсети, независимо от того, как они были изучены (connected, local, IGP и т. д.).

Данный функционал поддержан для протокола IPv4.

13.4.1.2. Пример настройки



Задача:

Назначить MPLS-метки только FEC 10.10.0.2/32 и 10.10.0.1/32.

Решение:

На RTT_A и RTT_B создадим object-group ADV_LABELS типа network и добавим в нее префиксы 10.10.0.1/32 и 10.10.0.2/32 соответственно:

RTT_A

```
rtt(config)# object-group network ADV_LABELS
rtt(config-object-group-network)# ip prefix 10.10.0.1/32
rtt(config-object-group-network)# ip prefix 10.10.0.2/32
```

RTT_B

```
rtt(config)# object-group network ADV_LABELS
rtt(config-object-group-network)# ip prefix 10.10.0.1/32
rtt(config-object-group-network)# ip prefix 10.10.0.2/32
```

Применим созданную object-group на обоих маршрутизаторах:

RTT_A и RTT_B

```
rtt(config)# mpls
rtt(config-ldp)# ldp
rtt(config-ldp)# advertise-labels ADV_LABELS
```

Проверка:

На RTT_B убедимся, что метка назначена для соответствующих префиксов:

```
rtt# sh mpls ldp bindings 10.10.0.1/32
10.10.0.1/32
local label: exp-null
remote label: 75 lsr: 172.16.0.1
```

И не назначена для 192.168.2.0/24:

```
rtt# sh mpls ldp bindings 192.168.2.0/24
rtt#
```

13.4.2. Метод на основе Prefix-list

Данный метод позволяет управлять выделением меток для протокола LDP на основе подсетей, описанный в соответствующем Prefix-list. В отличие от метода на основе Advertise-labels, гибкость функционала Prefix-list позволяет задать диапазон (префикс), для подсетей которого будет аллоцирована метка. Данный вариант настройки является более гибким и масштабируемым.

13.4.2.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Настроить протокол LDP (см. раздел Настройка протокола LDP).		

Шаг	Описание	Команда	Ключи
2	Создать prefix-list.	<code>rtt(config)# ip prefix-list <NAME></code>	<NAME> – имя конфигурируемого списка подсетей, задаётся строкой до 31 символа.
3	Описать подсеть, для адресов которой будут выделяться метки.	<code>rtt(config-object-group-network)# ip prefix <ADDR/LEN> [{ eq <LEN> le <LEN> ge <LEN> [le <LEN>] }]</code>	<p><ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><LEN> – длина префикса, принимает значения [1..32];</p> <p>eq – при указании команды длина префикса должна соответствовать указанной;</p> <p>le – при указании команды длина префикса должна быть меньше либо соответствовать указанной;</p> <p>ge – при указании команды длина префикса должна быть больше либо соответствовать указанной.</p>
4	В контексте настройки LDP применить prefix-list для соответствующей address-family.	<code>rtt(config-ldp-af-ipv4)# prefix-list <NAME></code>	<NAME> – имя конфигурируемого списка, задаётся строкой до 31 символа.

13.4.2.2. Пример настройки



Задача:

Для организации сервисов L2VPN и L3VPN выделена подсеть 10.10.0.0/24. Необходимо средствами протокола LDP назначить транспортные MPLS-метки для всех адресов (подсетей/32), входящих в выделенный диапазон.

Решение:

На RTT_A и RTT_B создадим prefix-list и опишем необходимый для сервисов диапазон адресов:

RTT_A

```
rtt(config)# ip prefix-list LDP_ALLOCATE
```

```
rtt(config-pl)# permit 10.10.0.0/24 eq 32
```

RTT_B

```
rtt(config)# ip prefix-list LDP_ALLOCATE  
rtt(config-pl)# permit 10.10.0.0/24 eq 32
```

Применим созданный prefix-list на обоих маршрутизаторах:

RTT_A и RTT_B

```
rtt(config)# mpls  
rtt(config-ldp)# ldp  
rtt(config-ldp)# address-family ipv4  
rtt(config-ldp-af-ipv4)# prefix-list LDP_ALLOCATE  
rtt(config-ldp-af-ipv4)# do commit  
rtt(config-ldp-af-ipv4)# do confirm
```

Проверка:

На примере RTT_B убедимся, что метка назначена для соответствующих префиксов:

```
rtt# sh mpls ldp bindings 10.10.0.1/32  
10.10.0.1/32  
local label: exp-null  
remote label: 45 lsr: 172.16.0.1
```

И не назначена для 192.168.2.0/24:

```
rtt# sh mpls ldp bindings 192.168.2.0/24  
rtt#
```

13.5. Настройка сервиса L2VPN Martini mode

L2VPN позволяет организовать передачу ethernet-фреймов через MPLS-домен. Выделение и распространение туннельных меток в данном режиме осуществляется по средствам протокола LDP. В реализации L2VPN можно условно выделить два случая:

1. P2P – туннель, создаваемый по схеме «точка-точка».
2. VPLS – туннель, создаваемый по схеме «точка-многоточка».

В обоих случаях для передачи ethernet-фреймов между маршрутизаторами создается виртуальный канал (далее pseudo-wire). Для согласования параметров pseudo-wire, а также для выделения и передачи туннельных меток между маршрутизаторами устанавливается LDP-сессия в targeted-режиме.

13.5.1. Алгоритм настройки L2VPN VPWS

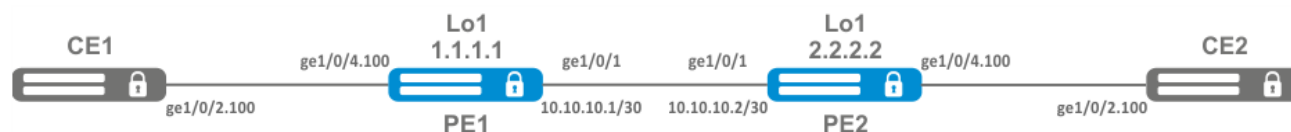
Шаг	Описание	Команда	Ключи
1	Настроить протокол LDP (см. раздел Настройка протокола LDP).		
2	Создать pw-class в системе и осуществить переход в режим настройки параметров pw-class.	<code>rtt(config-l2vpn)# pw-class <WORD></code>	<WORD> – имя pw-class длиной [1..31] символов.
3	Добавить описание для pw-class (необязательно).	<code>rtt(config-l2vpn-pw-class)# description <LINE></code>	<LINE> – описание. Задается в виде строки длиной [1..255] символов.
4	Установить значение MTU для pseudo-wire входящих в pw-class (необязательно).	<code>rtt(config-l2vpn-pw-class)# encapsulation mpls mtu <MTU></code>	<MTU> – значение MTU, принимает значение в диапазоне [552..10000] Значение по умолчанию: 1500.
5	Отключить обмен status-tlv сообщениями (необязательно).	<code>rtt(config-l2vpn-pw-class)# encapsulation mpls status-tlv disable</code>	Значение по умолчанию: status-tlv enable.
6	Создать p2p-туннель в системе и осуществить переход в режим настройки параметров p2p-туннеля.	<code>rtt(config-l2vpn)# p2p <NAME></code>	<NAME> – имя p2p-сервиса, задается строкой до 31 символа.
7	Задать Attached Circuit интерфейс.	<code>rtt(config-l2vpn-p2p)# interface { <IF> <TUN> }</code>	<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора ; <TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора .
8	Включить p2p-туннель.	<code>rtt(config-l2vpn-p2p)# enable</code>	
9	Задать транспортный режим (необязательно).	<code>rtt(config-l2vpn-p2p)# transport-mode { ethernet vlan }</code>	<ethernet> – режим, при котором при входе в pseudo-wire из заголовка удаляется 802.1Q тег; <vlan> – режим, при котором 802.1Q тег может быть сохранен при передаче через pseudo-wire. Значение по умолчанию: ethernet.

Шаг	Описание	Команда	Ключи
10	Создать pseudo-wire и осуществить переход в режим настройки его параметров.	<code>rtt(config-l2vpn-p2p) # pw <PW_ID> <LSR_ID></code>	<PW_ID> – идентификатор pseudowire, задается в виде числа в диапазоне [1..4294967295] <LSR_ID> – идентификатор LSR, до которого строится pseudo-wire, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]
11	Добавить описание для pseudo-wire (необязательно).	<code>rtt(config-l2vpn-pw) # description <LINE></code>	<LINE> – описание. Задается в виде строки длиной [1..255] символов.
12	Задать pw-class для pseudo-wire.	<code>rtt(config-l2vpn-pw) # pw-class <WORD></code>	<WORD> – имя pw-class длиной [1..31] символов.
13	Задать адрес LSR до которого устанавливается pseudo-wire (не обязательно, если neighbor address совпадает с LSR_ID).	<code>rtt(config-l2vpn-pw) # neighbor-address <ADDR></code>	<ADDR> – IP-адрес маршрутизатора, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
14	Включить pseudo-wire.	<code>rtt(config-l2vpn-pw) # enable</code>	
В случае если необходимо изменить параметры по умолчанию для targeted LDP-сессии, обратитесь к разделу Конфигурирование параметров сессии в протоколе targeted-LDP .			

13.5.2. Пример настройки L2VPN VPWS

Задача:

Настроить l2vpn таким образом, чтобы интерфейс ge1/0/2.100 маршрутизатора CE1 и интерфейс ge1/0/2.100 маршрутизатора CE2 работали в рамках одного широковещательного домена.



Решение:

Предварительно нужно:

- Включить поддержку Jumbo-фреймов с помощью команды **system jumbo-frames** (для вступления изменений в силу требуется перезагрузка устройства);
- Настроить IP-адреса на интерфейсах согласно схеме сети, приведенной на рисунке выше;
- Организовать обмен маршрутами между PE1 и PE2 при помощи IGP-протокола (OSPF, IS-IS, RIP).

На маршрутизаторе PE1 создадим саб-интерфейс, на который будем принимать трафик от CE1:

```
PE1# configure
PE1(config)# interface gigabitethernet 1/0/4.100
PE1(config-if-sub)# exit
```

Выставим на интерфейсе в сторону PE2 значение MTU равным 9600, для того чтобы избежать ситуации с превышением MTU после инкапсуляции MPLS-заголовка, а также отключим межсетевой экран:

```
PE1#(config)# interface gigabitethernet 1/0/1
PE1(config-if-gi)# mtu 9600
PE1(config-if-gi)# ip firewall disable
PE1(config-if-gi)# exit
```

Разрешим прием пакетов с MPLS-заголовком на интерфейсе в сторону MPLS-сети (в данном примере интерфейс в сторону PE2):

```
PE1(config)# mpls
PE1(config-mpls)# forwarding interface gigabitethernet 1/0/1
```

Настроим протокол LDP и включим обнаружение соседей на интерфейсе в сторону PE2:

```
PE1(config-mpls)# ldp
PE1(config-ldp)# router-id 1.1.1.1
PE1(config-ldp)# address-family ipv4
PE1(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
PE1(config-ldp-af-ipv4-if)# exit
PE1(config-ldp-af-ipv4)# transport-address 1.1.1.1
PE1(config-ldp-af-ipv4)# exit
PE1(config-ldp)# enable
PE1(config-ldp)# exit
```

Создадим pw-class, на основе которого в дальнейшем будет создан виртуальный канал (pw). Так как в данном примере на pw будут применяться параметры по умолчанию, достаточно будет указать имя класса:

```
PE1(config-mpls)# l2vpn
PE1(config-l2vpn)# pw-class for_p2p_VLAN100
PE1(config-l2vpn-pw-class)# exit
```

Создадим новый l2vpn типа p2p и добавим pw до маршрутизатора PE2, идентификатор pw для удобства возьмем равным VID (в данном случае равным 100):

```
PE1(config-l2vpn)# p2p to_PE2_VLAN100
PE1(config-l2vpn-p2p)# interface gigabitethernet 1/0/4.100
PE1(config-l2vpn-p2p)# pw 100 2.2.2.2
PE1(config-l2vpn-pw)# pw-class for_p2p_VLAN100
PE1(config-l2vpn-pw)# enable
PE1(config-l2vpn-pw)# exit
PE1(config-l2vpn-p2p)# enable
PE1(config-l2vpn-p2p)# end
```

Применим конфигурацию:

```
PE1# commit
```

```
PE1# confirm
```

Проведем настройку маршрутизатора PE2 по аналогии с PE1:

```
PE2# configure
PE2(config)# interface gigabitethernet 1/0/4.100
PE2(config-if-sub)# exit
PE2#(config)# interface gigabitethernet 1/0/1
PE2(config-if-gi)# mtu 9600
PE2(config-if-gi)# ip firewall disable
PE2(config-if-gi)# exit
PE2(config)# mpls
PE2(config-mpls)# forwarding interface gigabitethernet 1/0/1
PE2(config-mpls)# ldp
PE2(config-ldp)# router-id 2.2.2.2
PE2(config-ldp)# address-family ipv4
PE2(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
PE2(config-ldp-af-ipv4-if)# exit
PE2(config-ldp-af-ipv4)# transport-address 2.2.2.2
PE2(config-ldp-af-ipv4)# exit
PE2(config-ldp)# enable
PE2(config-ldp)# exit
PE2(config-mpls)# l2vpn
PE2(config-l2vpn)# pw-class for_p2p_VLAN100
PE2(config-l2vpn-pw-class)# exit
PE2(config-l2vpn)# p2p to_PE1_VLAN100
PE2(config-l2vpn-p2p)# interface gigabitethernet 1/0/4.100
PE2(config-l2vpn-p2p)# pw 100 1.1.1.1
PE2(config-l2vpn-pw)# pw-class for_p2p_VLAN100
PE2(config-l2vpn-pw)# enable
PE2(config-l2vpn-pw)# exit
PE2(config-l2vpn-p2p)# enable
PE2(config-l2vpn-p2p)# end
PE2# commit
PE2# confirm
```

Убедимся в установлении соседства по протоколу LDP и выведем информацию по статусу виртуального канала (pseudowire) между PE1 и PE2:

```
PE2# show mpls ldp neighbor
Peer LDP ID: 1.1.1.1; Local LDP ID 2.2.2.2
  State: Operational
  TCP connection: 1.1.1.1:646 - 2.2.2.2:34625
  Messages sent/received: 12/12
  Uptime: 00:03:50
  LDP discovery sources:
    2.2.2.2 -> 1.1.1.1
```

```
PE2# show mpls l2vpn pseudowire
```

Neighbor	PW ID	Type	Status
1.1.1.1	100	Ethernet	Up

Соседство по протоколу LDP установлено, pseudowire перешел в статус 'UP'. Настройка l2vpn типа p2p завершена.

13.5.3. Алгоритм настройки L2VPN VPLS

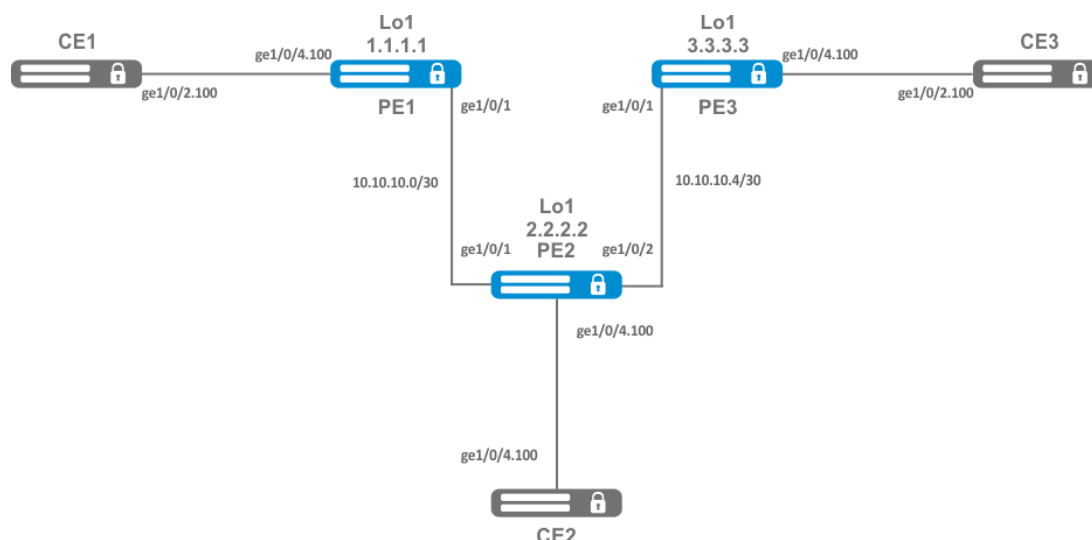
Шаг	Описание	Команда	Ключи
1	Настроить протокол LDP (см. раздел Настройка протокола LDP).		
2	Создать сетевой мост в системе без указания IP-адреса (см. раздел Настройка Bridge).		
3	Создать pw-class в системе и осуществить переход в режим настройки параметров pw-class.	<code>rtt(config-l2vpn)# pw-class <WORD></code>	<WORD> – имя pw-class длиной [1..31] символов.
4	Добавить описание для pw-class (необязательно).	<code>rtt(config-l2vpn-pw-class)# description <LINE></code>	<LINE> – описание. Задается в виде строки длиной [1..255] символов.
5	Установить значение MTU для pseudo-wire входящих в pw-class (необязательно).	<code>rtt(config-l2vpn-pw-class)# encapsulation mpls mtu <MTU></code>	<MTU> – значение MTU, принимает значение в диапазоне [552..10000]. Значение по умолчанию: 1500.
6	Отключить обмен status-tlv сообщениями (необязательно).	<code>rtt(config-l2vpn-pw-class)# encapsulation mpls status-tlv disable</code>	Значение по умолчанию: status-tlv enable.
7	Создать VPLS-домен в системе и осуществить переход в режим настройки параметров VPLS-домена.	<code>rtt(config-l2vpn)# vpls <NAME></code>	<NAME> – имя p2p-сервиса, задается строкой до 31 символа.
8	Включить VPLS-туннель.	<code>rtt(config-l2vpn-vpls)# enable</code>	
9	Добавить бридж-домен.	<code>rtt (config-l2vpn-vpls)# bridge-group <ID></code>	<ID> – идентификатор бридж-домена, задается в виде числа в диапазоне [1..250].
10	Задать транспортный режим (необязательно).	<code>rtt(config-l2vpn-vpls)# transport-mode { ethernet vlan }</code>	<ethernet> – режим, при котором при входе в pseudo-wire из заголовка удаляется 802.1Q тег; <vlan> – режим, при котором 802.1Q тег может быть сохранен при передаче через pseudo-wire. Значение по умолчанию: ethernet.
11	Создать pseudo-wire и осуществить переход в режим настройки его параметров.	<code>rtt(config-l2vpn-vpls)# pw <PW_ID> <LSR_ID></code>	<PW_ID> – идентификатор psewdowire, задается в виде числа в диапазоне [1..4294967295] <LSR_ID> – идентификатор LSR до которого строится pseudo-wire, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
12	Добавить описание для pseudo-wire (необязательно).	<code>rtt(config-l2vpn-pw)# description <LINE></code>	<LINE> – описание. Задается в виде строки длиной [1..255] символов.

Шаг	Описание	Команда	Ключи
13	Задать pw-class для pseudo-wire.	<code>rtt(config-l2vpn-pw) # pw-class <WORD></code>	<WORD> – имя pw-class длиной [1..31] символов.
14	Задать адрес LSR до которого устанавливается pseudo-wire (необязательно, если neighbor address совпадает с LSR_ID).	<code>rtt(config-l2vpn-pw) # neighbor-address <ADDR></code>	<ADDR> – IP-адрес маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
15	Включить pseudo-wire.	<code>rtt(config-l2vpn-pw) # enable</code>	
16	В случае если топология создаваемого VPLS-домена требует установить более одного pseudo-wire, повторить шаги с 10 по 14.		
17	В случае если необходимо изменить параметры по умолчанию для targeted LDP-сессии, обратитесь к разделу Конфигурирование параметров сессии в протоколе targeted-LDP .		

13.5.4. Пример настройки L2VPN VPLS

Задача:

Настроить l2vpn таким образом, чтобы маршрутизаторы CE1, CE2, CE3 имели L2-связность через интерфейсы gi1/0/2.100 и gi1/0/4 (CE2).



Решение:

Предварительно необходимо:

- Включить поддержку Jumbo-фреймов с помощью команды **system jumbo-frames** (для вступления изменений в силу требуется перезагрузка устройства);
- Настроить IP-адреса на интерфейсах согласно схеме сети, приведенной на рисунке выше;
- Организовать обмен маршрутами между PE1, PE2 и PE3 при помощи IGP протокола (OSPF, IS-IS).

На маршрутизаторе PE1 создадим бридж-группу и включим ее:

```
PE1# configure
PE1(config)# bridge 10
PE1(config-bridge)# enable
PE1(config-bridge)# exit
```

Интерфейсе в сторону CE1 включим в созданную бридж-группу:

```
PE1(config)# interface gigabitethernet 1/0/4.100
PE1(config-if-sub)# bridge-group 10
PE1(config-if-sub)# exit
```

Выставим на интерфейсе в сторону PE2 значение MTU равным 9600, для того чтобы избежать ситуации с превышением MTU после инкапсуляции MPLS-заголовка, а также отключим межсетевой экран:

```
PE1#(config)# interface gigabitethernet 1/0/1
PE1(config-if-gi)# mtu 9600
PE1(config-if-gi)# ip firewall disable
PE1(config-if-gi)# exit
```

Разрешим прием пакетов с MPLS-заголовком на интерфейсе в сторону MPLS-сети (в данном примере интерфейс в сторону PE2):

```
PE1(config)# mpls
PE1(config-mpls)# forwarding interface gigabitethernet 1/0/1
```

Настроим протокол LDP и включим обнаружение соседей на интерфейсе в сторону PE2:

```
PE1(config-mpls)# ldp
PE1(config-ldp)# router-id 1.1.1.1
PE1(config-ldp)# address-family ipv4
PE1(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
PE1(config-ldp-af-ipv4-if)# exit
PE1(config-ldp-af-ipv4)# transport-address 1.1.1.1
PE1(config-ldp-af-ipv4)# exit
PE1(config-ldp)# enable
PE1(config-ldp)# exit
```

Создадим pw-class, на основе которого в дальнейшем будет созданы виртуальные каналы (pw). Так как в данном примере на pw будут применяться параметры по умолчанию, достаточно будет указать имя класса:

```
PE1(config-mpls)# l2vpn
PE1(config-l2vpn)# pw-class for_vpls1
PE1(config-l2vpn-pw-class)# exit
```

Создадим новый l2vpn типа vpls и добавим pw до маршрутизаторов PE2 и PE3, идентификатор pw для удобства возьмем равным VID (в данном случае равным 100):

```
PE1(config-l2vpn)# vpls vpls1
PE1(config-l2vpn-vpls)# bridge-group 10
```

```
PE1(config-l2vpn-vpls)# pw 100 2.2.2.2
PE1(config-l2vpn-pw)# pw-class for_vpls1
PE1(config-l2vpn-pw)# enable
PE1(config-l2vpn-pw)# exit
PE1(config-l2vpn-vpls)# pw 100 3.3.3.3
PE1(config-l2vpn-pw)# pw-class for_vpls1
PE1(config-l2vpn-pw)# enable
PE1(config-l2vpn-pw)# exit
PE1(config-l2vpn-vpls)# enable
PE1(config-l2vpn-vpls)# end
```

Применим созданную конфигурацию:

```
PE1# commit
PE1# confirm
```

Проведем настройку маршрутизатора PE2 и PE3 по аналогии с PE1:

```
PE2# configure
PE2(config)# bridge 10
PE2(config-bridge)# enable
PE2(config-bridge)# exit
PE2(config)# interface gigabitethernet 1/0/4.100
PE2(config-if-sub)# bridge-group 10
PE2(config-if-sub)# exit
PE2(config)# interface gigabitethernet 1/0/2
PE2(config-if-gi)# mtu 9600
PE2(config-if-gi)# ip firewall disable
PE2(config-if-gi)# exit
PE2(config)# mpls
PE2(config-mpls)# forwarding interface gigabitethernet 1/0/1
PE2(config-mpls)# forwarding interface gigabitethernet 1/0/2
PE2(config-mpls)# ldp
PE2(config-ldp)# enable
PE2(config-ldp)# router-id 2.2.2.2
PE2(config-ldp)# address-family ipv4
PE2(config-ldp-af-ipv4)# transport-address 2.2.2.2
PE2(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
PE2(config-ldp-af-ipv4-if)# exit
PE2(config-ldp-af-ipv4)# interface gigabitethernet 1/0/2
PE2(config-ldp-af-ipv4-if)# exit
PE2(config-ldp-af-ipv4)# exit
PE2(config-ldp)# exit
PE2(config-mpls)# l2vpn
PE2(config-l2vpn)# pw-class for_vpls1
PE2(config-l2vpn-pw-class)# exit
PE2(config-l2vpn)# vpls vpls1
PE2(config-l2vpn-vpls)# enable
PE2(config-l2vpn-vpls)# bridge-group 10
PE2(config-l2vpn-vpls)# pw 100 1.1.1.1
PE2(config-l2vpn-pw)# pw-class for_vpls1
PE2(config-l2vpn-pw)# enable
PE2(config-l2vpn-pw)# exit
PE2(config-l2vpn-vpls)# pw 100 3.3.3.3
PE2(config-l2vpn-pw)# pw-class for_vpls1
PE2(config-l2vpn-pw)# enable
PE2(config-l2vpn-pw)# end
```

```
PE2# commit
PE2# confirm
PE3(config)# bridge 10
PE3(config-bridge)# enable
PE3(config-bridge)# exit
PE3(config)# interface gigabitethernet 1/0/4.100
PE3(config-if-sub)# bridge-group 10
PE3(config-if-sub)# exit
PE3(config)# interface gigabitethernet 1/0/1
PE3(config-if-gi)# mtu 9600
PE3(config-if-gi)# ip firewall disable
PE3(config-if-gi)# exit
PE3(config)# mpls
PE3(config-mpls)# forwarding interface gigabitethernet 1/0/1
PE3(config-mpls)# exit
PE3(config)# mpls
PE3(config-mpls)# ldp
PE3(config-ldp)# enable
PE3(config-ldp)# router-id 3.3.3.3
PE3(config-ldp)# address-family ipv4
PE3(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
PE3(config-ldp-af-ipv4-if)# exit
PE3(config-ldp-af-ipv4)# transport-address 3.3.3.3
PE3(config-ldp-af-ipv4)# exit
PE3(config-ldp)# exit
PE3(config-mpls)# l2vpn
PE3(config-l2vpn)# pw-class for_vpls
PE3(config-l2vpn-pw-class)# exit
PE3(config-l2vpn)# vpls vpls1
PE3(config-l2vpn-vpls)# enable
PE3(config-l2vpn-vpls)# bridge-group 10
PE3(config-l2vpn-vpls)# pw 100 2.2.2.2
PE3(config-l2vpn-pw)# pw-class for_vpls
PE3(config-l2vpn-pw)# enable
PE3(config-l2vpn-pw)# exit
PE3(config-l2vpn-vpls)# pw 100 1.1.1.1
PE3(config-l2vpn-pw)# pw-class for_vpls
PE3(config-l2vpn-pw)# enable
PE3(config-l2vpn-pw)# end
PE3# commit
PE3# confirm
```

Убедимся в установлении соседства по протоколу LDP и выведем информацию по статусу виртуального канала (pseudowire) между PE1, PE2 и PE3:

```
PE3# show mpls ldp neighbor
Peer LDP ID: 1.1.1.1; Local LDP ID 3.3.3.3
  State: Operational
  TCP connection: 1.1.1.1:646 - 3.3.3.3:45979
  Messages sent/received: 22/22
  Uptime: 00:13:16
  LDP discovery sources:
    3.3.3.3 -> 1.1.1.1
Peer LDP ID: 2.2.2.2; Local LDP ID 3.3.3.3
  State: Operational
  TCP connection: 2.2.2.2:646 - 3.3.3.3:59627
  Messages sent/received: 22/22
  Uptime: 00:13:20
```

```
LDP discovery sources:
  3.3.3.3 -> 2.2.2.2
  gigabitethernet 1/0/1
PE3# show mpls l2vpn pseudowire
```

Neighbor	PW ID	Type	Status
1.1.1.1	100	Ethernet	Up
2.2.2.2	100	Ethernet	Up

Соседство по протоколу LDP установлено, pseudowire перешел в статус 'UP'. Настройка l2vpn завершена.

13.6. Настройка сервиса L2VPN Kompella mode

В отличие от Martini mode, где вся работа ложится на LDP, в данном режиме LDP отводится только работа с транспортными метками. Автообнаружение и построение псевдо-провода возложено на протокол BGP.

13.6.1. Алгоритм настройки L2VPN VPLS

Шаг	Описание	Команда	Ключи
1	Настроить протокол LDP (см. раздел Настройка протокола LDP).		
2	Создать сетевой мост в системе без указания IP-адреса (см. раздел Настройка Bridge).		
3	Создать VPLS-домен в системе и осуществить переход в режим настройки параметров VPLS-домена.	<code>rtt(config-l2vpn)# vpls <NAME></code>	<NAME> – имя p2p-сервиса, задается строкой до 31 символа.
4	Включить VPLS-туннель.	<code>rtt(config-l2vpn- vpls)# enable</code>	
5	Добавить бридж-домен.	<code>rtt(config-l2vpn- vpls)# bridge-group <ID></code>	<ID> – идентификатор бридж-домена, задается в виде числа в диапазоне [1..250].
6	Перейти в контекст настройки autodiscovery bgp.	<code>rtt(config-l2vpn- vpls)# autodiscovery bgp</code>	

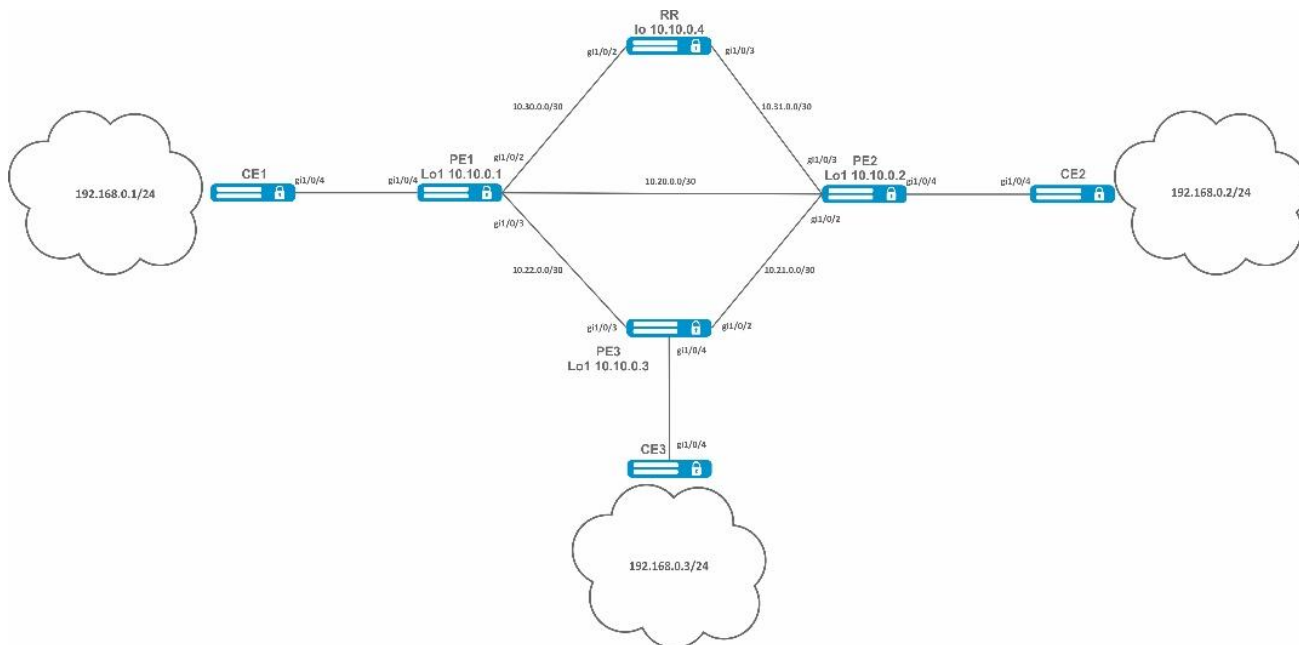
Шаг	Описание	Команда	Ключи
7	Указать route distinguisher для данного экземпляра VPLS.	<code>rtt(config-bgp) # rd <RD></code>	<p><RD> – значение Route distinguisher, задается в одном из следующих видов:</p> <ul style="list-style-type: none"> • <ASN>:<nn> – где <ASN> – принимает значение [1..65535], nn – принимает значение [1..65535]; • <ADDR>:<nn> – где <ADDR> имеет вид – AAA.BBB.CCC.DDD/EE и AAA-DDD принимают значения [0..255], а nn – принимает значение [1..65535]; • <4ASN>:<nn> – где <4ASN> – принимает значение [1..4294967295], nn – принимает значение [1..65535].
8	Указать route target import для данного экземпляра VPLS.	<code>rtt(config-bgp) # route-target import <RT></code>	<p><RT> – значение route-target, задается в одном из следующих видов:</p> <ul style="list-style-type: none"> • <ASN>:<nn> – где <ASN> – принимает значение [1..65535], nn – принимает значение [1..65535]; • <ADDR>:<nn> – где <ADDR> имеет вид – AAA.BBB.CCC.DDD/EE и AAA-DDD принимают значения [0..255], а nn – принимает значение [1..65535]; • <4ASN>:<nn> – где <4ASN> – принимает значение [1..4294967295], nn – принимает значение [1..65535].
9	Указать route target export для данного экземпляра VPLS.	<code>rtt(config-bgp) # route-target export <RT></code>	<p><RT> – значение route-target, задается в одном из следующих видов:</p> <ul style="list-style-type: none"> • <ASN>:<nn> – где <ASN> – принимает значение [1..65535], nn – принимает значение [1..65535]; • <ADDR>:<nn> – где <ADDR> имеет вид – AAA.BBB.CCC.DDD/EE и AAA-DDD принимают значения [0..255], а nn – принимает значение [1..65535]; • <4ASN>:<nn> – где <4ASN> – принимает значение [1..4294967295], nn – принимает значение [1..65535].
10	Указать ve id.	<code>rtt(config-bgp) # ve id <ID></code>	<p><ID> – идентификатор экземпляра VPLS, задается в виде числа в диапазоне [1..16384].</p>

Шаг	Описание	Команда	Ключи
11	Указать vpn id.	<code>rtt (config-bgp) # vpn id <ID></code>	<ID> – идентификатор VPN, задается в виде числа в диапазоне [1..4294967295].
12	Указать ve range (необязательно).	<code>rtt (config-bgp) # ve range <RANGE></code>	<RANGE> – диапазон идентификаторов пограничных устройств VPLS [8..100].
13	Указать mtu (необязательно).	<code>rtt (config-bgp) # mtu <VALUE></code>	<VALUE> – значение MTU [552..10000].
14	Включить игнорирование типа инкапсуляции (необязательно).	<code>rtt (config-bgp) # ignore encapsulation-mismatch</code>	
15	Включить игнорирование значений MTU (необязательно).	<code>rtt (config-bgp) # ignore mtu-mismatch</code>	
16	В контексте настройки address-family l2vpn vpls протокола BGP включить передачу расширенных атрибутов.	<code>rtt(config-bgp-neighbor-af)# send-community extended</code>	

13.6.2. Пример настройки L2VPN VPLS

Задача:

Настроить L2VPN-сервис: все CE-устройства должны работать в рамках одного широковещательного домена.



Решение:

Предварительно необходимо:

- Включить поддержку Jumbo-фреймов с помощью команды **system jumbo-frames** (для вступления изменений в силу требуется перезагрузка устройства);
- Настроить IP-адреса на интерфейсах согласно схеме сети, приведенной на рисунке выше;
- Организовать обмен маршрутами между PE1, PE2, PE3 и RR при помощи IGP-протокола (OSPF, IS-IS).

Настроим маршрутизатор RR:

```
hostname RR

system jumbo-frames

router ospf 1
area 0.0.0.0
enable
exit
enable
exit

interface gigabitethernet 1/0/2
mtu 9500
ip firewall disable
ip address 10.30.0.2/30
ip ospf instance 1
ip ospf
exit
interface gigabitethernet 1/0/3
mtu 9500
ip firewall disable
ip address 10.31.0.2/30
ip ospf instance 1
ip ospf
exit
interface loopback 1
ip address 10.10.0.4/32
ip ospf instance 1
ip ospf
exit
mpls
ldp
router-id 10.10.0.4
address-family ipv4
interface gigabitethernet 1/0/2
exit
interface gigabitethernet 1/0/3
exit
exit
enable
exit
forwarding interface gigabitethernet 1/0/2
```



```
forwarding interface gigabitethernet 1/0/3
exit
```

Настроим BGP Route Reflector для address family l2vpn:

```
RR(config)# router bgp 65500
RR(config-bgp)# router-id 10.10.0.4
RR(config-bgp)# neighbor 10.10.0.1
RR(config-bgp-neighbor)# remote-as 65500
RR(config-bgp-neighbor)# route-reflector-client
RR(config-bgp-neighbor)# update-source 10.10.0.4
RR(config-bgp-neighbor)# address-family l2vpn vpls
RR(config-bgp-neighbor-af)# send-community extended
RR(config-bgp-neighbor-af)# enable
RR(config-bgp-neighbor-af)# exit
RR(config-bgp-neighbor)# enable
RR(config-bgp-neighbor)# exit
RR(config-bgp)# neighbor 10.10.0.2
RR(config-bgp-neighbor)# remote-as 65500
RR(config-bgp-neighbor)# route-reflector-client
RR(config-bgp-neighbor)# update-source 10.10.0.4
RR(config-bgp-neighbor)# address-family l2vpn vpls
RR(config-bgp-neighbor-af)# send-community extended
RR(config-bgp-neighbor-af)# enable
RR(config-bgp-neighbor-af)# exit
RR(config-bgp-neighbor)# enable
RR(config-bgp-neighbor)# exit
RR(config-bgp)# neighbor 10.10.0.3
RR(config-bgp-neighbor)# remote-as 65500
RR(config-bgp-neighbor)# route-reflector-client
RR(config-bgp-neighbor)# update-source 10.10.0.4
RR(config-bgp-neighbor)# address-family l2vpn vpls
RR(config-bgp-neighbor-af)# send-community extended
RR(config-bgp-neighbor-af)# enable
RR(config-bgp-neighbor-af)# exit
RR(config-bgp-neighbor)# enable
RR(config-bgp-neighbor)# exit
RR(config-bgp)# enable
```

Настройка протокола BGP на PE-маршрутизаторах:

Предварительная конфигурация

```
hostname PE1

system jumbo-frames

router ospf 1
area 0.0.0.0
enable
exit
enable
exit

interface gigabitethernet 1/0/1
mtu 9500
```

```
ip firewall disable
ip address 10.20.0.1/30
ip ospf instance 1
ip ospfexit
interface gigabitethernet 1/0/2
mtu 9500
ip firewall disable
ip address 10.30.0.1/30
ip ospf instance 1
ip ospf
exitinterface gigabitethernet 1/0/3
mtu 9500
ip firewall disable
ip address 10.22.0.1/30
ip ospf instance 1
ip ospf
exit
interface loopback 1
ip address 10.10.0.1/32
ip ospf instance 1
ip ospf
exit
mpls
ldp
router-id 10.10.0.1
address-family ipv4
interface gigabitethernet 1/0/1
exit
interface gigabitethernet 1/0/2
exit
interface gigabitethernet 1/0/3
exit

exit

enable

exit
forwarding interface gigabitethernet 1/0/1
forwarding interface gigabitethernet 1/0/2
forwarding interface gigabitethernet 1/0/3
exit
```

Настройка протокола BGP:

```
PE1(config)# router bgp 65500
PE1(config-bgp)# neighbor 10.10.0.4
PE2(config-bgp)# router-id 10.10.0.1
PE1(config-bgp-neighbor)# remote-as 65500
PE1(config-bgp-neighbor)# update-source 10.10.0.1
PE1(config-bgp-neighbor)# address-family l2vpn vpls
PE1(config-bgp-neighbor-af)# send-community extended
PE1(config-bgp-neighbor-af)# enable
PE1(config-bgp-neighbor-af)# exit
PE1(config-bgp-neighbor)# enable
PE1(config-bgp-neighbor)# exit
```

```
PE1(config-bgp)# enable
PE1(config-bgp)# exit
```

Проверим, что BGP-сессия успешно установлена с RR:

```
PE1# show bgp neighbors
BGP neighbor is 10.10.0.4
BGP state: Established
Neighbor address: 10.10.0.4
Neighbor AS: 65500
Neighbor ID: 10.10.0.4
Neighbor caps: refresh enhanced-refresh restart-aware AS4
Session: internal multihop AS4
Source address: 10.10.0.1
Weight: 0
Hold timer: 110/180
Keepalive timer: 21/60
Uptime: 7375 s
```

Настройка BGP на PE2:

Предварительная конфигурация

```
hostname PE2

system jumbo-frames

router ospf 1
area 0.0.0.0
enable
exit
enable
exit
```

Предварительная конфигурация

```
interface gigabitethernet 1/0/1
mtu 9500
ip firewall disable
ip address 10.20.0.2/30
ip ospf instance 1
ip ospf
exit
interface gigabitethernet 1/0/2
mtu 9500
ip firewall disable
ip address 10.21.0.1/30
ip ospf instance 1
ip ospf
exit
interface gigabitethernet 1/0/3
mtu 9500
ip firewall disable
ip address 10.31.0.1/30
ip ospf instance 1
```

```
ip ospf
exit
interface loopback 1
ip address 10.10.0.2/32
ip ospf instance 1
ip ospf
exit
mpls
ldp
router-id 10.10.0.2
address-family ipv4
interface gigabitethernet 1/0/1
exit
interface gigabitethernet 1/0/2
exit
interface gigabitethernet 1/0/3
exit
exit

enable

exit
forwarding interface gigabitethernet 1/0/1
forwarding interface gigabitethernet 1/0/2
forwarding interface gigabitethernet 1/0/3
exit
PE2(config)# router bgp 65500
PE2(config-bgp)# router-id 10.10.0.2
PE2(config-bgp)# neighbor 10.10.0.4
PE2(config-bgp-neighbor)# remote-as 65500
PE2(config-bgp-neighbor)# update-source 10.10.0.2
PE2(config-bgp-neighbor)# address-family l2vpn vpls
PE2(config-bgp-neighbor-af)# send-community extended
PE2(config-bgp-neighbor-af)# enable
PE2(config-bgp-neighbor-af)# exit
PE2(config-bgp-neighbor)# enable
PE2(config-bgp-neighbor)# exit
PE2(config-bgp)# enable
PE2(config-bgp)# exit
```

Убедимся, что сессия с RR поднялась успешно:

```
PE2# show bgp neighbors
BGP neighbor is 10.10.0.4
BGP state: Established
Neighbor address: 10.10.0.4
Neighbor AS: 65500
Neighbor ID: 10.10.0.4
Neighbor caps: refresh enhanced-refresh restart-aware AS4
Session: internal multihop AS4
Source address: 10.10.0.2
Weight: 0
Hold timer: 113/180
Keepalive timer: 56/60
Uptime: 47 s
```

Настройка BGP на PE3:

Предварительная конфигурация

```
hostname PE3

system jumbo-frames

router ospf 1
area 0.0.0.0
enable
exit
enable
exit

interface gigabitethernet 1/0/2
mtu 9500
ip firewall disable
ip address 10.21.0.2/30
ip ospf instance 1
ip ospf
exit
interface gigabitethernet 1/0/3
mtu 9500
ip firewall disable
ip address 10.22.0.2/30
ip ospf instance 1
ip ospf
exit
interface loopback 1
ip address 10.10.0.3/24
ip ospf instance 1
ip ospf
exit
mpls
ldp
router-id 10.10.0.3
address-family ipv4
interface gigabitethernet 1/0/2
exit
interface gigabitethernet 1/0/3
exit
exit
enable
exit
forwarding interface gigabitethernet 1/0/2
forwarding interface gigabitethernet 1/0/3
exit
PE3(config)# router bgp 65500
PE3(config-bgp)# router-id 10.10.0.3
PE3(config-bgp)# neighbor 10.10.0.4
PE3(config-bgp-neighbor)# remote-as 65500
PE3(config-bgp-neighbor)# update-source 10.10.0.3
PE3(config-bgp-neighbor)# address-family l2vpn vpls
PE3(config-bgp-neighbor-af)# send-community extended
PE3(config-bgp-neighbor-af)# enable
PE3(config-bgp-neighbor-af)# exit
PE3(config-bgp-neighbor)# enable
PE3(config-bgp-neighbor)# exit
```

```
PE3(config-bgp)# enable
PE3(config-bgp)# exit
```

Проверим, что сессия BGP установлена успешно:

```
PE3# show bgp neighbors
BGP neighbor is 10.10.0.4
BGP state: Established
Neighbor address: 10.10.0.4
Neighbor AS: 65500
Neighbor ID: 10.10.0.4
Neighbor caps: refresh enhanced-refresh restart-aware AS4
Session: internal multihop AS4
Source address: 10.10.0.3
Weight: 0
Hold timer: 141/180
Keepalive timer: 27/60
Uptime: 77 s
```

Следующим этапом на каждом PE-маршрутизаторе создадим бридж-домен и включим в него интерфейс (Attachment circuit, AC), смотрящий в сторону CE:

PE1:

```
PE1(config)# bridge 1
PE1(config-bridge)# enable
PE1(config-bridge)# exit
PE1(config)# interface gigabitethernet 1/0/4
PE1(config-if-gi)# mode switchport
PE1(config-if-gi)# bridge-group 1
```

Проверим, что интерфейс включен в бридж-домен:

```
PE1# show interfaces bridge
Bridges      Interfaces
-----
bridge 1     gil/0/4
```

```
PE1# sh interfaces status bridge 1
Interface 'bridge 1' status information:
Description:      --
Operational state: Up
Administrative state: Up
Supports broadcast: Yes
Supports multicast: Yes
MTU:              1500
MAC address:      a8:f9:4b:ac:4d:15
Last change:      4 minutes and 22 seconds
Mode:             Routerport
```

PE2:

```
PE2(config)# bridge 1
PE2(config-bridge)# enable
PE2(config-bridge)# exit
PE2(config)# interface gigabitethernet 1/0/4
```

```

PE2(config-if-gi)# mode switchport
PE2(config-if-gi)# bridge-group 1
PE2# show interfaces bridge 1
Bridges      Interfaces
-----
bridge 1     gil/0/4

PE2# sh interfaces status bridge 1
Interface 'bridge 1' status information:
Description:      --
Operational state: Up
Administrative state: Up
Supports broadcast: Yes
Supports multicast: Yes
MTU:              1500
MAC address:      a8:f9:4b:ad:f2:45
Last change:      10 seconds
Mode:             routerport

```

PE3:

```

PE3(config)# bridge 1
PE3(config-bridge)# enable
PE3(config-bridge)# exit
PE3(config)# interface gigabitethernet 1/0/4
PE3(config-if-gi)# mode switchport
PE3(config-if-gi)# bridge-group 1
PE3# show interfaces bridge
Bridges      Interfaces
-----
bridge 1     gil/0/4
PE3# sh interfaces status bridge
Interface      Admin  Link  MTU  MAC address  Last change
Mode
              state  state
-----
bridge 1      Up    Up    1500  a8:f9:4b:ac:df:f0  1 minute and
21 seconds    Routerport

PE3# sh interfaces status bridge 1
Interface 'bridge 1' status information:
Description:      --
Operational state: Up
Administrative state: Up
Supports broadcast: Yes
Supports multicast: Yes
MTU:              1500
MAC address:      a8:f9:4b:ac:df:f0
Last change:      1 minute and 24 seconds
Mode:             Routerport

```

Далее выполним настройку VPLS:

PE1:

Переходим в контекст настройки L2VPN и включим в него заранее созданный бридж-домен.

```
PE1(config)# mpls
PE1(config-mpls)# l2vpn
PE1(config-l2vpn)# vpls l2vpn
PE1(config-l2vpn-vpls)# bridge-group 1
```

Укажем RD, RT, VE-ID, VPN ID согласно схеме сети, приведенной выше, и активируем сервис:

В некоторых случаях можно отказаться от ввода таких параметров, как RD и RT: если указать только VPN ID, то они будут сформированы следующим образом: <номер AS> : <vpn-id>.

Например, есть номер автономной системы AS 65550, vpn-id указан 10, тогда сгенерируются следующие параметры:

RD - 65550:10.

RT import/export - 65550:10.

```
PE1(config-l2vpn-vpls)# autodiscovery bgp
PE1(config-bgp)# rd 65500:100
PE1(config-bgp)# route-target import 65500:100
PE1(config-bgp)# route-target export 65500:100
PE1(config-bgp)# ve id 1
PE1(config-bgp)# vpn id 1
PE1(config-bgp)# exit
PE1(config-l2vpn-vpls)# enable
```

После активации сервиса проверим, что в таблице l2vpn появилась маршрутная информация и она анонсируется на RR:

```
PE1# show bgp l2vpn vpls all
Status codes: * - valid, > - best, i - internal, S - stale
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Codes	Route	Distinguisher	VID	VBO	VBS	Next hop	Metric	LocPrf	Weight	Path
*>	65500:100		1	1	10	--	--	--	--	

```
PE1# show bgp l2vpn vpls all neighbor 10.10.0.4 advertise-routes
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Route	Distinguisher	VID	VBO	VBS	Next hop	Metric	LocPrf	Path
65500:100		1	1	10	10.10.0.1	--	100	i

* Подробный вывод анонсированного маршрута *

```
PE1# show bgp l2vpn vpls all neighbor 10.10.0.4 advertise-routes ve-id 1 block
-offset 1
BGP routing table entry for 65500:100 VE ID 1 VE Block Offset 1
  VE Block Size:      10
  Label Base:         86
  Next hop:           10.10.0.1
  AS path:            --
```



```
Origin: IGP
Local preference: 100
Extended Community: RT:65500:100
Layer2-info: encaps (VPLS), control flags(0x00), MTU (1500)
```

Переходим к настройке PE2:

```
PE2(config-mpls)# l2vpn
PE2(config-l2vpn)# vpls l2vpn
PE2(config-l2vpn-vpls)# bridge-group 1
PE2(config-l2vpn-vpls)# autodiscovery bgp
PE2(config-bgp)# rd 65500:100
PE2(config-bgp)# route-target export 65500:100
PE2(config-bgp)# route-target import 65500:100
PE2(config-bgp)# vpn id 2
PE2(config-bgp)# ve id 2
PE2(config-bgp)# exit
PE2(config-l2vpn-vpls)# enable
```

Проверим, что PE2 анонсирует маршрутную информацию на RR:

```
PE2# show bgp l2vpn vpls all neighbor 10.10.0.4 advertise-routes
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Route	Distinguisher	VID	VBO	VBS	Next hop	Metric	LocPrf	Path
65500:100		2	1	10	10.10.0.2	--	100	i

В таблице l2vpn видны как свои маршруты, так и от PE1:

```
PE2# show bgp l2vpn vpls all
Status codes: * - valid, > - best, i - internal, S - stale
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Codes	Route	Distinguisher	VID	VBO	VBS	Next hop	Metric	LocPrf	Weight	Path
*>	65500:100		2	1	10	--	--	--	--	
*>i	65500:100		1	1	10	10.10.0.1	--	100	0	i

Просмотреть вычисленные сервисные метки можно следующим образом:

1)

```
PE2# show mpls l2vpn bindings
Neighbor: 10.10.0.1, PW ID: 2, VE ID: 1
Local label: 45
Encapsulation Type: VPLS
Control flags: 0x00
MTU: 1500
Remote label: 87
Encapsulation Type: VPLS
Control flags: 0x00
MTU: 1500
```

2)

```
PE2# show mpls forwarding-table
```

Local label	Outgoing label	Prefix or tunnel ID	Outgoing Interface	Next Hop
-----	-----	-----	-----	-----
45	87	PW ID 2	--	10.10.0.1

Проверим состояние сервиса:

```
PE2# show mpls l2vpn vpls l2vpn
```

```
VPLS: l2vpn
```

```
bridge 1:
```

```
MTU: 1500
```

```
Status: Up
```

```
ACs:
```

```
gigabitethernet 1/0/4:
```

```
MTU: 1500
```

```
Status: Up
```

```
PWs:
```

```
PW ID 2, Neighbor 10.10.0.1:
```

```
MTU: 1500
```

```
Last change: 00:21:33
```

```
Status: Up
```

Перейдем к настройке PE3:

```
PE3# config
```

```
PE3(config)# mpls
```

```
PE3(config-mpls)# l2vpn
```

```
PE3(config-l2vpn)# vpls l2vpn
```

```
PE3(config-l2vpn-vpls)# bridge-group 1
```

```
PE3(config-l2vpn-vpls)# autodiscovery bgp
```

```
PE3(config-bgp)# rd 65500:100
```

```
PE3(config-bgp)# route-target export 65500:100
```

```
PE3(config-bgp)# route-target import 65500:100
```

```
PE3(config-bgp)# ve id 3
```

```
PE3(config-bgp)# vpn id 3
```

```
PE3(config-bgp)# exit
```

```
PE3(config-l2vpn-vpls)# enable
```

Проверим маршрутную информацию на PE3:

```
PE3# show bgp l2vpn vpls all
```

```
Status codes: * - valid, > - best, i - internal, S - stale
```

```
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Codes	Route	Distinguisher	VID	VBO	VBS	Next hop	Metric	LocPrf	Weight	Path
----	-----	-----	----	----	----	-----	-----	-----	-----	-----
*>	65500:100		3	1	10	--	--	--	--	
*>i	65500:100		2	1	10	10.10.0.2	--	100	0	i
*>i	65500:100		1	1	10	10.10.0.1	--	100	0	i

Убедимся, что PE3 анонсирует маршрутную информацию на RR:

```
PE3# show bgp l2vpn vpls all neighbor 10.10.0.4 advertise-routes
```

```
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Route	Distinguisher	VID	VBO	VBS	Next hop	Metric	LocPrf	Path
-------	---------------	-----	-----	-----	----------	--------	--------	------

```
-----
65500:100          3      1      10      10.10.0.3      --          100          i
-----
```

Проверим, что псевдо-провод построен до обоих PE и находится в статусе 'UP':

```
PE3# show mpls l2vpn vpls l2vpn
VPLS: l2vpn
  bridge 1:
    MTU:      1500
    Status: Up
  ACs:
    gigabitethernet 1/0/4:
      MTU:      1500
      Status: Up
  PWs:
    PW ID 3, Neighbor 10.10.0.2:
      MTU:      1500
      Last change: 00:06:08
      Status:    Up
    PW ID 3, Neighbor 10.10.0.1:
      MTU:      1500
      Last change: 00:06:08
      Status:    Up
```

Проверим сетевую доступность клиентских устройств (CE):

```
CE3# ping 192.168.0.1
PING 192.168.0.1 (192.168.0.1) 56(84) bytes of data.
!!!!
--- 192.168.0.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 0.173/0.208/0.290/0.045 ms
CE3# ping 192.168.0.2
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.
!!!!
--- 192.168.0.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 0.158/0.204/0.255/0.032 ms
```

```
PE3# sh mac address-table bridge 1
VID      MAC Address              Interface                               Type
-----
--      a8:f9:4b:aa:11:08             gigabitethernet 1/0/4                 Dynamic
--      a8:f9:4b:aa:11:06             dypseudowire 3_10.10.0.1              Dynamic
--      a8:f9:4b:aa:11:07             dypseudowire 3_10.10.0.2              Dynamic
3 valid mac entries
```

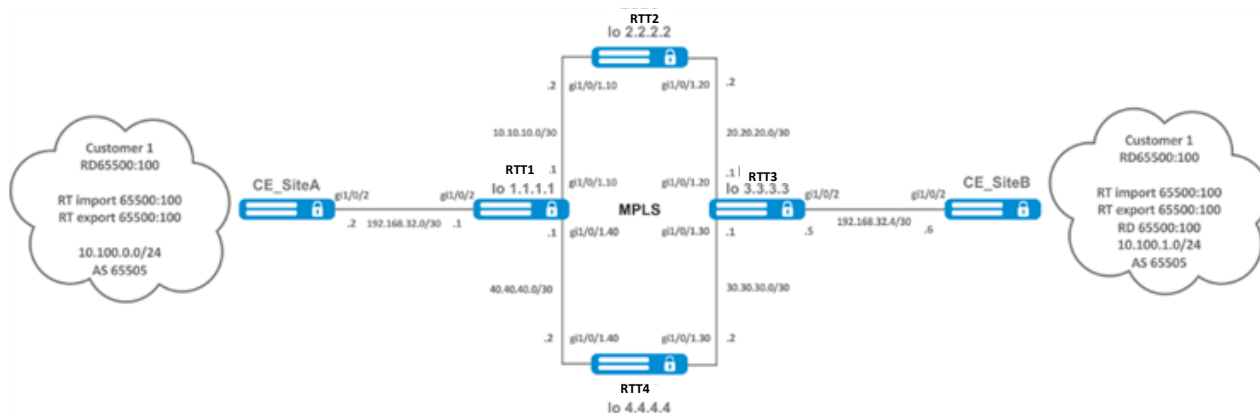
Настройка L2VPN-сервиса завершена.

13.7. Настройка сервиса L3VPN

Сервис L3VPN позволяет объединить распределенные клиентские IP-сети и обеспечить передачу трафика между ними в рамках единой VRF.



В текущей реализации протокола MP-BGP поддерживается передача только VPN-IPv4 маршрутов (AFI = 1, SAFI = 128).



13.7.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Настроить адресацию и один из протоколов IGP на всех PE и PE-маршрутизаторах.		
2	Настроить распространение транспортных меток по протоколу LDP.		
3	Создать VRF.	<code>rtt(config)# ip vrf <VRF></code>	<VRF> – имя экземпляра VRF, задается строкой до 31 символа.
4	Указать route distinguisher для данного VRF.	<code>rtt(config-vrf)# rd <RD></code>	<p><RD> – значение Route distinguisher, задается в одном из следующих видов:</p> <ul style="list-style-type: none"> <ASN>:<nn> – где <ASN> – принимает значение [1..65535], nn – принимает значение [1..65535]; <ADDR>:<nn> – где <ADDR> имеет вид – AAA.BBB.CCC.DDD/EE и AAA-DDD принимают значения [0..255], а nn – принимает значение [1..65535]; <4ASN>:<nn> – где <4ASN> – принимает значение [1..4294967295], nn – принимает значение [1..65535].

Шаг	Описание	Команда	Ключи
5	Указать route target import для данного VRF.	<code>rtt(config-vrf) # route-target import <RT></code>	<p><RT> – значение route-target, задается в одном из следующих видов:</p> <ul style="list-style-type: none"> • <ASN>:<nn> – где <ASN> – принимает значение [1..65535], nn – принимает значение [1..65535]; • <ADDR>:<nn> – где <ADDR> имеет вид – AAA.BBB.CCC.DDD/EE и AAA-DDD принимают значения [0..255], а nn – принимает значение [1..65535]; • <4BASN>:<nn> – где <4ASN> – принимает значение [1..4294967295], nn – принимает значение [1..65535].
6	Указать route target export для данного VRF.	<code>rtt(config-vrf) # route-target export <RT></code>	<p><RT> – значение route-target, задается в одном из следующих видов:</p> <ul style="list-style-type: none"> • <ASN>:<nn> – где <ASN> – принимает значение [1..65535], nn – принимает значение [1..65535]; • <ADDR>:<nn> – где <ADDR> имеет вид – AAA.BBB.CCC.DDD/EE и AAA-DDD принимают значения [0..255], а nn – принимает значение [1..65535]; • <4BASN>:<nn> – где <4ASN> – принимает значение [1..4294967295], nn – принимает значение [1..65535].
7	Указать разрешенное количество маршрутов для данного VRF.	<code>rtt(config-vrf) # ip protocols <PROTOCOLS> max- routes <VALUE></code>	<p><PROTOCOL> – вид протокола, принимает значения: rip (только в глобальном режиме), ospf, isis, bgp;</p> <p><VALUE> – количество маршрутов в маршрутной таблице, принимает значения в диапазоне:</p> <ul style="list-style-type: none"> • BGP: R800 – [1..5000000]; R100/200 – [1..2500000]. • OSPF и IS-IS: R800 – [1..500000]; R100/200 – [1..300000].
8	В рамках настройки address-family VPNv4 протокола BGP включить передачу расширенных атрибутов.	<code>rtt(config-bgp- neighbor-af) # send- community extended</code>	

13.7.2. Пример настройки

Задача:

Настроить L3VPN на базе технологии MPLS между RTT1 и RTT3. Конечным результатом настройки является появление связности между узлами, подключенными к VRF на различных маршрутизаторах сети (то есть объединение VRF на разных маршрутизаторах через MPLS-транспорт). При этом должна быть обеспечена передача сервисных MPLS-меток для сервиса L3VPN посредством MP-BGP и передача транспортных меток для достижения nexthop-адресов полученных BGP-маршрутов.

Решение:

13.7.2.1. *Настройка адресации и включение IGP на P/PE-маршрутизаторах*

RTT1

```
RTT1(config)# router ospf log-adjacency-changes
RTT1(config)# router ospf 1
RTT1(config-ospf)# router-id 1.1.1.1
RTT1(config-ospf)# area 0.0.0.0
RTT1(config-ospf-area)# enable
RTT1(config-ospf-area)# exit
RTT1(config-ospf)# enable
RTT1(config-ospf)# exit
RTT1(config)#
RTT1(config)# interface loopback 1
RTT1(config-loopback)# ip address 1.1.1.1/32
RTT1(config-loopback)# ip ospf instance 1
RTT1(config-loopback)# ip ospf
RTT1(config-loopback)# exit
RTT1(config)#
RTT1(config)# interface gigabitethernet 1/0/1.10
RTT1(config-if-sub)# ip firewall disable
RTT1(config-if-sub)# ip address 10.10.10.1/30
RTT1(config-if-sub)# ip ospf instance 1
RTT1(config-if-sub)# ip ospf
RTT1(config-if-sub)# exit
RTT1(config)#
RTT1(config)# interface gigabitethernet 1/0/1.40
RTT1(config-if-sub)# ip firewall disable
RTT1(config-if-sub)# ip address 40.40.40.1/30
RTT1(config-if-sub)# ip ospf instance 1
RTT1(config-if-sub)# ip ospf
RTT1(config-if-sub)# exit
RTT1(config)#
RTT1(config)# system jumbo-frames
RTT1(config)# do commit
RTT1(config)# do confirm
```

RTT2

```
RTT2(config)# router ospf log-adjacency-changes
RTT2(config)# router ospf 1
RTT2(config-ospf)# router-id 2.2.2.2
```

```
RTT2(config-ospf)# area 0.0.0.0
RTT2(config-ospf-area)# enable
RTT2(config-ospf-area)# exit
RTT2(config-ospf)# enable
RTT2(config-ospf)# exit
RTT2(config)#
RTT2(config)# interface loopback 1
RTT2(config-loopback)# ip address 2.2.2.2/32
RTT2(config-loopback)# ip ospf instance 1
RTT2(config-loopback)# ip ospf
RTT2(config-loopback)# exit
RTT2(config)#
RTT2(config)# interface gigabitethernet 1/0/1.10
RTT2(config-if-sub)# ip firewall disable
RTT2(config-if-sub)# ip address 10.10.10.2/30
RTT2(config-if-sub)# ip ospf instance 1
RTT2(config-if-sub)# ip ospf
RTT2(config-if-sub)# exit
RTT2(config)#
RTT2(config)# interface gigabitethernet 1/0/1.20
RTT2(config-if-sub)# ip firewall disable
RTT2(config-if-sub)# ip address 20.20.20.2/30
RTT2(config-if-sub)# ip ospf instance 1
RTT2(config-if-sub)# ip ospf
RTT2(config-if-sub)# exit
RTT2(config)#
RTT2(config)# system jumbo-frames
RTT2(config)# do commit
RTT2(config)# do confirm
```

RTT3

```
RTT3(config)# router ospf log-adjacency-changes
RTT3(config)# router ospf 1
RTT3(config-ospf)# router-id 3.3.3.3
RTT3(config-ospf)# area 0.0.0.0
RTT3(config-ospf-area)# enable
RTT3(config-ospf-area)# exit
RTT3(config-ospf)# enable
RTT3(config-ospf)# exit
RTT3(config)#
RTT3(config)# interface loopback 1
RTT3(config-loopback)# ip address 3.3.3.3/32
RTT3(config-loopback)# ip ospf instance 1
RTT3(config-loopback)# ip ospf
RTT3(config-loopback)# exit
RTT3(config)#
RTT3(config)# interface gigabitethernet 1/0/1.20
RTT3(config-if-sub)# ip firewall disable
RTT3(config-if-sub)# ip address 20.20.20.1/30
RTT3(config-if-sub)# ip ospf instance 1
RTT3(config-if-sub)# ip ospf
RTT3(config-if-sub)# exit
RTT3(config)#
RTT3(config)# interface gigabitethernet 1/0/1.30
RTT3(config-if-sub)# ip firewall disable
RTT3(config-if-sub)# ip address 30.30.30.1/30
```

```
RTT3(config-if-sub)# ip ospf instance 1
RTT3(config-if-sub)# ip ospf
RTT3(config-if-sub)# exit
RTT3(config)#
RTT3(config)# system jumbo-frames
RTT3(config)# do commit
RTT3(config)# do confirm
```

RTT4

```
RTT4(config)# router ospf log-adjacency-changes
RTT4(config)# router ospf 1
RTT4(config-ospf)# router-id 4.4.4.4
RTT4(config-ospf)# area 0.0.0.0
RTT4(config-ospf-area)# enable
RTT4(config-ospf-area)# exit
RTT4(config-ospf)# enable
RTT4(config-ospf)# exit
RTT4(config)#
RTT4(config)# interface loopback 1
RTT4(config-loopback)# ip address 4.4.4.4/32
RTT4(config-loopback)# ip ospf instance 1
RTT4(config-loopback)# ip ospf
RTT4(config-loopback)# exit
RTT4(config)#
RTT4(config)# interface gigabitethernet 1/0/1.40
RTT4(config-if-sub)# ip firewall disable
RTT4(config-if-sub)# ip address 40.40.40.2/30
RTT4(config-if-sub)# ip ospf instance 1
RTT4(config-if-sub)# ip ospf
RTT4(config-if-sub)# exit
RTT4(config)#
RTT4(config)# interface gigabitethernet 1/0/1.30
RTT4(config-if-sub)# ip firewall disable
RTT4(config-if-sub)# ip address 30.30.30.2/30
RTT4(config-if-sub)# ip ospf instance 1
RTT4(config-if-sub)# ip ospf
RTT4(config-if-sub)# exit
RTT4(config)#
RTT4(config)# system jumbo-frames
RTT4(config)# do commit
RTT4(config)# do confirm
```

Необходимо убедиться, что протокол OSPF запущен на каждом маршрутизаторе:

```
RTT1# show ip ospf neighbors
```

Router ID	Pri	State	DTime	Interface	Router IP
2.2.2.2	128	Full/BDR	00:39	gil/0/1.10	10.10.10.2
4.4.4.4	128	Full/BDR	00:32	gil/0/1.40	40.40.40.2

```
RTT1# show ip ospf
```

O	40.40.40.0/30	[150/10]	dev	gil/0/1.40	[ospf1 1970-01-08]	(1.1.1.1)
O	* 30.30.30.0/30	[150/20]	via	40.40.40.2 on gil/0/1.40	[ospf1 1970-01-08]	(3.3.3.3)
O	1.1.1.1/32	[150/0]	dev	lo1	[ospf1 1970-01-08]	(1.1.1.1)
O	* 4.4.4.4/32	[150/10]	via	40.40.40.2 on gil/0/1.40	[ospf1 1970-01-08]	(4.4.4.4)
O	* 20.20.20.0/30	[150/20]	via	10.10.10.2 on gil/0/1.10	[ospf1 22:05:45]	(3.3.3.3)
O	10.10.10.0/30	[150/10]	dev	gil/0/1.10	[ospf1 22:05:33]	(1.1.1.1)


```
O * 3.3.3.3/32      [150/20] multipath                                [ospf1 22:05:45] (3.3.3.3)
                                via 40.40.40.2 on gil/0/1.40 weight 1
O * 2.2.2.2/32      [150/10] via 10.10.10.2 on gil/0/1.10 [ospf1 22:05:45] (2.2.2.2)
```

13.7.2.2. Настройка LDP на R/PE-маршрутизаторах

RTT1

```
RTT1# config
RTT1(config)# mpls
RTT1(config-mpls)# ldp
RTT1(config-ldp)# address-family ipv4
RTT1(config-ldp-af-ipv4)# transport-address 1.1.1.1
RTT1(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1.10
RTT1(config-ldp-af-ipv4-if)# exit
RTT1(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1.40
RTT1(config-ldp-af-ipv4-if)# exit
RTT1(config-ldp-af-ipv4)# exit
RTT1(config-ldp)# enable
RTT1(config-ldp)# exit
RTT1(config-mpls)# forwarding interface gigabitethernet 1/0/1.10
RTT1(config-mpls)# forwarding interface gigabitethernet 1/0/1.40
RTT1(config-mpls)# exit
RTT1(config)# do commit
RTT1(config)# do confirm
```

RTT2

```
RTT2# config
RTT2(config)# mpls
RTT2(config-mpls)# ldp
RTT2(config-ldp)# address-family ipv4
RTT2(config-ldp-af-ipv4)# transport-address 2.2.2.2
RTT2(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1.10
RTT2(config-ldp-af-ipv4-if)# exit
RTT2(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1.20
RTT2(config-ldp-af-ipv4-if)# exit
RTT2(config-ldp-af-ipv4)# exit
RTT2(config-ldp)# enable
RTT2(config-ldp)# exit
RTT2(config-mpls)# forwarding interface gigabitethernet 1/0/1.10
RTT2(config-mpls)# forwarding interface gigabitethernet 1/0/1.20
RTT2(config-mpls)# exit
RTT2(config)# do commit
RTT2(config)# do confirm
```

RTT3

```
RTT3# config
RTT3(config)# mpls
RTT3(config-mpls)# ldp
RTT3(config-ldp)# address-family ipv4
RTT3(config-ldp-af-ipv4)# transport-address 3.3.3.3
RTT3(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1.20
RTT3(config-ldp-af-ipv4-if)# exit
```

```
RTT3(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1.30
RTT3(config-ldp-af-ipv4-if)# exit
RTT3(config-ldp-af-ipv4)# exit
RTT3(config-ldp)# enable
RTT3(config-ldp)# exit
RTT3(config-mpls)# forwarding interface gigabitethernet 1/0/1.20
RTT3(config-mpls)# forwarding interface gigabitethernet 1/0/1.30
RTT3(config-mpls)# exit
RTT3(config)# do commit
RTT3(config)# do confirm
```

RTT4

```
RTT4# config
RTT4(config)# mpls
RTT4(config-mpls)# ldp
RTT4(config-ldp)# address-family ipv4
RTT4(config-ldp-af-ipv4)# transport-address 4.4.4.4
RTT4(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1.30
RTT4(config-ldp-af-ipv4-if)# exit
RTT4(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1.40
RTT4(config-ldp-af-ipv4-if)# exit
RTT4(config-ldp-af-ipv4)# exit
RTT4(config-ldp)# enable
RTT4(config-ldp)# exit
RTT4(config-mpls)# forwarding interface gigabitethernet 1/0/1.30
RTT4(config-mpls)# forwarding interface gigabitethernet 1/0/1.40
RTT4(config-mpls)# exit
RTT4(config)# do commit
RTT4(config)# do confirm
```

Для проверки сходимости LDP можно воспользоваться одной из следующих команд:

```
RTT1# show mpls ldp neighbor
Peer LDP ID: 2.2.2.2; Local LDP ID 1.1.1.1
    State: Operational
    TCP connection: 2.2.2.2:33933 - 1.1.1.1:646
    Messages sent/received: 1059/1070
    Uptime: 17:32:07
    LDP discovery sources:
        gigabitethernet 1/0/1.10
Peer LDP ID: 4.4.4.4; Local LDP ID 1.1.1.1
    State: Operational
    TCP connection: 4.4.4.4:40894 - 1.1.1.1:646
    Messages sent/received: 1376/1386
    Uptime: 22:38:38
    LDP discovery sources:
        gigabitethernet 1/0/1.40
```

13.7.2.3. Настройка MP-BGP

Создадим VRF на RTT1 и RTT3 соответственно. Укажем RD, rt-export/import в соответствии со схемой, настроим интерфейс для взаимодействия с СЕ (CE-SiteA и CE-SiteB). Дополнительно создадим route-мар для разрешения анонсирования маршрутов по протоколу BGP:



Без указания атрибутов RD и RT маршрутная информация не попадет в таблицу VPNv4.

RTT1

```
RTT1(config)# ip vrf Customer1
RTT1(config-vrf)# ip protocols bgp max-routes 1000
RTT1(config-vrf)# rd 65500:100
RTT1(config-vrf)# route-target import 65500:100
RTT1(config-vrf)# route-target export 65500:100
RTT1(config-vrf)# exit
RTT1(config)# interface gigabitethernet 1/0/2
RTT1(config-if-gi)# ip vrf forwarding Customer1
RTT1(config-if-gi)# description "Customer1"
RTT1(config-if-gi)# ip firewall disable
RTT1(config-if-gi)# ip address 192.168.32.1/30
RTT1(config-if-gi)# exit
RTT1(config)# route-map OUTPUT
RTT1(config-route-map)# rule 1
RTT1(config-route-map-rule)# action permit
RTT1(config-route-map-rule)# exit
RTT1(config-route-map)# exit
RTT1(config)# do commit
RTT1(config)# do confirm
```

RTT3

```
RTT3(config)# ip vrf Customer1
RTT3(config-vrf)# ip protocols bgp max-routes 1000
RTT3(config-vrf)# rd 65500:100
RTT3(config-vrf)# route-target export 65500:100
RTT3(config-vrf)# route-target import 65500:100
RTT3(config-vrf)# exit
RTT3(config)# interface gigabitethernet 1/0/2
RTT3(config-if-gi)# ip vrf forwarding Customer1
RTT3(config-if-gi)# description "Customer1"
RTT3(config-if-gi)# ip firewall disable
RTT3(config-if-gi)# ip address 192.168.32.5/30
RTT3(config-if-gi)# exit
RTT3(config)# route-map OUTPUT
RTT3(config-route-map)# rule 1
RTT3(config-route-map-rule)# action permit
RTT3(config-route-map-rule)# exit
RTT3(config-route-map)# exit
RTT3(config)# do commit
RTT3(config)# do confirm
```

Настроим iBGP между RTT1 и RTT3. Включим отправку extended community на обоих устройствах:

RTT1

```
RTT1(config)# router bgp log-neighbor-changes
RTT1(config)# router bgp 65500
RTT1(config-bgp)# router-id 1.1.1.1
```

```
RTT1(config-bgp)# enable
RTT1(config-bgp)# neighbor 3.3.3.3
RTT1(config-bgp-neighbor)# remote-as 65500
RTT1(config-bgp-neighbor)# update-source 1.1.1.1
RTT1(config-bgp-neighbor)# enable
RTT1(config-bgp-neighbor)# address-family vpnv4 unicast
RTT1(config-bgp-neighbor-af)# send-community extended
RTT1(config-bgp-neighbor-af)# enable
RTT1(config-bgp-neighbor-af)# exit
RTT1(config-bgp-neighbor)# exit
RTT1(config-bgp)# exit
RTT1(config)# do commit
RTT1(config)# do confirm
```

RTT3

```
RTT3(config)# router bgp log-neighbor-changes
RTT3(config)# router bgp 65500
RTT3(config-bgp)# router-id 3.3.3.3
RTT3(config-bgp)# enable
RTT3(config-bgp)# neighbor 1.1.1.1
RTT3(config-bgp-neighbor)# remote-as 65500
RTT3(config-bgp-neighbor)# update-source 3.3.3.3
RTT3(config-bgp-neighbor)# enable
RTT3(config-bgp-neighbor)# address-family vpnv4 unicast
RTT3(config-bgp-neighbor-af)# send-community extended
RTT3(config-bgp-neighbor-af)# enable
RTT3(config-bgp-neighbor-af)# exit
RTT3(config-bgp-neighbor)# exit
RTT3(config-bgp)# exit
RTT3(config)# do commit
RTT3(config)# do confirm
```

Необходимо убедиться, что BGP-сессия успешно установлена:

```
RTT1# show bgp neighbors
BGP neighbor is 3.3.3.3
  BGP state: Established
  Neighbor address: 3.3.3.3
  Neighbor AS: 65500
  Neighbor ID: 3.3.3.3
  Neighbor caps: refresh enhanced-refresh restart-aware AS4
  Session: internal multihop AS4
  Source address: 1.1.1.1
  Weight: 0
  Hold timer: 126/180
  Keepalive timer: 40/60
  Address family ipv4 unicast:
  Default originate: No
  Default information originate: No
  Uptime: 88495 s
```

13.7.2.4. Настройка маршрутизации PE-CE

Согласно топологии, Customer1 анонсирует по BGP (AS65505) подсеть 10.100.0.0/24. Необходимо настроить соответствующие интерфейсы, eBGP между RTT1 и CE_SiteA. Также необходимо разрешить анонсирование маршрутов в сторону PE.



По умолчанию для eBGP анонсирование маршрутов запрещено, необходимо настроить разрешающее правило. Для iBGP анонсирование маршрутов разрешено.

Необходимая конфигурация на маршрутизаторе CE-SiteA:

CE_SiteA

```
CE-SiteA(config)# interface gigabitethernet 1/0/2
CE-SiteA(config-if-gi)# ip firewall disable
CE-SiteA(config-if-gi)# ip address 192.168.32.2/30
CE-SiteA(config-if-gi)# exit
CE-SiteA(config)# interface loopback 1
CE-SiteA(config-loopback)# ip address 10.100.0.1/24
CE-SiteA(config-loopback)# exit
CE-SiteA(config)# route-map OUTPUT
CE-SiteA(config-route-map)# rule 1
CE-SiteA(config-route-map-rule)# match ip address 10.100.0.0/24
CE-SiteA(config-route-map-rule)# action permit
CE-SiteA(config-route-map-rule)# exit
CE-SiteA(config-route-map)# exit
CE-SiteA(config)# router bgp log-neighbor-changes
CE-SiteA(config)# router bgp 65505
CE-SiteA(config-bgp)# router-id 192.168.32.1
CE-SiteA(config-bgp)# neighbor 192.168.32.1
CE-SiteA(config-bgp-neighbor)# remote-as 65500
CE-SiteA(config-bgp-neighbor)# allow-local-as 1
CE-SiteA(config-bgp-neighbor)# update-source 192.168.32.2
CE-SiteA(config-bgp-neighbor)# address-family ipv4 unicast
CE-SiteA(config-bgp-neighbor-af)# route-map OUTPUT out
CE-SiteA(config-bgp-neighbor-af)# enable
CE-SiteA(config-bgp-neighbor-af)# exit
CE-SiteA(config-bgp-neighbor)# enable
CE-SiteA(config-bgp-neighbor)# exit
CE-SiteA(config-bgp)# address-family ipv4 unicast
CE-SiteA(config-bgp-af)# network 10.100.0.0/24
CE-SiteA(config-bgp-af)# exit
CE-SiteA(config-bgp)# enable
CE-SiteA(config-bgp)# exit
CE-SiteA(config)# do commit
CE-SiteA(config)# do confirm
```

Переходим к настройке eBGP на маршрутизаторе RTT1.

Создадим eBGP-сессию с CE_SiteA и разрешим передачу маршрутов BGP-пиру:

RTT1

```
RTT1(config)# router bgp 65500
RTT1(config-bgp)# vrf Customer1
RTT1(config-bgp-vrf)# router-id 192.168.32.1
RTT1(config-bgp-vrf)# neighbor 192.168.32.2
RTT1(config-bgp-vrf-neighbor)# remote-as 65505
RTT1(config-bgp-vrf-neighbor)# update-source 192.168.32.1
RTT1(config-bgp-vrf-neighbor)# address-family ipv4 unicast
RTT1(config-bgp-neighbor-af-vrf)# route-map OUTPUT out
RTT1(config-bgp-neighbor-af-vrf)# enable
RTT1(config-bgp-neighbor-af-vrf)# exit
RTT1(config-bgp-vrf-neighbor)# enable
RTT1(config-bgp-vrf-neighbor)# exit
RTT1(config-bgp-vrf)# address-family ipv4 unicast
RTT1(config-bgp-vrf-af)# redistribute connected
RTT1(config-bgp-vrf-af)# redistribute bgp 65500
RTT1(config-bgp-vrf-af)# exit
RTT1(config-bgp-vrf)# enable
RTT1(config-bgp-vrf)# exit
RTT1(config-bgp)# exit
RTT1(config)# do commit
RTT1(config)# do confirm
```

При передаче маршрутов из VRF в таблицу VPNv4 достаточно настроить передачу (redistribute) маршрутов в этом VRF.

Пример конфигурации передачи в VPNv4 таблицу connected- и static-сетей:

```
RTT1(config)# router bgp 65500
RTT1(config-bgp)# router-id 1.1.1.1
RTT1(config-bgp)# neighbor 3.3.3.3
RTT1(config-bgp-neighbor)# remote-as 65500
RTT1(config-bgp-neighbor)# update-source 1.1.1.1
RTT1(config-bgp-neighbor)# enable
RTT1(config-bgp-neighbor)# address-family vpnv4 unicast
RTT1(config-bgp-neighbor-af)# send-community extended
RTT1(config-bgp-neighbor-af)# enable
RTT1(config-bgp-neighbor-af)# exit
RTT1(config-bgp-neighbor)# exit
RTT1(config-bgp)# enable
ERTT1(config-bgp)# vrf Customer1
RTT1(config-bgp-vrf)# address-family ipv4 unicast
RTT1(config-bgp-vrf-af)# redistribute connected
RTT1(config-bgp-vrf-af)# redistribute static
RTT1(config-bgp-vrf-af)# exit
RTT1(config-bgp-vrf)# exit
RTT1(config-bgp)# exit
RTT1(config)# do commit
RTT1(config)# do confirm
```

Для проверки принятых и анонсированных маршрутов можно воспользоваться следующими командами:

```
RTT1# show bgp vpnv4 unicast vrf Customer1 neighbors 192.168.32.2 advertise-routes
Status codes: u - unicast, b - broadcast, m - multicast, a - anycast
               * - valid, > - best
```

Origin codes: i - IGP, e - EGP, ? - incomplete

Network	Next Hop	Metric	LocPrf	Weight	Path
*> u 10.100.1.0/24	192.168.32.1		100		65500 i
*> u 192.168.32.4/30	192.168.32.1		100		65500 i

Вывод анонсируемых маршрутов для определенного пира. Маршрутная информация отображается после применения фильтрации:

```
RTT1# show bgp vpnv4 unicast vrf Customer1 neighbors 192.168.32.2 routes
Status codes: u - unicast, b - broadcast, m - multicast, a - anycast
               * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Network	Next Hop	Metric	LocPrf	Weight	Path
*> u 10.100.0.0/24	192.168.32.2		100	0	65505

Вывод принятой маршрутной информации от определенного пира. Маршрутная информация отображается после применения фильтрации.

CE-SiteB

Необходимо проделать схожие операции между маршрутизаторами RTT3 и CE_SiteB.

Произвести настройку соответствующих интерфейсов и создать eBGP-сессию между RTT3 и CE_SiteB:

CE-SiteB

```
CE-SiteB(config)# interface gigabitethernet 1/0/2
CE-SiteB(config-if-gi)# ip firewall disable
CE-SiteB(config-if-gi)# ip address 192.168.32.6/30
CE-SiteB(config-if-gi)# exit
CE-SiteB(config)#
CE-SiteB(config)# interface loopback 1
CE-SiteB(config-loopback)# ip address 10.100.1.1/24
CE-SiteB(config-loopback)# exit
CE-SiteB(config)#
CE-SiteB(config)# route-map OUTPUT
CE-SiteB(config-route-map)# rule 1
CE-SiteB(config-route-map-rule)# match ip address 10.100.1.0/24
CE-SiteB(config-route-map-rule)# action permit
CE-SiteB(config-route-map-rule)# exit
CE-SiteB(config-route-map)# exit
CE-SiteB(config)#
CE-SiteB(config)# router bgp 65505
CE-SiteB(config-bgp)# router-id 192.168.32.6
CE-SiteB(config-bgp)# neighbor 192.168.32.5
CE-SiteB(config-bgp-neighbor)# remote-as 65500
CE-SiteB(config-bgp-neighbor)# allow-local-as 1
CE-SiteB(config-bgp-neighbor)# update-source 192.168.32.6
CE-SiteB(config-bgp-neighbor)# address-family ipv4 unicast
CE-SiteB(config-bgp-neighbor-af)# route-map OUTPUT out
CE-SiteB(config-bgp-neighbor-af)# enable
```

```
CE-SiteB(config-bgp-neighbor-af)# exit
CE-SiteB(config-bgp-neighbor)# enable
CE-SiteB(config-bgp-neighbor)# exit
CE-SiteB(config-bgp)# address-family ipv4 unicast
CE-SiteB(config-bgp-af)# network 10.100.1.0/24
CE-SiteB(config-bgp-af)# exit
CE-SiteB(config-bgp)# enable
CE-SiteB(config-bgp)# exit
CE-SiteB(config)# do commit
CE-SiteB(config)# do confirm
```

Со стороны RTT3 также настроить eBGP и разрешить передачу маршрутной информации из VRF в таблицу VPNv4:

RTT3

```
router bgp 65500
RTT3(config)# router bgp 65500
RTT3(config-bgp)# vrf Customer1
RTT3(config-bgp-vrf)# router-id 192.168.32.5
RTT3(config-bgp-vrf)# neighbor 192.168.32.6
RTT3(config-bgp-vrf-neighbor)# remote-as 65505
RTT3(config-bgp-vrf-neighbor)# update-source 192.168.32.5
RTT3(config-bgp-vrf-neighbor)# address-family ipv4 unicast
RTT3(config-bgp-neighbor-af-vrf)# route-map OUTPUT out
RTT3(config-bgp-neighbor-af-vrf)# enable
RTT3(config-bgp-neighbor-af-vrf)# exit
RTT3(config-bgp-vrf-neighbor)# enable
RTT3(config-bgp-vrf-neighbor)# exit
RTT3(config-bgp-vrf)# address-family ipv4 unicast
RTT3(config-bgp-vrf-af)# redistribute connected
RTT3(config-bgp-vrf-af)# redistribute bgp 65500
RTT3(config-bgp-vrf-af)# exit
RTT3(config-bgp-vrf)# enable
RTT3(config-bgp-vrf)# exit
RTT3(config-bgp)# exit
RTT3(config)# do commit
RTT3(config)# do confirm
```

Для просмотра VPNv4-таблицы воспользоваться командой:

```
RTT1# show bgp vpnv4 unicast all
Status codes: * - valid, > - best, i - internal, S - stale
Origin codes: i - IGP, e - EGP, ? - incomplete
```

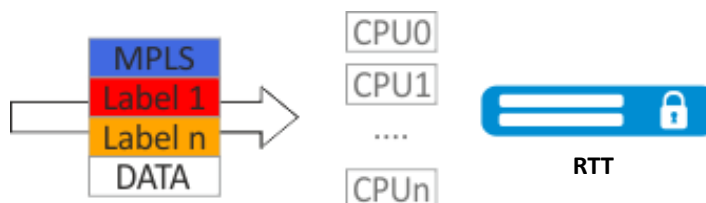
Codes	Route	Distinguisher	IP Prefix	Next hop	Metric	Label	LocPrf	Weight	Path
*>	65500:100		10.100.0.0/24	--	--	23	--	--	?
*>i	65500:100		192.168.32.4/30	3.3.3.3	--	84	100	0	i
*>i	65500:100		10.100.1.0/24	3.3.3.3	--	84	100	0	i

Данная команда выводит все принятые VPNv4-маршруты после применения фильтрации.

13.8. Балансировка трафика MPLS

Маршрутизаторы RTT имеют многоядерную архитектуру. Одним из первых звеньев обработки поступающего трафика является load balancer daemon (lbd), который выполняет две основных функции:

1. Равномерно распределяет нагрузку между всеми CPU маршрутизатора.
2. Выявляет аномальные ситуации с высокой нагрузкой на отдельные CPU и перераспределяет обработку с этих CPU на менее загруженные.



По умолчанию lbd использует только MPLS-метки для вычисления хеша и дальнейшего распределения нагрузки на различные CPU. Данное поведение не всегда дает преимущество, особенно когда существуют «большие» однородные потоки MPLS-трафика. Для добавления энтропии в хеш можно включить дополнительную функцию: **cpu load-balance mpls passenger ip**

Включает возможность «заглядывать» дальше MPLS-заголовка для поиска IP-заголовка и добавления ip-src и ip-dst в расчет хеша:

Для L3VPN: идет поиск пары ip-src и ip-dst в ip-заголовке, находящимся за MPLS-заголовком.



Для L2VPN: RTT попытается «заглянуть» в ethernet-фрейм (который находится за mpls-заголовком) и получить ip-src и ip-dst в ip-заголовке для добавления в расчет хеша.



cpu load-balance mpls passenger ip-over-ethernet-pseudowire-with-cw

cpu load-balance mpls passenger ip-over-ethernet-pseudowire-without-cw

Позволяет явно указать, используется ли при построении L2VPN функция Control Word. Это позволяет исключить возникновение ошибки, когда пакет с наличием Control word может быть ошибочно распознан как пакет без него.

При хешировании MPLS-меток действуют следующие ограничения:

- В расчет не добавляются метки 0-15 (Special-Purpose Labels) – см. RFC 7274;
- В расчет не добавляется метка, если непосредственно перед ней следует метка 15 (Extension Label) – см. RFC 7274;
- В расчет хеша добавляется не более трёх меток.



Во избежание падения LDP-сессии при большой нагрузке на CPU маршрутизатора на моделях R200, R800 после включения функции все пакеты протокола LDP будут обрабатываться управляющими CPU (Management CPU), которые не участвуют в обработке трафика. Для R200, R800 – это CPU 0.

13.8.1. Пример настройки

Задача:

Включить балансировку L2VPN-трафика без использования функционала Control Word.

Решение:

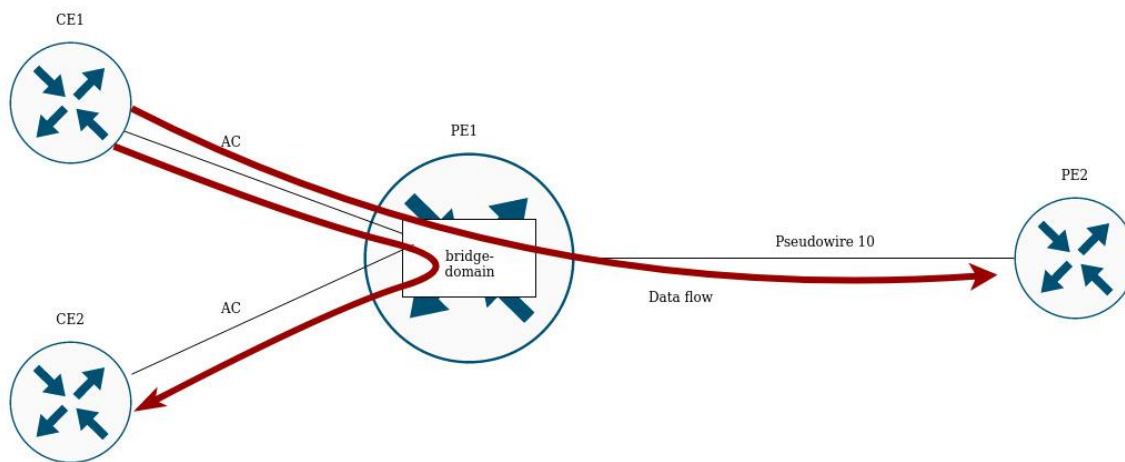
RTT

```
RTT(config)# system cpu load-balance mpls passenger ip
RTT(config)# system cpu load-balance mpls passenger ipoe-pw-without-cw
```

13.9. Работа с бридж-доменом в рамках MPLS

Для организации L2VPN-сервиса необходимо настроить на устройстве бридж-домен, создать требуемые AC, PW (LDP-signaling) и связать все данные элементы с бридж-доменом.

Для point-to-point бридж-домен создается автоматически.



Между элементами бридж-домена осуществляется коммутация трафика на основании перечисленных правил:

1. Для каждого бридж-домена автоматически создается таблица MAC-адресов по аналогии с Ethernet-коммутаторами. Ethernet-кадры коммутируются на основании анализа MAC-адреса получателя (DST MAC).
2. Кадры с известным DST MAC будут отправляться в соответствующие AC/PW.
3. Кадры с неизвестным DST MAC, broadcast- и multicast-кадры (т. н. BUM-трафик, Broadcast, Unknown unicast и Multicast) будут отправляться во все элементы бридж-домена, за исключением того элемента (AC либо PW), с которого вошли в бридж-домен.
4. При коммутации учитываются DST MAC в кадрах, но не учитываются VLAN-теги, имеющиеся на кадрах – таким образом, коммутация внутри бридж-домена не является «VLAN-aware».

Бридж-домен может работать в двух транспортных режимах: ethernet или vlan. Транспортный режим задает правила обработки трафика на входе и выходе с бридж-домена.

В LDP signaling по умолчанию используется ethernet mode (Raw mode, type 5). Для каждого отдельного экземпляра VPLS можно задать транспортный режим.

В BGP signaling бридж-домен работает только в ethernet mode.

```
PE1# config
PE1(config)# mpls
PE1(config-mpls)# l2vpn
PE1(config-l2vpn)# vpls MARTINI_br
PE1(config-l2vpn-vpls)# transport-mode vlan
```

```
PE1# sh mpls l2vpn pseudowire
```

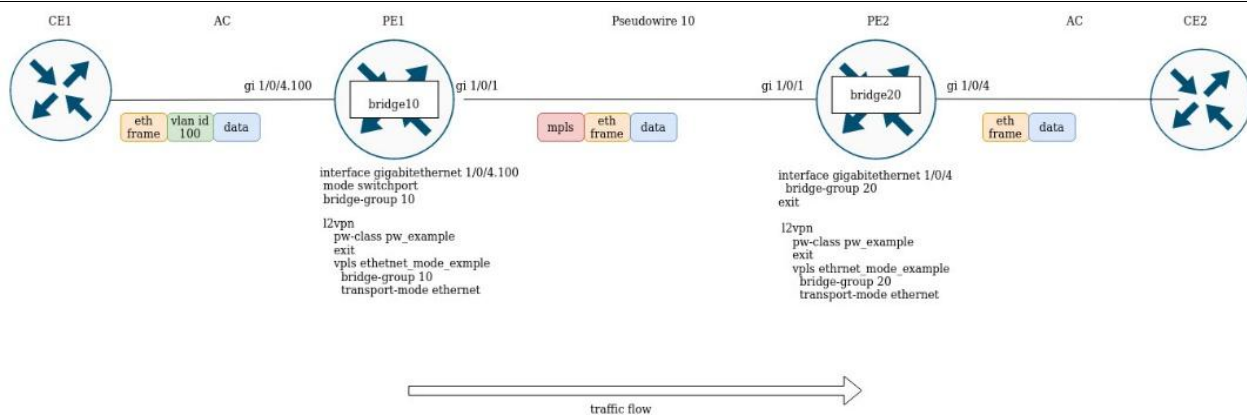
Neighbor	PW ID	Sig Type	Status
10.10.0.2	200	LDP Eth Tagged	Up

В LDP signaling транспортный режим согласуется между PE в процессе создания псевдо-провода, поэтому он должен совпадать на обоих PE.



В LDP signaling транспортный режим согласуется между PE в процессе создания псевдо-провода, поэтому он должен совпадать на обоих PE.

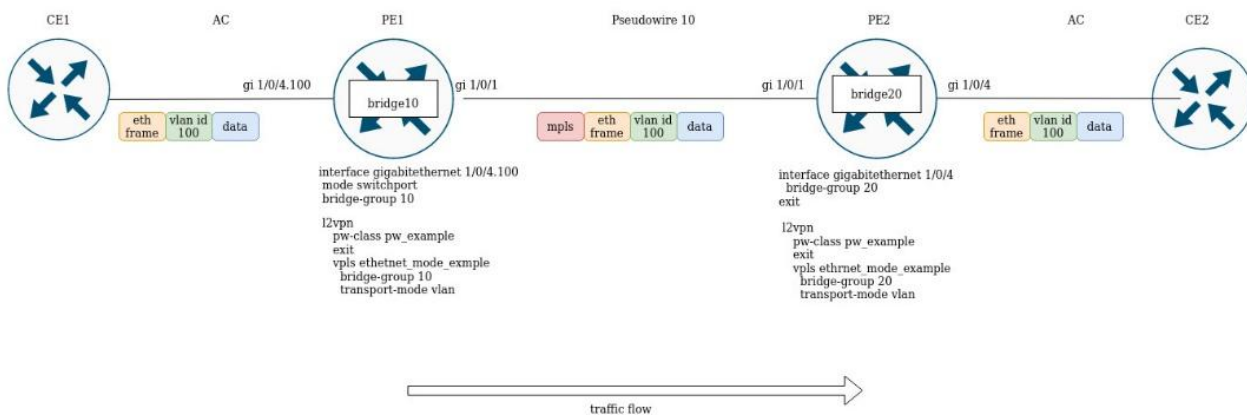
1. Ethernet (Raw) mode:
 - Если AC является саб-интерфейсом, то vlan-тег перед помещением в бридж снимается. При выходе из бриджа vlan-тег восстанавливается.
 - Если AC является интерфейсом, то тегированный и нетегированный трафики проходят в обоих направлениях без модификаций.



Предположим, PE1 и PE2 сконфигурированы в ethernet mode. Со стороны PE1 в бридж-домен включен саб-интерфейс gigabitethernet 1/0/4.100, поэтому vlan-тег (vlan id 100) с входящего трафика будет удален перед помещением в Pseudowire 10 (соответственно, восстановлен при трафике в сторону AC). С другой стороны, AC на PE2 является интерфейсом, значит трафик будет проходить без модификаций в обоих направлениях.

2. Vlan (Tagged) mode:

- Если AC является саб-интерфейсом, то vlan-тег перед помещением в бридж сохраняется. При выходе из бриджа vlan-тег может быть сохранен или перезаписан в зависимости от конфигурации.
- Если AC является интерфейсом, то модификация тегов не происходит в обоих направлениях.



13.10. Назначение MTU при работе с MPLS

Важно правильно настроить MTU на интерфейсах, участвующих в передаче трафика. Существует два ключевых момента:

1. Размер Ethernet-заголовка (18 байт), inner tag (4 байта), outer tag (4 байта) не учитываются на AC-интерфейсах;
2. На интерфейсах, принимающих участие в пересылке MPLS-трафика, необходимо увеличить MTU на количество меток (каждая метка равна 4 байтам).

Значение MTU также участвует в сигнализации при построении псевдо-провода как в LDP-signaling, так и в BGP-signaling. Ниже рассмотрены примеры настройки для обоих случаев:

Для сигнализации (LDP, BGP) значение MTU по умолчанию – 1500.

Значения MTU, участвующие в сигнализации, не влияют на фактический размер пакета, проходящего по псевдо-проводу.

В LDP-signaling MTU задается в рамках настройки pw – class:

LDP-signaling. Настройка MTU для согласования

```
PE2(config)# mpls
PE2(config-mpls)# l2vpn
PE2(config-l2vpn)# pw-class MTU_example
PE2(config-l2vpn-pw-class)# encapsulation mpls mtu 9000
PE2(config-l2vpn-pw-class)# exit
PE2(config-mpls)# l2vpn
PE2(config-l2vpn)# vpls MTU_Example_PW
PE2(config-l2vpn-vpls)# pw 200 10.10.0.1
PE2(config-l2vpn-pw)# pw-class
PE2(config-l2vpn-pw)# pw-class MTU_example
```

Просмотр созданных pw-class'ов

```
PE2# sh mpls l2vpn pw-class
```

PW-class	Neighbor	PW ID	Status	Status-tlv	MTU
MTU_example	10.10.0.1	200	Up	Enable	9000

```
PE2# sh mpls l2vpn vpls MTU_Example_PW
```

```
VPLS: MTU_Example_PW
```

```
...
```

```
PWs:
```

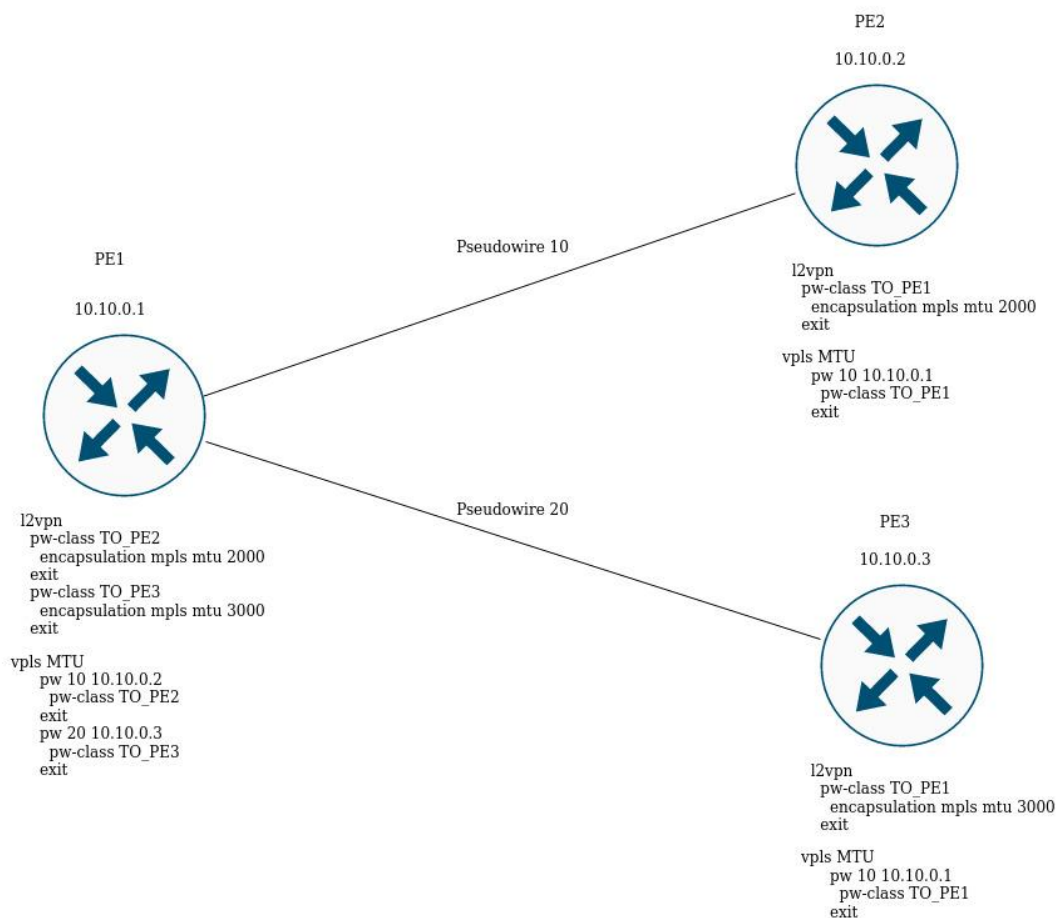
```
  PW ID 2, Neighbor 10.10.0.1:
```

```
    MTU:          9000
```

```
    Last change: 01:27:42
```

```
    Status:        Up
```

* Для сигнализации PW 2 данного VPLS выбрано MTU 9000*



На рисунке выше PE1 поднимает два псевдо-провода: pseudowire 10 до PE2, и pseudowire 20 до PE3 соответственно. Для сигнализации с PE2 MTU будет равным 2000 (pw-class TO_PE2), для PE3 – MTU будет равным 3000 (pw-class TO_PE3).

Для BGP-signaling MTU указывается в рамках конфигурации l2vpn-сервиса:

BGP -signaling. Настройка MTU для согласования

```

PE1(config)# mpls
PE1(config-mpls)# l2vpn
PE1(config-l2vpn)# vpls l2vpn_MTU
PE1(config-l2vpn-vpls)# autodiscovery bgp
PE1(config-bgp)# mtu 1500

PE2# sh mpls l2vpn vpls l2vpn_MTU
VPLS: l2vpn_MTU
...
PWs:
  PW ID 2, Neighbor 10.10.0.1:
    MTU:          1500
    Last change:  01:27:42
    Status:       Up

```

* Для сигнализации всех псевдо-проводов данного VPLS будет выбрано MTU 1500 *

Если при согласовании значение MTU не совпадает, то статус псевдо-провода будет – 'DOWN', 'Reason: MTU mismatch':

```
PE1(config-l2vpn)# vpls l2vpn_MTU
PE1(config-l2vpn-vpls)# autodiscovery bgp
PE1(config-bgp)# mtu 2000
```

```
PE2# sh mpls l2vpn vpls l2vpn_MTU
...
PWs:
  PW ID 2, Neighbor 10.10.0.1:
    MTU:          2000
    Last change:  00:00:10
    Status:       Down
    Reason:       MTU mismatch
```

В BGP-signaling можно отключить проверку MTU для сервиса:

```
PE1(config)# mpls
PE1(config-mpls)# l2vpn
PE1(config-l2vpn)# vpls l2vpn_MTU
PE1(config-l2vpn-vpls)# autodiscovery bgp
PE1(config-bgp)# ignore mtu-mismatch
```

Теперь при согласовании значение MTU будет игнорироваться.

По умолчанию бридж-домен имеет MTU равным 1500 байт. Стоит отметить, что бридж-домен автоматически выбирает наименьшее значение MTU, исходя из собственного MTU и MTU интерфейсов, включенных в бридж-домен.

* Например, имеем бридж-домен 100, в который включены интерфейсы gil/0/1 со значением MTU 2000, и gil/0/2 со значением MTU 3000 *

```
CE3(config)# bridge 100
CE3(config-bridge)# enable
CE3(config-bridge)# exit
CE3(config)# interface gigabitethernet 1/0/1
CE3(config-if-gi)# mtu 2000
CE3(config-if-gi)# bridge-group 100
CE3(config-if-gi)# exit
CE3(config)# interface gigabitethernet 1/0/2
CE3(config-if-gi)# mtu 3000
CE3(config-if-gi)# bridge-group 100
CE3(config-if-gi)# do com
```

* MTU бридж-домена будет равным 1500, так как по умолчанию сам бридж имеет MTU 1500 (значение по умолчанию), которое и стало наименьшим:

```
MTU bridge 100 = 1500  <-- Наименьшее значение MTU
MTU gil/0/1  = 2000
MTU gil/0/2  = 3000
*
```

```
CE3# sh interfaces bridge
Bridges      Interfaces
-----
```

```
bridge 100    gil/0/1-2
```

```
CE3# sh interfaces status bridge 100
Interface 'bridge 100' status information:
  Description:      --
  Operational state: UP
  Administrative state: Up
  Supports broadcast: Yes
  Supports multicast: Yes
  MTU:              1500
  MAC address:      a8:f9:4b:aa:11:00
  Last change:      1 minute and 46 seconds
  Mode:             Routerport
```

* Изменим MTU на самом бридж-домене: *

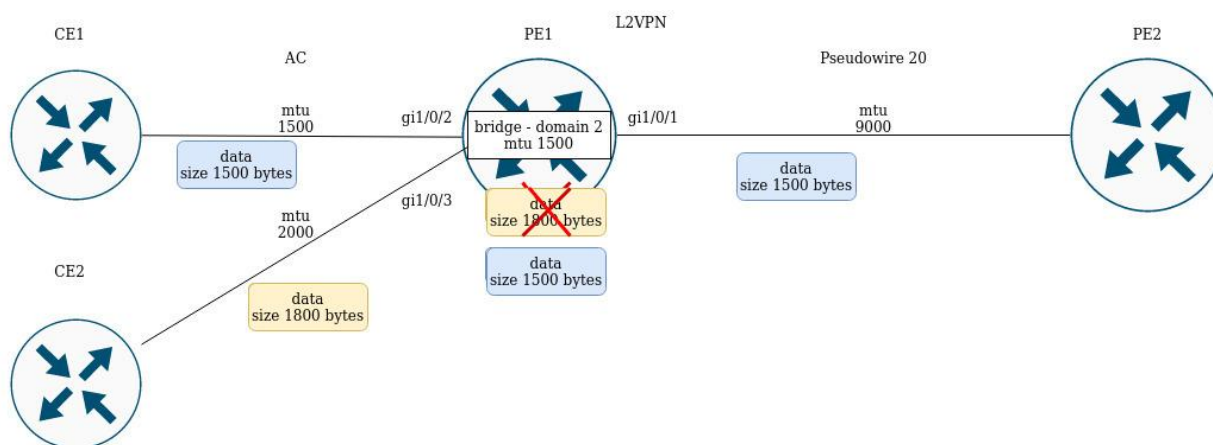
```
CE3(config)# bridge 100
CE3(config-bridge)# mtu 6000
CE3(config-bridge)# do com
```

* MTU бридж-домена стало равным 2000 байт, так как gil/0/2 имеет наименьшее MTU:
MTU bridge 100 = 6000
MTU gil/0/1 = 2000 <-- Наименьшее значение MTU
MTU gil/0/2 = 3000
*

```
CE3# sh interfaces bridge
Bridges      Interfaces
-----
bridge 100    gil/0/1-2
```

```
CE3# sh interfaces status bridge 100
Interface 'bridge 100' status information:
  Description:      --
  Operational state: Up
  Administrative state: Up
  Supports broadcast: Yes
  Supports multicast: Yes
  MTU:              2000
  MAC address:      a8:f9:4b:aa:11:00
  Last change:      6 minutes and 42 seconds
  Mode:             Routerport
```

Рассмотрим пример прохождения трафика в L2VPN-сервисе:



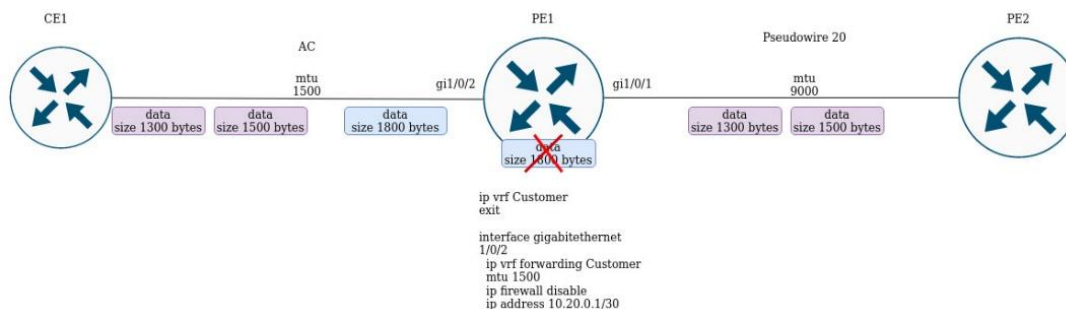
PE1 имеет следующие значения MTU на интерфейсах:

```
PE1# sh interfaces status
```

Interface	Admin state	Link state	MTU	MAC address	Last change	Mode
-----	-----	-----	-----	-----	-----	-----
gi1/0/1	Up	Up	9000	a8:f9:4b:ac:4d:16	5 hours, 25 minutes and 2 seconds	Routerport
gi1/0/2	Up	Up	1500	a8:f9:4b:ac:4d:17	4 days, 4 hours, 49 minutes and 40 seconds	Switchport
gi1/0/3	Up	Up	1800	a8:f9:4b:ac:4d:18	4 days, 1 hour, 49 minutes and 38 seconds	Switchport
bridge 2	Up	Up	1500	a8:f9:4b:ac:4d:15	1 day, 1 hour, 27 minutes and 28 seconds	Routerport

CE1 посылает пакеты размером 1500 байт, CE2 – 1800 байт соответственно. Так как MTU бридж-домена меньше, чем MTU пакета от CE2, то пакет от CE2 будет отброшен перед попаданием в бридж-домен. Аналогичные действия будут, если MTU интерфейса, смотрящего в сторону mpls-core (gi1/0/1), меньше чем MTU, приходящих от CE-пакетов (с учетом MPLS-заголовка).

Схожее поведение и при прохождении трафика в L3VPN-сервисе:

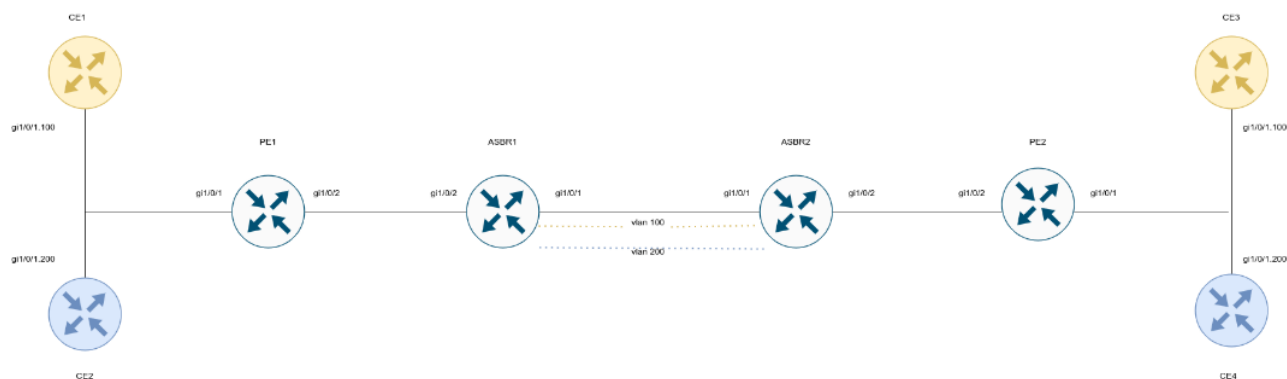


Если CE1 пошлет пакет с большим MTU, чем на интерфейсе, смотрящим в сторону клиента (gi1/0/2) или в сторону mpls-core (gi1/0/1), то пакет будет отброшен.

13.11. Inter-AS Option A

Рассмотрим примеры настройки на базе построения сервисов l3vpn и l2vpn. Главная особенность inter-AS Option A – отсутствие mpls-меток в трафике при передаче между ASBR. Для разделения трафика клиентских сервисов между ASBR обычно используют VRF для l3vpn или тегирование (dot1q, q-in-q) для сервисов l2vpn.

13.11.1. L2VPN



Настроим CE:

CE1

```

RTT# config
RTT(config)# hostname CE1
RTT(config)# interface gigabitethernet 1/0/1.100
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# ip address 192.168.1.1/24
RTT(config-if-gi)# do com
RTT(config-if-gi)# do conf
  
```

CE2

```
RTT# config
RTT(config)# hostname CE2
RTT(config)# interface gigabitethernet 1/0/1.200
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# ip address 192.168.2.1/24
RTT(config-if-gi)# do com
RTT(config-if-gi)# do conf
```

CE3

```
RTT# config
RTT(config)# hostname CE3
RTT(config)# interface gigabitethernet 1/0/1.100
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# ip address 192.168.1.2/24
RTT(config-if-gi)# do com
RTT(config-if-gi)# do conf
```

CE4

```
RTT# config
RTT(config)# hostname CE4
RTT(config)# interface gigabitethernet 1/0/1.200
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# ip address 192.168.2.2/24
RTT(config-if-gi)# do com
RTT(config-if-gi)# do conf
```

Произведем настройку PE1 и PE2. Анонсирование сервисных меток возложим на протокол BGP (Kompella mode):

PE1

```
RTT(config)# hostname PE1
RTT(config)# system jumbo-frames
RTT(config)# router bgp log-neighbor-changes
RTT(config)# router bgp 65500
RTT(config-bgp)# neighbor 10.10.1.2
RTT(config-bgp-neighbor)# remote-as 65500
RTT(config-bgp-neighbor)# update-source 10.10.1.1
RTT(config-bgp-neighbor)# address-family l2vpn vpls
RTT(config-bgp-neighbor-af)# send-community extended
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# enable
RTT(config-bgp)# exit
RTT(config)#
RTT(config)# router ospf 1
RTT(config-ospf)# area 0.0.0.0
```

```
RTT(config-ospf-area)# enable
RTT(config-ospf-area)# exit
RTT(config-ospf)# enable
RTT(config-ospf)# exit
RTT(config)#
RTT(config)# bridge 100
RTT(config-bridge)# enable
RTT(config-bridge)# exit
RTT(config)# bridge 200
RTT(config-bridge)# enable
RTT(config-bridge)# exit
RTT(config)#
RTT(config)# interface gigabitethernet 1/0/1.100
RTT(config-if-sub)# description "to CE1"
RTT(config-if-sub)# bridge-group 100
RTT(config-if-sub)# exit
RTT(config)# interface gigabitethernet 1/0/1.200
RTT(config-if-sub)# description "to CE2"
RTT(config-if-sub)# bridge-group 200
RTT(config-if-sub)# exit
RTT(config)# interface gigabitethernet 1/0/2
RTT(config-if-gi)# mtu 1522
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# ip address 10.100.0.1/30
RTT(config-if-gi)# ip ospf instance 1
RTT(config-if-gi)# ip ospf
RTT(config-if-gi)# exit
RTT(config)# interface loopback 1
RTT(config-loopback)# ip address 10.10.1.1/32
RTT(config-loopback)# ip ospf instance 1
RTT(config-loopback)# ip ospf
RTT(config-loopback)# exit
RTT(config)# mpls
RTT(config-mpls)# ldp
RTT(config-ldp)# router-id 10.10.1.1
RTT(config-ldp)# address-family ipv4
RTT(config-ldp-af-ipv4)# interface gigabitethernet 1/0/2
RTT(config-ldp-af-ipv4-if)# exit
RTT(config-ldp-af-ipv4)# exit
RTT(config-ldp)# enable
RTT(config-ldp)# exit
RTT(config-mpls)# l2vpn
RTT(config-l2vpn)# vpls CE1
RTT(config-l2vpn-vpls)# bridge-group 100
RTT(config-l2vpn-vpls)# autodiscovery bgp
RTT(config-bgp)# vpn id 1
RTT(config-bgp)# ve id 2
RTT(config-bgp)# rd 65500:1
RTT(config-bgp)# route-target export 65500:1
RTT(config-bgp)# route-target import 65500:1
RTT(config-bgp)# exit
RTT(config-l2vpn-vpls)# enable
RTT(config-l2vpn-vpls)# exit
RTT(config-l2vpn)# vpls CE2
RTT(config-l2vpn-vpls)# bridge-group 200
RTT(config-l2vpn-vpls)# autodiscovery bgp
RTT(config-bgp)# vpn id 2
RTT(config-bgp)# ve id 2
RTT(config-bgp)# rd 65500:2
```

```
RTT(config-bgp)# route-target export 65500:2
RTT(config-bgp)# route-target import 65500:2
RTT(config-bgp)# exit
RTT(config-l2vpn-vpls)# enable
RTT(config-l2vpn-vpls)# exit
RTT(config-l2vpn)# exit
RTT(config-mpls)# forwarding interface gigabitethernet 1/0/2
RTT(config-mpls)# exit
RTT(config)# do com
RTT(config)# do conf
```

PE2

```
RTT(config)# hostname RTT
RTT(config)# system jumbo-frames
RTT(config)#
RTT(config)# router bgp log-neighbor-changes
RTT(config)# router bgp 65500
RTT(config-bgp)# router-id 10.11.1.1
RTT(config-bgp)# neighbor 10.11.1.2
RTT(config-bgp-neighbor)# remote-as 65500
RTT(config-bgp-neighbor)# update-source 10.11.1.1
RTT(config-bgp-neighbor)# address-family l2vpn vpls
RTT(config-bgp-neighbor-af)# send-community extended
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# enable
RTT(config-bgp)# exit
RTT(config)#
RTT(config)# router ospf 1
RTT(config-ospf)# area 0.0.0.0
RTT(config-ospf-area)# enable
RTT(config-ospf-area)# exit
RTT(config-ospf)# enable
RTT(config-ospf)# exit
RTT(config)#
RTT(config)# bridge 100
RTT(config-bridge)# enable
RTT(config-bridge)# exit
RTT(config)# bridge 200
RTT(config-bridge)# enable
RTT(config-bridge)# exit
RTT(config)#
RTT(config)# interface gigabitethernet 1/0/1.100
RTT(config-if-sub)# description "to CE3"
RTT(config-if-sub)# bridge-group 100
RTT(config-if-sub)# exit
RTT(config)# interface gigabitethernet 1/0/1.200
RTT(config-if-sub)# description "to CE4"
RTT(config-if-sub)# bridge-group 200
RTT(config-if-sub)# exit
RTT(config)# interface gigabitethernet 1/0/2
RTT(config-if-gi)# mtu 1522
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# ip address 10.101.0.1/30
```

```
RTT(config-if-gi)# ip ospf instance 1
RTT(config-if-gi)# ip ospf
RTT(config-if-gi)# exit
RTT(config)# interface loopback 1
RTT(config-loopback)# ip address 10.11.1.1/32
RTT(config-loopback)# ip ospf instance 1
RTT(config-loopback)# ip ospf
RTT(config-loopback)# exit
RTT(config)# mpls
RTT(config-mpls)# ldp
RTT(config-ldp)# router-id 10.11.1.1
RTT(config-ldp)# address-family ipv4
RTT(config-ldp-af-ipv4)# interface gigabitethernet 1/0/2
RTT(config-ldp-af-ipv4-if)# exit
RTT(config-ldp-af-ipv4)# exit
RTT(config-ldp)# enable
RTT(config-ldp)# exit
RTT(config-mpls)# l2vpn
RTT(config-l2vpn)# vpls CE1
RTT(config-l2vpn-vpls)# bridge-group 100
RTT(config-l2vpn-vpls)# autodiscovery bgp
RTT(config-bgp)# vpn id 1
RTT(config-bgp)# ve id 2
RTT(config-bgp)# rd 65500:1
RTT(config-bgp)# route-target export 65500:1
RTT(config-bgp)# route-target import 65500:1
RTT(config-bgp)# exit
RTT(config-l2vpn-vpls)# enable
RTT(config-l2vpn-vpls)# exit
RTT(config-l2vpn)# vpls CE2
RTT(config-l2vpn-vpls)# bridge-group 200
RTT(config-l2vpn-vpls)# autodiscovery bgp
RTT(config-bgp)# vpn id 2
RTT(config-bgp)# ve id 2
RTT(config-bgp)# rd 65500:2
RTT(config-bgp)# route-target export 65500:2
RTT(config-bgp)# route-target import 65500:2
RTT(config-bgp)# exit
RTT(config-l2vpn-vpls)# enable
RTT(config-l2vpn-vpls)# exit
RTT(config-l2vpn)# exit
RTT(config-mpls)# forwarding interface gigabitethernet 1/0/2
RTT(config-mpls)# exit
RTT(config)# do com
RTT(config)# do conf
```

Настроим ASBR1 и ASBR2. Для разделения трафика от CE1 и CE2 в сторону ASBR2 сделаем интерфейс gi1/0/1 транковым. Vlan 100 и 200 будут предназначены для трафика от CE1 и CE2 соответственно:

ASBR1

```
RTT(config)# hostname ASBR1
RTT(config)#
RTT(config)# system jumbo-frames
RTT(config)#
RTT(config)# vlan 100,200
```

```
RTT(config-vlan)# exit
RTT(config)#
RTT(config)# router bgp 65500
RTT(config-bgp)# router-id 10.10.1.2
RTT(config-bgp)# neighbor 10.10.1.1
RTT(config-bgp-neighbor)# remote-as 65500
RTT(config-bgp-neighbor)# update-source 10.10.1.2
RTT(config-bgp-neighbor)# address-family l2vpn vpls
RTT(config-bgp-neighbor-af)# send-community extended
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# enable
RTT(config-bgp)# exit
RTT(config)#
RTT(config)# router ospf 1
RTT(config-ospf)# area 0.0.0.0
RTT(config-ospf-area)# enable
RTT(config-ospf-area)# exit
RTT(config-ospf)# enable
RTT(config-ospf)# exit
RTT(config)#
RTT(config)# bridge 10
RTT(config-bridge)# vlan 100
RTT(config-bridge)# enable
RTT(config-bridge)# exit
RTT(config)# bridge 20
RTT(config-bridge)# vlan 200
RTT(config-bridge)# enable
RTT(config-bridge)# exit
RTT(config)#
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# description "to ASBR2"
RTT(config-if-gi)# mode switchport
RTT(config-if-gi)# spanning-tree disable
RTT(config-if-gi)# switchport forbidden default-vlan
RTT(config-if-gi)# switchport mode trunk
RTT(config-if-gi)# switchport trunk allowed vlan add 100,200
RTT(config-if-gi)# exit
RTT(config)# interface gigabitethernet 1/0/2
RTT(config-if-gi)# description "to PE1"
RTT(config-if-gi)# mtu 1522
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# ip address 10.100.0.2/30
RTT(config-if-gi)# ip ospf instance 1
RTT(config-if-gi)# ip ospf
RTT(config-if-gi)# exit
RTT(config)# interface loopback 1
RTT(config-loopback)# ip address 10.10.1.2/32
RTT(config-loopback)# ip ospf instance 1
RTT(config-loopback)# ip ospf
RTT(config-loopback)# exit
RTT(config)# mpls
RTT(config-mpls)# ldp
RTT(config-ldp)# router-id 10.10.1.2
RTT(config-ldp)# address-family ipv4
RTT(config-ldp-af-ipv4)# interface gigabitethernet 1/0/2
```

```
RTT(config-ldp-af-ipv4-if) #      exit
RTT(config-ldp-af-ipv4) #      exit
RTT(config-ldp) #      enable
RTT(config-ldp) #      exit
RTT(config-mpls) #      l2vpn
RTT(config-l2vpn) #      vpls CE1
RTT(config-l2vpn-vpls) #      bridge-group 10
RTT(config-l2vpn-vpls) #      autodiscovery bgp
RTT(config-bgp) #      vpn id 1
RTT(config-bgp) #      ve id 1
RTT(config-bgp) #      rd 65500:1
RTT(config-bgp) #      route-target export 65500:1
RTT(config-bgp) #      route-target import 65500:1
RTT(config-bgp) #      exit
RTT(config-l2vpn-vpls) #      enable
RTT(config-l2vpn-vpls) #      exit
RTT(config-l2vpn) #      vpls CE2
RTT(config-l2vpn-vpls) #      bridge-group 20
RTT(config-l2vpn-vpls) #      autodiscovery bgp
RTT(config-bgp) #      vpn id 2
RTT(config-bgp) #      ve id 1
RTT(config-bgp) #      rd 65500:2
RTT(config-bgp) #      route-target export 65500:2
RTT(config-bgp) #      route-target import 65500:2
RTT(config-bgp) #      exit
RTT(config-l2vpn-vpls) #      enable
RTT(config-l2vpn-vpls) #      exit
RTT(config-l2vpn) #      exit
RTT(config-mpls) #      forwarding interface gigabitethernet 1/0/2
RTT(config-mpls) #      exit
RTT(config) #      do com
RTT(config) #      do conf
```

ASBR2

```
RTT(config) #      hostname ASBR2
RTT(config) #
RTT(config) #      system jumbo-frames
RTT(config) #
RTT(config) #      vlan 100,200
RTT(config-vlan) #      exit
RTT(config) #
RTT(config) #      router bgp 65500
RTT(config-bgp) #      router-id 10.10.1.2
RTT(config-bgp) #      neighbor 10.10.1.1
RTT(config-bgp-neighbor) #      remote-as 65500
RTT(config-bgp-neighbor) #      update-source 10.10.1.2
RTT(config-bgp-neighbor) #      address-family l2vpn vpls
RTT(config-bgp-neighbor-af) #      send-community extended
RTT(config-bgp-neighbor-af) #      enable
RTT(config-bgp-neighbor-af) #      exit
RTT(config-bgp-neighbor) #      enable
RTT(config-bgp-neighbor) #      exit
RTT(config-bgp) #      enable
RTT(config-bgp) #      exit
RTT(config) #
RTT(config) #      router ospf 1
```



```
RTT(config-ospf)# area 0.0.0.0
RTT(config-ospf-area)# enable
RTT(config-ospf-area)# exit
RTT(config-ospf)# enable
RTT(config-ospf)# exit
RTT(config)#
RTT(config)# bridge 10
RTT(config-bridge)# vlan 100
RTT(config-bridge)# enable
RTT(config-bridge)# exit
RTT(config)# bridge 20
RTT(config-bridge)# vlan 200
RTT(config-bridge)# enable
RTT(config-bridge)# exit
RTT(config)#
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# description "to ASBR1"
RTT(config-if-gi)# mode switchport
RTT(config-if-gi)# spanning-tree disable
RTT(config-if-gi)# switchport forbidden default-vlan
RTT(config-if-gi)# switchport mode trunk
RTT(config-if-gi)# switchport trunk allowed vlan add 100,200
RTT(config-if-gi)# exit
RTT(config)# interface gigabitethernet 1/0/2
RTT(config-if-gi)# description "to PE1"
RTT(config-if-gi)# mtu 1522
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# ip address 10.100.0.2/30
RTT(config-if-gi)# ip ospf instance 1
RTT(config-if-gi)# ip ospf
RTT(config-if-gi)# exit
RTT(config)# interface loopback 1
RTT(config-loopback)# ip address 10.10.1.2/32
RTT(config-loopback)# ip ospf instance 1
RTT(config-loopback)# ip ospf
RTT(config-loopback)# exit
RTT(config)# mpls
RTT(config-mpls)# ldp
RTT(config-ldp)# router-id 10.10.1.2
RTT(config-ldp)# address-family ipv4
RTT(config-ldp-af-ipv4)# interface gigabitethernet 1/0/2
RTT(config-ldp-af-ipv4-if)# exit
RTT(config-ldp-af-ipv4)# exit
RTT(config-ldp)# enable
RTT(config-ldp)# exit
RTT(config-mpls)# l2vpn
RTT(config-l2vpn)# vpls CE1
RTT(config-l2vpn-vpls)# bridge-group 10
RTT(config-l2vpn-vpls)# autodiscovery bgp
RTT(config-bgp)# vpn id 1
RTT(config-bgp)# ve id 1
RTT(config-bgp)# rd 65500:1
RTT(config-bgp)# route-target export 65500:1
RTT(config-bgp)# route-target import 65500:1
RTT(config-bgp)# exit
RTT(config-l2vpn-vpls)# enable
RTT(config-l2vpn-vpls)# exit
RTT(config-l2vpn)# vpls CE2
```

```

RTT(config-l2vpn-vpls)#      bridge-group 20
RTT(config-l2vpn-vpls)#      autodiscovery bgp
RTT(config-bgp)#             vpn id 2
RTT(config-bgp)#             ve id 1
RTT(config-bgp)#             rd 65500:2
RTT(config-bgp)#             route-target export 65500:2
RTT(config-bgp)#             route-target import 65500:2
RTT(config-bgp)#             exit
RTT(config-l2vpn-vpls)#      enable
RTT(config-l2vpn-vpls)#      exit
RTT(config-l2vpn)#           exit
RTT(config-mpls)#           forwarding interface gigabitethernet 1/0/2
RTT(config-mpls)#           exit
RTT(config)#                 do com
RTT(config)#                 do conf

```

Проверим назначение меток, статус сервисов, а также сетевую доступность между СЕ:

Информация о метках

```

ASBR2# sh bgp l2vpn vpls all
Status codes: * - valid, > - best, i - internal, S - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

```

Codes	Route	Distinguisher	VID	VBO	VBS	Next hop	Metric	LocPrf	Weight	Path
*>i	65500:1		2	1	10	10.11.1.1	--	100	0	i
*>i	65500:2		2	1	10	10.11.1.1	--	100	0	i
*>	65500:1		1	1	10	--	--	--	--	
*>	65500:2		1	1	10	--	--	--	--	

```

ASBR2# sh mpls forwarding-table
Local      Outgoing Prefix          Outgoing      Next Hop
label      label      or tunnel ID    Interface
-----
56          imp-null  10.11.1.1/32    gi1/0/2       10.101.0.1
47          37        PW ID 1         --            10.11.1.1
37          47        PW ID 2         --            10.11.1.1

```

Статус сервисов

```

ASBR2# sh mpls l2vpn vpls
VPLS: CE1
  bridge 10:
    MTU:      1500
    Status:   Up
  PWs:
    PW ID 1, Neighbor 10.11.1.1:
      MTU:      1500
      Last change: 00:16:59
      Status:    Up
VPLS: CE2
  bridge 20:
    MTU:      1500
    Status:   Up
  PWs:

```

```
PW ID 2, Neighbor 10.11.1.1:
MTU:      1500
Last change: 00:16:59
Status:    Up
```

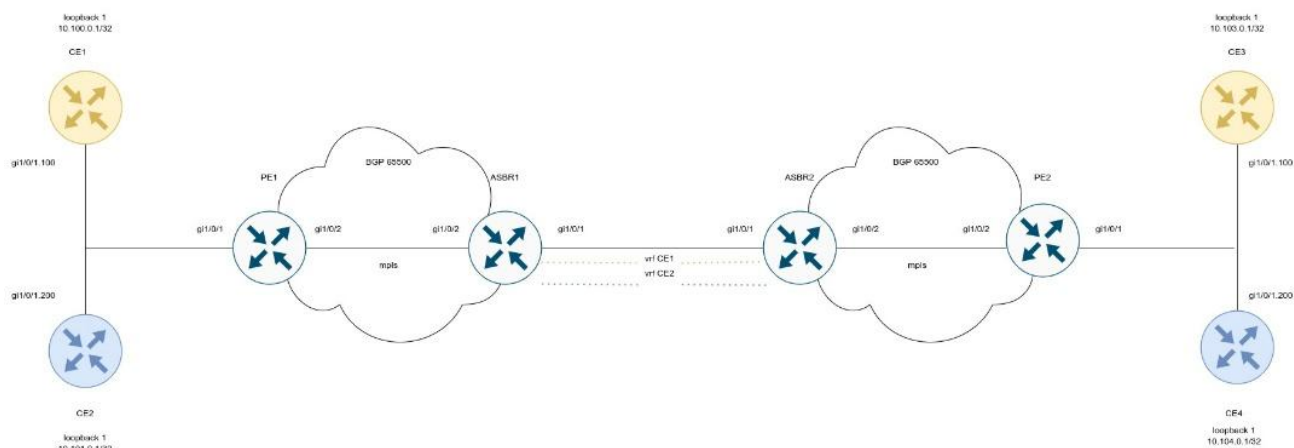
Проверка сетевой доступности

```
CE1# ping 192.168.1.2 detailed
PING 192.168.1.2 (192.168.1.2) 56 bytes of data.
64 bytes from 192.168.1.2: icmp_seq=1 ttl=0 time=1.08 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=0 time=1.06 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=0 time=1.01 ms
64 bytes from 192.168.1.2: icmp_seq=4 ttl=0 time=0.971 ms
64 bytes from 192.168.1.2: icmp_seq=5 ttl=0 time=0.972 ms
```

```
CE2# ping 192.168.2.2 detailed packets
PING 192.168.2.2 (192.168.2.2) 56 bytes of data.
64 bytes from 192.168.2.2: icmp_seq=1 ttl=0 time=1.17 ms
64 bytes from 192.168.2.2: icmp_seq=2 ttl=0 time=0.972 ms
64 bytes from 192.168.2.2: icmp_seq=3 ttl=0 time=0.960 ms
64 bytes from 192.168.2.2: icmp_seq=4 ttl=0 time=1.04 ms
64 bytes from 192.168.2.2: icmp_seq=5 ttl=0 time=0.976 ms
```

```
ASBR2# sh mac address-table bridge 10
VID      MAC Address      Interface      Type
-----
--      e4:5a:d4:01:b9:73   vlan 100      Dynamic
--      e4:5a:d4:a1:34:61   dypseudowire 1_10.11.1.1  Dynamic
2 valid mac entries
ASBR2# sh mac address-table bridge 20
VID      MAC Address      Interface      Type
-----
--      e4:5a:d4:01:c1:80   vlan 200      Dynamic
--      e4:5a:d4:a1:34:61   dypseudowire 2_10.11.1.1  Dynamic
2 valid mac entries
```

13.11.2. L3VPN



CE1

```
RTT(config)# hostname CE1
RTT(config)#
RTT(config)# route-map BGP
RTT(config-route-map)# rule 1
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
RTT(config)# router bgp 65501
RTT(config-bgp)# neighbor 192.168.1.2
RTT(config-bgp-neighbor)# remote-as 65500
RTT(config-bgp-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af)# route-map BGP out
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# address-family ipv4 unicast
RTT(config-bgp-af)# network 10.110.0.1/32
RTT(config-bgp-af)# exit
RTT(config-bgp)# enable
RTT(config-bgp)# exit
RTT(config)# interface gigabitethernet 1/0/1.100
RTT(config-if-sub)# ip firewall disable
RTT(config-if-sub)# ip address 192.168.1.1/30
RTT(config-if-sub)# exit
RTT(config)# interface loopback 1
RTT(config-loopback)# ip address 10.110.0.1/32
RTT(config-loopback)# exit
RTT(config)# do com
RTT(config)# do conf
```

CE2

```
RTT(config)# hostname CE2
RTT(config)#
RTT(config)# route-map BGP
RTT(config-route-map)# rule 1
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
RTT(config)# router bgp 65501
RTT(config-bgp)# neighbor 192.168.2.2
RTT(config-bgp-neighbor)# remote-as 65500
RTT(config-bgp-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af)# route-map BGP out
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# address-family ipv4 unicast
RTT(config-bgp-af)# network 10.112.0.1/32
RTT(config-bgp-af)# exit
RTT(config-bgp)# enable
RTT(config-bgp)# exit
RTT(config)# interface gigabitethernet 1/0/1.100
```

```
RTT(config-if-sub)# ip firewall disable
RTT(config-if-sub)# ip address 192.168.2.1/30
RTT(config-if-sub)# exit
RTT(config)#
RTT(config)# interface loopback 1
RTT(config-loopback)# ip address 10.112.0.1/32
RTT(config-loopback)# exit
RTT(config)# do com
RTT(config)# do conf
```

CE3

```
RTT(config)# hostname CE3
RTT(config)#
RTT(config)# route-map BGP
RTT(config-route-map)# rule 1
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
RTT(config)# router bgp 65501
RTT(config-bgp)# neighbor 192.168.3.2
RTT(config-bgp-neighbor)# remote-as 65500
RTT(config-bgp-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af)# route-map BGP out
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# address-family ipv4 unicast
RTT(config-bgp-af)# network 10.113.0.1/32
RTT(config-bgp-af)# exit
RTT(config-bgp)# enable
RTT(config-bgp)# exit
RTT(config)# interface gigabitethernet 1/0/1.100
RTT(config-if-sub)# ip firewall disable
RTT(config-if-sub)# ip address 192.168.3.1/30
RTT(config-if-sub)# exit
RTT(config)#
RTT(config)# interface loopback 1
RTT(config-loopback)# ip address 10.113.0.1/32
RTT(config-loopback)# exit
RTT(config)# do com
RTT(config)# do conf
```

CE4

```
RTT(config)# hostname CE4
RTT(config)#
RTT(config)# route-map BGP
RTT(config-route-map)# rule 1
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
RTT(config)# router bgp 65501
RTT(config-bgp)# neighbor 192.168.4.2
RTT(config-bgp-neighbor)# remote-as 65500
RTT(config-bgp-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af)# route-map BGP out
```

```
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# address-family ipv4 unicast
RTT(config-bgp-af)# network 10.114.0.1/32
RTT(config-bgp-af)# exit
RTT(config-bgp)# enable
RTT(config-bgp)# exit
RTT(config)# interface gigabitethernet 1/0/1.100
RTT(config-if-sub)# ip firewall disable
RTT(config-if-sub)# ip address 192.168.4.1/30
RTT(config-if-sub)# exit
RTT(config)#
RTT(config)# interface loopback 1
RTT(config-loopback)# ip address 10.114.0.1/32
RTT(config-loopback)# exit
RTT(config)# do com
RTT(config)# do conf
```

Произведем настройку PE1 и PE2:

PE1

```
RTT(config)# hostname PE1
RTT(config)#
RTT(config)# ip vrf CE1
RTT(config-vrf)# ip protocols bgp max-routes 100
RTT(config-vrf)# rd 65500:1
RTT(config-vrf)# route-target export 65500:1
RTT(config-vrf)# route-target import 65500:1
RTT(config-vrf)# exit
RTT(config)# ip vrf CE2
RTT(config-vrf)# ip protocols bgp max-routes 100
RTT(config-vrf)# rd 65500:2
RTT(config-vrf)# route-target export 65500:2
RTT(config-vrf)# route-target import 65500:2
RTT(config-vrf)# exit
RTT(config)#
RTT(config)# system jumbo-frames
RTT(config)#
RTT(config)# route-map BGP
RTT(config-route-map)# rule 1
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
RTT(config)# router bgp log-neighbor-changes
RTT(config)# router bgp 65500
RTT(config-bgp)# neighbor 10.10.1.2
RTT(config-bgp-neighbor)# remote-as 65500
RTT(config-bgp-neighbor)# update-source 10.10.1.1
RTT(config-bgp-neighbor)# address-family vpnv4 unicast
RTT(config-bgp-neighbor-af)# send-community extended
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# enable
```

```
RTT(config-bgp)# vrf CE1
RTT(config-bgp-vrf)# neighbor 192.168.1.1
RTT(config-bgp-vrf-neighbor)# remote-as 65501
RTT(config-bgp-vrf-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af-vrf)# route-map BGP out
RTT(config-bgp-neighbor-af-vrf)# enable
RTT(config-bgp-neighbor-af-vrf)# exit
RTT(config-bgp-vrf-neighbor)# enable
RTT(config-bgp-vrf-neighbor)# exit
RTT(config-bgp-vrf)# address-family ipv4 unicast
RTT(config-bgp-vrf-af)# redistribute bgp 65500 route-map BGP
RTT(config-bgp-vrf-af)# exit
RTT(config-bgp-vrf)# enable
RTT(config-bgp-vrf)# exit
RTT(config-bgp)# vrf CE2
RTT(config-bgp-vrf)# neighbor 192.168.2.1
RTT(config-bgp-vrf-neighbor)# remote-as 65501
RTT(config-bgp-vrf-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af-vrf)# route-map BGP out
RTT(config-bgp-neighbor-af-vrf)# enable
RTT(config-bgp-neighbor-af-vrf)# exit
RTT(config-bgp-vrf-neighbor)# enable
RTT(config-bgp-vrf-neighbor)# exit
RTT(config-bgp-vrf)# address-family ipv4 unicast
RTT(config-bgp-vrf-af)# redistribute bgp 65500 route-map BGP
RTT(config-bgp-vrf-af)# exit
RTT(config-bgp-vrf)# enable
RTT(config-bgp-vrf)# exit
RTT(config-bgp)# exit
RTT(config)#
RTT(config)# router ospf 1
RTT(config-ospf)# area 0.0.0.0
RTT(config-ospf-area)# enable
RTT(config-ospf-area)# exit
RTT(config-ospf)# enable
RTT(config-ospf)# exit
RTT(config)#
RTT(config)# interface gigabitethernet 1/0/1.100
RTT(config-if-sub)# ip vrf forwarding CE1
RTT(config-if-sub)# description "to CE1"
RTT(config-if-sub)# ip firewall disable
RTT(config-if-sub)# ip address 192.168.1.2/30
RTT(config-if-sub)# exit
RTT(config)# interface gigabitethernet 1/0/1.200
RTT(config-if-sub)# ip vrf forwarding CE2
RTT(config-if-sub)# description "to CE2"
RTT(config-if-sub)# ip firewall disable
RTT(config-if-sub)# ip address 192.168.2.2/30
RTT(config-if-sub)# exit
RTT(config)# interface gigabitethernet 1/0/2
RTT(config-if-gi)# mtu 1522
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# ip address 10.100.0.1/30
RTT(config-if-gi)# ip ospf instance 1
RTT(config-if-gi)# ip ospf
RTT(config-if-gi)# exit
RTT(config)# interface loopback 1
RTT(config-loopback)# ip address 10.10.1.1/32
```

```
RTT(config-loopback)# ip ospf instance 1
RTT(config-loopback)# ip ospf
RTT(config-loopback)# exit
RTT(config)# mpls
RTT(config-mpls)# ldp
RTT(config-ldp)# router-id 10.10.1.1
RTT(config-ldp)# address-family ipv4
RTT(config-ldp-af-ipv4)# interface gigabitethernet 1/0/2
RTT(config-ldp-af-ipv4-if)# exit
RTT(config-ldp-af-ipv4)# exit
RTT(config-ldp)# enable
RTT(config-ldp)# exit
RTT(config-mpls)# forwarding interface gigabitethernet 1/0/2
RTT(config-mpls)# exit
RTT(config)# do com
RTT(config)# do conf
```

PE2

```
RTT(config)# hostname PE2
RTT(config)#
RTT(config)# ip vrf CE1
RTT(config-vrf)# ip protocols bgp max-routes 100
RTT(config-vrf)# rd 65500:1
RTT(config-vrf)# route-target export 65500:1
RTT(config-vrf)# route-target import 65500:1
RTT(config-vrf)# exit
RTT(config)# ip vrf CE2
RTT(config-vrf)# ip protocols bgp max-routes 100
RTT(config-vrf)# rd 65500:2
RTT(config-vrf)# route-target export 65500:2
RTT(config-vrf)# route-target import 65500:2
RTT(config-vrf)# exit
RTT(config)#
RTT(config)# system jumbo-frames
RTT(config)#
RTT(config)# route-map BGP
RTT(config-route-map)# rule 1
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
RTT(config)# router bgp log-neighbor-changes
RTT(config)# router bgp 65500
RTT(config-bgp)# router-id 10.11.1.1
RTT(config-bgp)# neighbor 10.11.1.2
RTT(config-bgp-neighbor)# remote-as 65500
RTT(config-bgp-neighbor)# update-source 10.11.1.1
RTT(config-bgp-neighbor)# address-family vpnv4 unicast
RTT(config-bgp-neighbor-af)# send-community extended
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# enable
RTT(config-bgp)# vrf CE1
RTT(config-bgp-vrf)# neighbor 192.168.3.1
RTT(config-bgp-vrf-neighbor)# remote-as 65501
RTT(config-bgp-vrf-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af-vrf)# route-map BGP out
```



```
RTT(config-bgp-neighbor-af-vrf)# enable
RTT(config-bgp-neighbor-af-vrf)# exit
RTT(config-bgp-vrf-neighbor)# enable
RTT(config-bgp-vrf-neighbor)# exit
RTT(config-bgp-vrf)# address-family ipv4 unicast
RTT(config-bgp-vrf-af)# redistribute bgp 65500 route-map BGP
RTT(config-bgp-vrf-af)# exit
RTT(config-bgp-vrf)# enable
RTT(config-bgp-vrf)# exit
RTT(config-bgp)# vrf CE2
RTT(config-bgp-vrf)# neighbor 192.168.4.1
RTT(config-bgp-vrf-neighbor)# remote-as 65501
RTT(config-bgp-vrf-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af-vrf)# route-map BGP out
RTT(config-bgp-neighbor-af-vrf)# enable
RTT(config-bgp-neighbor-af-vrf)# exit
RTT(config-bgp-vrf-neighbor)# enable
RTT(config-bgp-vrf-neighbor)# exit
RTT(config-bgp-vrf)# address-family ipv4 unicast
RTT(config-bgp-vrf-af)# redistribute bgp 65500 route-map BGP
RTT(config-bgp-vrf-af)# exit
RTT(config-bgp-vrf)# enable
RTT(config-bgp-vrf)# exit
RTT(config-bgp)# exit
RTT(config)#
RTT(config)# router ospf 1
RTT(config-ospf)# area 0.0.0.0
RTT(config-ospf-area)# enable
RTT(config-ospf-area)# exit
RTT(config-ospf)# enable
RTT(config-ospf)# exit
RTT(config)#
RTT(config)# interface gigabitethernet 1/0/1.100
RTT(config-if-sub)# ip vrf forwarding CE1
RTT(config-if-sub)# description "to CE3"
RTT(config-if-sub)# ip firewall disable
RTT(config-if-sub)# ip address 192.168.3.2/30
RTT(config-if-sub)# exit
RTT(config)# interface gigabitethernet 1/0/1.200
RTT(config-if-sub)# ip vrf forwarding CE2
RTT(config-if-sub)# description "to CE4"
RTT(config-if-sub)# ip firewall disable
RTT(config-if-sub)# ip address 192.168.4.2/30
RTT(config-if-sub)# exit
RTT(config)# interface gigabitethernet 1/0/2
RTT(config-if-gi)# mtu 1522
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# ip address 10.101.0.1/30
RTT(config-if-gi)# ip ospf instance 1
RTT(config-if-gi)# ip ospf
RTT(config-if-gi)# exit
RTT(config)# interface loopback 1
RTT(config-loopback)# ip address 10.11.1.1/32
RTT(config-loopback)# ip ospf instance 1
RTT(config-loopback)# ip ospf
RTT(config-loopback)# exit
RTT(config)# mpls
RTT(config-mpls)# ldp
```

```
RTT(config-ldp)# router-id 10.11.1.1
RTT(config-ldp)# address-family ipv4
RTT(config-ldp-af-ipv4)# interface gigabitethernet 1/0/2
RTT(config-ldp-af-ipv4-if)# exit
RTT(config-ldp-af-ipv4)# exit
RTT(config-ldp)# enable
RTT(config-ldp)# exit
RTT(config-mpls)# forwarding interface gigabitethernet 1/0/2
RTT(config-mpls)# exit
RTT(config)# do com
RTT(config)# do conf
```

Настроим ASBR1 и ASBR2. Для передачи маршрутной информации между ними воспользуемся протоколом OSPF в соответствующих VRF:

ASBR1

```
RTT(config)# hostname ASBR1
RTT(config)#
RTT(config)# ip vrf CE1
RTT(config-vrf)# ip protocols ospf max-routes 100
RTT(config-vrf)# rd 65500:1
RTT(config-vrf)# route-target export 65500:1
RTT(config-vrf)# route-target import 65500:1
RTT(config-vrf)# exit
RTT(config)# ip vrf CE2
RTT(config-vrf)# ip protocols ospf max-routes 100
RTT(config-vrf)# rd 65500:2
RTT(config-vrf)# route-target export 65500:2
RTT(config-vrf)# route-target import 65500:2
RTT(config-vrf)# exit
RTT(config)#
RTT(config)# system jumbo-frames
RTT(config)#
RTT(config)# vlan 100,200
RTT(config-vlan)# exit
RTT(config)#
RTT(config)# router bgp 65500
RTT(config-bgp)# router-id 10.10.1.2
RTT(config-bgp)# neighbor 10.10.1.1
RTT(config-bgp-neighbor)# remote-as 65500
RTT(config-bgp-neighbor)# update-source 10.10.1.2
RTT(config-bgp-neighbor)# address-family vpnv4 unicast
RTT(config-bgp-neighbor-af)# send-community extended
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# enable
RTT(config-bgp)# vrf CE1
RTT(config-bgp-vrf)# address-family ipv4 unicast
RTT(config-bgp-vrf-af)# redistribute ospf 1 intra-area inter-area
external1 external2
RTT(config-bgp-vrf-af)# exit
RTT(config-bgp-vrf)# exit
RTT(config-bgp)# vrf CE2
RTT(config-bgp-vrf)# address-family ipv4 unicast
```

```
RTT(config-bgp-vrf-af)# redistribute ospf 1 intra-area inter-area
external1 external2
RTT(config-bgp-vrf-af)# exit
RTT(config-bgp-vrf)# exit
RTT(config-bgp)# exit
RTT(config)#
RTT(config)# router ospf log-adjacency-changes
RTT(config)# router ospf 1
RTT(config-ospf)# area 0.0.0.0
RTT(config-ospf-area)# enable
RTT(config-ospf-area)# exit
RTT(config-ospf)# enable
RTT(config-ospf)# exit
RTT(config)# router ospf 1 vrf CE1
RTT(config-ospf)# redistribute bgp 65500
RTT(config-ospf)# area 0.0.0.0
RTT(config-ospf-area)# enable
RTT(config-ospf-area)# exit
RTT(config-ospf)# enable
RTT(config-ospf)# exit
RTT(config)# router ospf 1 vrf CE2
RTT(config-ospf)# area 0.0.0.0
RTT(config-ospf-area)# enable
RTT(config-ospf-area)# exit
RTT(config-ospf)# enable
RTT(config-ospf)# exit
RTT(config)#
RTT(config)# bridge 10
RTT(config-bridge)# ip vrf forwarding CE1
RTT(config-bridge)# vlan 100
RTT(config-bridge)# ip firewall disable
RTT(config-bridge)# ip address 172.16.32.1/30
RTT(config-bridge)# ip ospf instance 1
RTT(config-bridge)# ip ospf
RTT(config-bridge)# enable
RTT(config-bridge)# exit
RTT(config)# bridge 20
RTT(config-bridge)# ip vrf forwarding CE2
RTT(config-bridge)# vlan 200
RTT(config-bridge)# ip firewall disable
RTT(config-bridge)# ip address 172.16.32.5/30
RTT(config-bridge)# ip ospf instance 1
RTT(config-bridge)# ip ospf
RTT(config-bridge)# enable
RTT(config-bridge)# exit
RTT(config)#
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# description "to ASBR2"
RTT(config-if-gi)# mode switchport
RTT(config-if-gi)# mtu 1522
RTT(config-if-gi)# spanning-tree disable
RTT(config-if-gi)# switchport forbidden default-vlan
RTT(config-if-gi)# switchport mode trunk
RTT(config-if-gi)# switchport trunk allowed vlan add 100,200
RTT(config-if-gi)# exit
RTT(config)# interface gigabitethernet 1/0/2
RTT(config-if-gi)# description "to PE1"
RTT(config-if-gi)# mtu 1522
```

```
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# ip address 10.100.0.2/30
RTT(config-if-gi)# ip ospf instance 1
RTT(config-if-gi)# ip ospf
RTT(config-if-gi)# exit
RTT(config)# interface loopback 1
RTT(config-loopback)# ip address 10.10.1.2/32
RTT(config-loopback)# ip ospf instance 1
RTT(config-loopback)# ip ospf
RTT(config-loopback)# exit
RTT(config)# mpls
RTT(config-mpls)# ldp
RTT(config-ldp)# router-id 10.10.1.2
RTT(config-ldp)# address-family ipv4
RTT(config-ldp-af-ipv4)# interface gigabitethernet 1/0/2
RTT(config-ldp-af-ipv4-if)# exit
RTT(config-ldp-af-ipv4)# exit
RTT(config-ldp)# enable
RTT(config-ldp)# exit
RTT(config-mpls)# forwarding interface gigabitethernet 1/0/2
RTT(config-mpls)# exit
RTT(config)# do com
RTT(config)# do conf
```

ASBR2

```
RTT(config)# hostname ASBR2
RTT(config)#
RTT(config)# ip vrf CE1
RTT(config-vrf)# ip protocols ospf max-routes 100
RTT(config-vrf)# rd 65500:1
RTT(config-vrf)# route-target export 65500:1
RTT(config-vrf)# route-target import 65500:1
RTT(config-vrf)# exit
RTT(config)# ip vrf CE2
RTT(config-vrf)# ip protocols ospf max-routes 100
RTT(config-vrf)# rd 65500:2
RTT(config-vrf)# route-target export 65500:2
RTT(config-vrf)# route-target import 65500:2
RTT(config-vrf)# exit
RTT(config)#
RTT(config)# system jumbo-frames
RTT(config)#
RTT(config)# vlan 100,200
RTT(config-vlan)# exit
RTT(config)#
RTT(config)# router bgp 65500
RTT(config-bgp)# router-id 10.11.1.2
RTT(config-bgp)# neighbor 10.11.1.1
RTT(config-bgp-neighbor)# remote-as 65500
RTT(config-bgp-neighbor)# update-source 10.11.1.2
RTT(config-bgp-neighbor)# address-family vpnv4 unicast
RTT(config-bgp-neighbor-af)# send-community extended
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# enable
```

```
RTT(config-bgp)# vrf CE1
RTT(config-bgp-vrf)# address-family ipv4 unicast
RTT(config-bgp-vrf-af)# redistribute ospf 1 intra-area inter-area
external1 external2
RTT(config-bgp-vrf-af)# exit
RTT(config-bgp-vrf)# exit
RTT(config-bgp)# vrf CE2
RTT(config-bgp-vrf)# address-family ipv4 unicast
RTT(config-bgp-vrf-af)# redistribute ospf 1 intra-area inter-area
external1 external2
RTT(config-bgp-vrf-af)# exit
RTT(config-bgp-vrf)# exit
RTT(config-bgp)# exit
RTT(config)#
RTT(config)# router ospf log-adjacency-changes
RTT(config)# router ospf 1
RTT(config-ospf)# area 0.0.0.0
RTT(config-ospf-area)# enable
RTT(config-ospf-area)# exit
RTT(config-ospf)# enable
RTT(config-ospf)# exit
RTT(config)# router ospf 1 vrf CE1
RTT(config-ospf)# redistribute bgp 65500
RTT(config-ospf)# area 0.0.0.0
RTT(config-ospf-area)# enable
RTT(config-ospf-area)# exit
RTT(config-ospf)# enable
RTT(config-ospf)# exit
RTT(config)# router ospf 1 vrf CE2
RTT(config-ospf)# redistribute bgp 65500
RTT(config-ospf)# area 0.0.0.0
RTT(config-ospf-area)# enable
RTT(config-ospf-area)# exit
RTT(config-ospf)# enable
RTT(config-ospf)# exit
RTT(config)#
RTT(config)# bridge 10
RTT(config-bridge)# ip vrf forwarding CE1
RTT(config-bridge)# vlan 100
RTT(config-bridge)# ip firewall disable
RTT(config-bridge)# ip address 172.16.32.2/30
RTT(config-bridge)# ip ospf instance 1
RTT(config-bridge)# ip ospf
RTT(config-bridge)# enable
RTT(config-bridge)# exit
RTT(config)# bridge 20
RTT(config-bridge)# ip vrf forwarding CE2
RTT(config-bridge)# vlan 200
RTT(config-bridge)# ip firewall disable
RTT(config-bridge)# ip address 172.16.32.6/30
RTT(config-bridge)# ip ospf instance 1
RTT(config-bridge)# ip ospf
RTT(config-bridge)# enable
RTT(config-bridge)# exit
RTT(config)#
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# description "to ASBR1"
RTT(config-if-gi)# mode switchport
```

```

RTT(config-if-gi)# mtu 1522
RTT(config-if-gi)# spanning-tree disable
RTT(config-if-gi)# switchport forbidden default-vlan
RTT(config-if-gi)# switchport mode trunk
RTT(config-if-gi)# switchport trunk allowed vlan add 100,200
RTT(config-if-gi)# exit
RTT(config)# interface gigabitethernet 1/0/2
RTT(config-if-gi)# description "to PE2"
RTT(config-if-gi)# mtu 1522
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# ip address 10.101.0.2/30
RTT(config-if-gi)# ip ospf instance 1
RTT(config-if-gi)# ip ospf
RTT(config-if-gi)# exit
RTT(config)# interface loopback 1
RTT(config-loopback)# ip address 10.11.1.2/32
RTT(config-loopback)# ip ospf instance 1
RTT(config-loopback)# ip ospf
RTT(config-loopback)# exit
RTT(config)# mpls
RTT(config-mpls)# ldp
RTT(config-ldp)# router-id 10.11.1.2
RTT(config-ldp)# address-family ipv4
RTT(config-ldp-af-ipv4)# interface gigabitethernet 1/0/2
RTT(config-ldp-af-ipv4-if)# exit
RTT(config-ldp-af-ipv4)# exit
RTT(config-ldp)# enable
RTT(config-ldp)# exit
RTT(config-mpls)# forwarding interface gigabitethernet 1/0/2
RTT(config-mpls)# exit
RTT(config)# do com
RTT(config)# do conf

```

Проверим распространение маршрутной информации и сетевую доступность узлов:

```

PE1# sh bgp vpnv4 unicast all
Status codes: * - valid, > - best, i - internal, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

```

Codes	Route	Distinguisher	IP Prefix	Next hop	Metric	Label	LocPrf	Weight	Path
*>	65500:1		10.110.0.1/32	--	--	37	100	--	65501 i
*>	65500:1		10.111.0.1/32	--	--	35	100	--	65501 i
*>i	65500:1		10.113.0.1/32	10.10.1.2	--	43	100	0	?
*>i	65500:1		10.114.0.1/32	10.10.1.2	--	48	100	0	?

```

CE1# ping 10.113.0.1 source ip 10.110.0.1 detailed
PING 10.113.0.1 (10.113.0.1) from 10.110.0.1 : 56 bytes of data.
64 bytes from 10.113.0.1: icmp_seq=1 ttl=0 time=1.31 ms
64 bytes from 10.113.0.1: icmp_seq=2 ttl=0 time=1.14 ms
64 bytes from 10.113.0.1: icmp_seq=3 ttl=0 time=1.08 ms
64 bytes from 10.113.0.1: icmp_seq=4 ttl=0 time=1.06 ms
64 bytes from 10.113.0.1: icmp_seq=5 ttl=0 time=1.16 ms

```

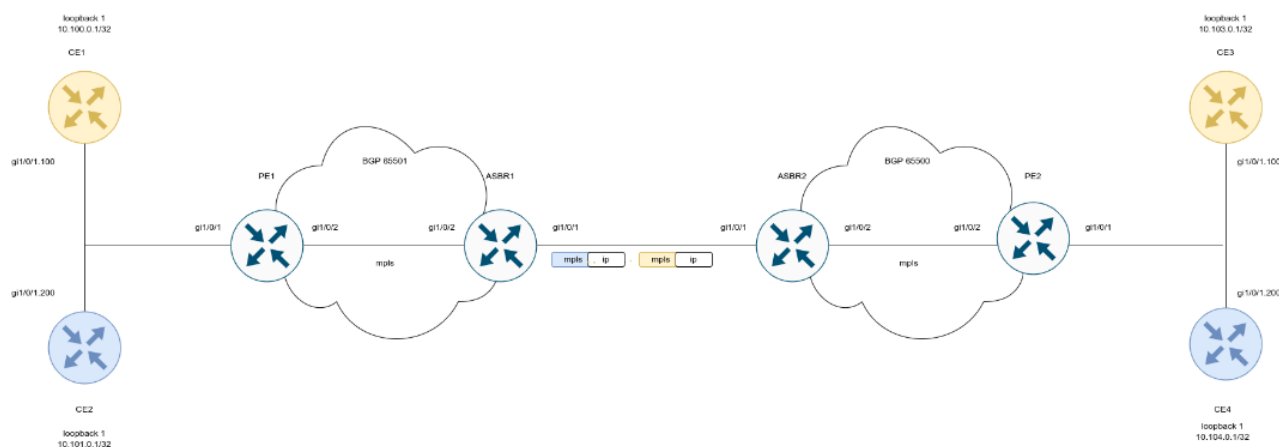
13.12. Inter-AS Option B

В отличие от Option A, между ASBR нет необходимости использовать VRF: при передаче трафика между ASBR будет навешиваться mpls-метка. Данная схема имеет лучшую масштабируемость.



В текущей реализации Option B поддерживается только для VPN-IPv4 маршрутов (AFI = 1, SAFI = 128).

13.12.1. L3VPN



Настроим CE:

CE1

```
RTT(config)# hostname CE1
RTT(config)#
RTT(config)# route-map BGP
RTT(config-route-map)# rule 1
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
RTT(config)# router bgp 65510
RTT(config-bgp)# neighbor 192.168.1.2
RTT(config-bgp-neighbor)# remote-as 65500
RTT(config-bgp-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af)# route-map BGP out
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# address-family ipv4 unicast
RTT(config-bgp-af)# network 10.100.0.1/32
RTT(config-bgp-af)# exit
RTT(config-bgp)# enable
RTT(config-bgp)# exit
RTT(config)# interface gigabitethernet 1/0/1.100
RTT(config-if-sub)# ip firewall disable
```

```
RTT(config-if-sub)# ip address 192.168.1.1/30
RTT(config-if-sub)# exit
RTT(config)# interface loopback 1
RTT(config-loopback)# ip address 10.100.0.1/32
RTT(config-loopback)# exit
RTT(config)# do com
RTT(config)# do conf
```

CE2

```
RTT(config)# hostname CE2
RTT(config)#
RTT(config)# route-map BGP
RTT(config-route-map)# rule 1
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
RTT(config)# router bgp 65511
RTT(config-bgp)# neighbor 192.168.2.2
RTT(config-bgp-neighbor)# remote-as 65500
RTT(config-bgp-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af)# route-map BGP out
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# address-family ipv4 unicast
RTT(config-bgp-af)# network 10.101.0.1/32
RTT(config-bgp-af)# exit
RTT(config-bgp)# enable
RTT(config-bgp)# exit
RTT(config)# interface gigabitethernet 1/0/1.100
RTT(config-if-sub)# ip firewall disable
RTT(config-if-sub)# ip address 192.168.2.1/30
RTT(config-if-sub)# exit
RTT(config)#
RTT(config)# interface loopback 1
RTT(config-loopback)# ip address 10.101.0.1/32
RTT(config-loopback)# exit
RTT(config)# do com
RTT(config)# do conf
```

CE3

```
RTT(config)# hostname CE3
RTT(config)#
RTT(config)# route-map BGP
RTT(config-route-map)# rule 1
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
RTT(config)# router bgp 65512
RTT(config-bgp)# neighbor 192.168.3.2
RTT(config-bgp-neighbor)# remote-as 65500
RTT(config-bgp-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af)# route-map BGP out
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
```



```
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# address-family ipv4 unicast
RTT(config-bgp-af)# network 10.103.0.1/32
RTT(config-bgp-af)# exit
RTT(config-bgp)# enable
RTT(config-bgp)# exit
RTT(config)# interface gigabitethernet 1/0/1.100
RTT(config-if-sub)# ip firewall disable
RTT(config-if-sub)# ip address 192.168.3.1/30
RTT(config-if-sub)# exit
RTT(config)#
RTT(config)# interface loopback 1
RTT(config-loopback)# ip address 10.103.0.1/32
RTT(config-loopback)# exit
RTT(config)# do com
RTT(config)# do conf
```

CE4

```
RTT(config)# hostname CE4
RTT(config)#
RTT(config)# route-map BGP
RTT(config-route-map)# rule 1
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
RTT(config)# router bgp 65513
RTT(config-bgp)# neighbor 192.168.4.2
RTT(config-bgp-neighbor)# remote-as 65500
RTT(config-bgp-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af)# route-map BGP out
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# address-family ipv4 unicast
RTT(config-bgp-af)# network 10.104.0.1/32
RTT(config-bgp-af)# exit
RTT(config-bgp)# enable
RTT(config-bgp)# exit
RTT(config)# interface gigabitethernet 1/0/1.100
RTT(config-if-sub)# ip firewall disable
RTT(config-if-sub)# ip address 192.168.4.1/30
RTT(config-if-sub)# exit
RTT(config)#
RTT(config)# interface loopback 1
RTT(config-loopback)# ip address 10.104.0.1/32
RTT(config-loopback)# exit
RTT(config)# do com
RTT(config)# do conf
```

Произведем настройку PE1 и PE2:

PE1

```
PE1(config)# hostname PE1
PE1(config)#
```

```
PE1(config)# ip vrf CE1
PE1(config-vrf)# ip protocols bgp max-routes 100
PE1(config-vrf)# rd 65501:1
PE1(config-vrf)# route-target export 65501:1
PE1(config-vrf)# route-target import 65501:1
PE1(config-vrf)# exit
PE1(config)# ip vrf CE2
PE1(config-vrf)# ip protocols bgp max-routes 100
PE1(config-vrf)# rd 65501:2
PE1(config-vrf)# route-target export 65501:2
PE1(config-vrf)# route-target import 65501:2
PE1(config-vrf)# exit
PE1(config)#
PE1(config)# system jumbo-frames
PE1(config)#
PE1(config)# route-map BGP_OUT
PE1(config-route-map)# rule 1
PE1(config-route-map-rule)# exit
PE1(config-route-map)# exit
PE1(config)# router bgp 65501
PE1(config-bgp)# neighbor 10.10.1.2
PE1(config-bgp-neighbor)# remote-as 65501
PE1(config-bgp-neighbor)# update-source 10.10.1.1
PE1(config-bgp-neighbor)# address-family vpnv4 unicast
PE1(config-bgp-neighbor-af)# send-community extended
PE1(config-bgp-neighbor-af)# enable
PE1(config-bgp-neighbor-af)# exit
PE1(config-bgp-neighbor)# enable
PE1(config-bgp-neighbor)# exit
PE1(config-bgp)# enable
PE1(config-bgp)# vrf CE1
PE1(config-bgp-vrf)# neighbor 192.168.1.1
PE1(config-bgp-vrf-neighbor)# remote-as 65510
PE1(config-bgp-vrf-neighbor)# address-family ipv4 unicast
PE1(config-bgp-neighbor-af-vrf)# route-map BGP_OUT out
PE1(config-bgp-neighbor-af-vrf)# enable
PE1(config-bgp-neighbor-af-vrf)# exit
PE1(config-bgp-vrf-neighbor)# enable
PE1(config-bgp-vrf-neighbor)# exit
PE1(config-bgp-vrf)# address-family ipv4 unicast
PE1(config-bgp-vrf-af)# redistribute bgp 65501 route-map BGP_OUT
PE1(config-bgp-vrf-af)# exit
PE1(config-bgp-vrf)# enable
PE1(config-bgp-vrf)# exit
PE1(config-bgp)# vrf CE2
PE1(config-bgp-vrf)# neighbor 192.168.2.1
PE1(config-bgp-vrf-neighbor)# remote-as 65511
PE1(config-bgp-vrf-neighbor)# address-family ipv4 unicast
PE1(config-bgp-neighbor-af-vrf)# route-map BGP_OUT out
PE1(config-bgp-neighbor-af-vrf)# enable
PE1(config-bgp-neighbor-af-vrf)# exit
PE1(config-bgp-vrf-neighbor)# enable
PE1(config-bgp-vrf-neighbor)# exit
PE1(config-bgp-vrf)# address-family ipv4 unicast
PE1(config-bgp-vrf-af)# redistribute bgp 65501 route-map BGP_OUT
PE1(config-bgp-vrf-af)# exit
PE1(config-bgp-vrf)# enable
PE1(config-bgp-vrf)# exit
PE1(config-bgp)# exit
```

```
PE1(config)#
PE1(config)# router ospf 1
PE1(config-ospf)# area 0.0.0.0
PE1(config-ospf-area)# enable
PE1(config-ospf-area)# exit
PE1(config-ospf)# enable
PE1(config-ospf)# exit
PE1(config)#
PE1(config)# interface gigabitethernet 1/0/1.100
PE1(config-if-sub)# ip vrf forwarding CE1
PE1(config-if-sub)# description "to CE1"
PE1(config-if-sub)# ip firewall disable
PE1(config-if-sub)# ip address 192.168.1.2/30
PE1(config-if-sub)# exit
PE1(config)# interface gigabitethernet 1/0/1.200
PE1(config-if-sub)# ip vrf forwarding CE2
PE1(config-if-sub)# description "to CE2"
PE1(config-if-sub)# ip firewall disable
PE1(config-if-sub)# ip address 192.168.2.2/30
PE1(config-if-sub)# exit
PE1(config)# interface gigabitethernet 1/0/2
PE1(config-if-gi)# description "to ASBR1"
PE1(config-if-gi)# mtu 1522
PE1(config-if-gi)# ip firewall disable
PE1(config-if-gi)# ip address 10.100.0.1/30
PE1(config-if-gi)# ip ospf instance 1
PE1(config-if-gi)# ip ospf
PE1(config-if-gi)# exit
PE1(config)# interface loopback 1
PE1(config-loopback)# ip address 10.10.1.1/32
PE1(config-loopback)# ip ospf instance 1
PE1(config-loopback)# ip ospf
PE1(config-loopback)# exit
PE1(config)# mpls
PE1(config-mpls)# ldp
PE1(config-ldp)# router-id 10.10.1.1
PE1(config-ldp)# address-family ipv4
PE1(config-ldp-af-ipv4)# interface gigabitethernet 1/0/2
PE1(config-ldp-af-ipv4-if)# exit
PE1(config-ldp-af-ipv4)# exit
PE1(config-ldp)# enable
PE1(config-ldp)# exit
PE1(config-mpls)# forwarding interface gigabitethernet 1/0/2
PE1(config-mpls)# exit
PE1(config)# do com
PE1(config)# do conf
```

PE2

```
PE2(config)# hostname PE2
PE2(config)#
PE2(config)# ip vrf CE3
PE2(config-vrf)# ip protocols bgp max-routes 100
PE2(config-vrf)# rd 65501:1
PE2(config-vrf)# route-target export 65501:1
PE2(config-vrf)# route-target import 65501:1
PE2(config-vrf)# exit
```

```
PE2(config)# ip vrf CE4
PE2(config-vrf)# ip protocols bgp max-routes 100
PE2(config-vrf)# rd 65501:2
PE2(config-vrf)# route-target export 65501:2
PE2(config-vrf)# route-target import 65501:2
PE2(config-vrf)# exit
PE2(config)#
PE2(config)# system jumbo-frames
PE2(config)#
PE2(config)# route-map BGP_OUT
PE2(config-route-map)# rule 1
PE2(config-route-map-rule)# exit
PE2(config-route-map)# exit
PE2(config)# router bgp 65500
PE2(config-bgp)# neighbor 10.11.1.2
PE2(config-bgp-neighbor)# remote-as 65500
PE2(config-bgp-neighbor)# update-source 10.11.1.1
PE2(config-bgp-neighbor)# address-family vpnv4 unicast
PE2(config-bgp-neighbor-af)# send-community extended
PE2(config-bgp-neighbor-af)# enable
PE2(config-bgp-neighbor-af)# exit
PE2(config-bgp-neighbor)# enable
PE2(config-bgp-neighbor)# exit
PE2(config-bgp)# enable
PE2(config-bgp)# vrf CE3
PE2(config-bgp-vrf)# neighbor 192.168.3.1
PE2(config-bgp-vrf-neighbor)# remote-as 65512
PE2(config-bgp-vrf-neighbor)# address-family ipv4 unicast
PE2(config-bgp-vrf-neighbor-af-vrf)# route-map BGP_OUT out
PE2(config-bgp-vrf-neighbor-af-vrf)# enable
PE2(config-bgp-vrf-neighbor-af-vrf)# exit
PE2(config-bgp-vrf-neighbor)# enable
PE2(config-bgp-vrf-neighbor)# exit
PE2(config-bgp-vrf)# address-family ipv4 unicast
PE2(config-bgp-vrf-af)# redistribute bgp 65500 route-map BGP_OUT
PE2(config-bgp-vrf-af)# exit
PE2(config-bgp-vrf)# enable
PE2(config-bgp-vrf)# exit
PE2(config-bgp)# vrf CE4
PE2(config-bgp-vrf)# neighbor 192.168.4.1
PE2(config-bgp-vrf-neighbor)# remote-as 65513
PE2(config-bgp-vrf-neighbor)# address-family ipv4 unicast
PE2(config-bgp-vrf-neighbor-af-vrf)# route-map BGP_OUT out
PE2(config-bgp-vrf-neighbor-af-vrf)# enable
PE2(config-bgp-vrf-neighbor-af-vrf)# exit
PE2(config-bgp-vrf-neighbor)# enable
PE2(config-bgp-vrf-neighbor)# exit
PE2(config-bgp-vrf)# address-family ipv4 unicast
PE2(config-bgp-vrf-af)# redistribute bgp 65500 route-map BGP_OUT
PE2(config-bgp-vrf-af)# exit
PE2(config-bgp-vrf)# enable
PE2(config-bgp-vrf)# exit
PE2(config-bgp)# exit
PE2(config)#
PE2(config)# router ospf 1
PE2(config-ospf)# router-id 10.11.1.1
PE2(config-ospf)# area 0.0.0.0
PE2(config-ospf-area)# enable
PE2(config-ospf-area)# exit
```

```
PE2(config-ospf)# enable
PE2(config-ospf)# exit
PE2(config)#
PE2(config)# interface gigabitethernet 1/0/1.100
PE2(config-if-sub)# ip vrf forwarding CE3
PE2(config-if-sub)# description "to CE3"
PE2(config-if-sub)# ip firewall disable
PE2(config-if-sub)# ip address 192.168.3.2/30
PE2(config-if-sub)# exit
PE2(config)# interface gigabitethernet 1/0/1.200
PE2(config-if-sub)# ip vrf forwarding CE4
PE2(config-if-sub)# description "CE4"
PE2(config-if-sub)# ip firewall disable
PE2(config-if-sub)# ip address 192.168.4.2/30
PE2(config-if-sub)# exit
PE2(config)# interface gigabitethernet 1/0/2
PE2(config-if-gi)# description "to ASBR2"
PE2(config-if-gi)# mtu 1522
PE2(config-if-gi)# ip firewall disable
PE2(config-if-gi)# ip address 10.102.0.1/30
PE2(config-if-gi)# ip ospf instance 1
PE2(config-if-gi)# ip ospf
PE2(config-if-gi)# exit
PE2(config)# interface loopback 1
PE2(config-loopback)# ip address 10.11.1.1/32
PE2(config-loopback)# ip ospf instance 1
PE2(config-loopback)# ip ospf
PE2(config-loopback)# exit
PE2(config)# mpls
PE2(config-mpls)# ldp
PE2(config-ldp)# router-id 10.11.1.1
PE2(config-ldp)# address-family ipv4
PE2(config-ldp-af-ipv4)# interface gigabitethernet 1/0/2
PE2(config-ldp-af-ipv4-if)# exit
PE2(config-ldp-af-ipv4)# exit
PE2(config-ldp)# enable
PE2(config-ldp)# exit
PE2(config-mpls)# forwarding interface gigabitethernet 1/0/2
PE2(config-mpls)# exit
PE2(config)# do com
PE2(config)# do conf
```

Настроим ASBR1 и ASBR2:

ASBR1

```
ASBR1(config)# hostname ASBR1
ASBR1(config)#
ASBR1(config)# system jumbo-frames
ASBR1(config)#
ASBR1(config)# route-map VPNv4
ASBR1(config-route-map)# rule 1
ASBR1(config-route-map-rule)# exit
ASBR1(config-route-map)# exit
ASBR1(config)# router bgp 65501
ASBR1(config-bgp)# router-id 10.10.1.2
```

```
ASBR1(config-bgp)# neighbor 10.10.1.1
ASBR1(config-bgp-neighbor)# remote-as 65501
ASBR1(config-bgp-neighbor)# update-source 10.10.1.2
ASBR1(config-bgp-neighbor)# address-family vpnv4 unicast
ASBR1(config-bgp-neighbor-af)# next-hop-self
ASBR1(config-bgp-neighbor-af)# send-community extended
ASBR1(config-bgp-neighbor-af)# enable
ASBR1(config-bgp-neighbor-af)# exit
ASBR1(config-bgp-neighbor)# enable
ASBR1(config-bgp-neighbor)# exit
ASBR1(config-bgp)# neighbor 10.101.0.1
ASBR1(config-bgp-neighbor)# remote-as 65500
ASBR1(config-bgp-neighbor)# address-family vpnv4 unicast
ASBR1(config-bgp-neighbor-af)# route-map VPNv4 out
ASBR1(config-bgp-neighbor-af)# send-community extended
ASBR1(config-bgp-neighbor-af)# enable
ASBR1(config-bgp-neighbor-af)# exit
ASBR1(config-bgp-neighbor)# enable
ASBR1(config-bgp-neighbor)# exit
ASBR1(config-bgp)# enable
ASBR1(config-bgp)# exit
ASBR1(config)#
ASBR1(config)# router ospf 1
ASBR1(config-ospf)# area 0.0.0.0
ASBR1(config-ospf-area)# enable
ASBR1(config-ospf-area)# exit
ASBR1(config-ospf)# enable
ASBR1(config-ospf)# exit
ASBR1(config)#
ASBR1(config)# interface gigabitethernet 1/0/1
ASBR1(config-if-gi)# description "to ASBR2"
ASBR1(config-if-gi)# ip firewall disable
ASBR1(config-if-gi)# ip address 10.101.0.2/30
ASBR1(config-if-gi)# exit
ASBR1(config)# interface gigabitethernet 1/0/2
ASBR1(config-if-gi)# description "to PE1"
ASBR1(config-if-gi)# mtu 1522
ASBR1(config-if-gi)# ip firewall disable
ASBR1(config-if-gi)# ip address 10.100.0.2/30
ASBR1(config-if-gi)# ip ospf instance 1
ASBR1(config-if-gi)# ip ospf
ASBR1(config-if-gi)# exit
ASBR1(config)# interface loopback 1
ASBR1(config-loopback)# ip address 10.10.1.2/32
ASBR1(config-loopback)# ip ospf instance 1
ASBR1(config-loopback)# ip ospf
ASBR1(config-loopback)# exit
ASBR1(config)# mpls
ASBR1(config-mpls)# ldp
ASBR1(config-ldp)# router-id 10.10.1.2
ASBR1(config-ldp)# address-family ipv4
ASBR1(config-ldp-af-ipv4)# interface gigabitethernet 1/0/2
ASBR1(config-ldp-af-ipv4-if)# exit
ASBR1(config-ldp-af-ipv4)# exit
ASBR1(config-ldp)# enable
ASBR1(config-ldp)# exit
ASBR1(config-mpls)# forwarding interface gigabitethernet 1/0/1
ASBR1(config-mpls)# forwarding interface gigabitethernet 1/0/2
ASBR1(config-mpls)# exit
```

```
ASBR1(config)# do com
ASBR1(config)# do conf
```

ASBR2

```
ASBR2(config)# hostname ASBR2
ASBR2(config)#
ASBR2(config)# system jumbo-frames
ASBR2(config)#
ASBR2(config)# route-map VPNv4
ASBR2(config-route-map)# rule 1
ASBR2(config-route-map-rule)# exit
ASBR2(config-route-map)# exit
ASBR2(config)# router bgp 65500
ASBR2(config-bgp)# router-id 10.11.1.2
ASBR2(config-bgp)# neighbor 10.101.0.2
ASBR2(config-bgp-neighbor)# remote-as 65501
ASBR2(config-bgp-neighbor)# address-family vpnv4 unicast
ASBR2(config-bgp-neighbor-af)# route-map VPNv4 out
ASBR2(config-bgp-neighbor-af)# send-community extended
ASBR2(config-bgp-neighbor-af)# enable
ASBR2(config-bgp-neighbor-af)# exit
ASBR2(config-bgp-neighbor)# enable
ASBR2(config-bgp-neighbor)# exit
ASBR2(config-bgp)# neighbor 10.11.1.1
ASBR2(config-bgp-neighbor)# remote-as 65500
ASBR2(config-bgp-neighbor)# update-source 10.11.1.2
ASBR2(config-bgp-neighbor)# address-family vpnv4 unicast
ASBR2(config-bgp-neighbor-af)# next-hop-self
ASBR2(config-bgp-neighbor-af)# send-community extended
ASBR2(config-bgp-neighbor-af)# enable
ASBR2(config-bgp-neighbor-af)# exit
ASBR2(config-bgp-neighbor)# enable
ASBR2(config-bgp-neighbor)# exit
ASBR2(config-bgp)# enable
ASBR2(config-bgp)# exit
ASBR2(config)#
ASBR2(config)# router ospf 1
ASBR2(config-ospf)# router-id 10.11.1.2
ASBR2(config-ospf)# area 0.0.0.0
ASBR2(config-ospf-area)# enable
ASBR2(config-ospf-area)# exit
ASBR2(config-ospf)# enable
ASBR2(config-ospf)# exit
ASBR2(config)#
ASBR2(config)# interface gigabitethernet 1/0/1
ASBR2(config-if-gi)# description "to ASBR1"
ASBR2(config-if-gi)# ip firewall disable
ASBR2(config-if-gi)# ip address 10.101.0.1/30
ASBR2(config-if-gi)# exit
ASBR2(config)# interface gigabitethernet 1/0/2
ASBR2(config-if-gi)# description "to PE2"
ASBR2(config-if-gi)# mtu 1522
ASBR2(config-if-gi)# ip firewall disable
ASBR2(config-if-gi)# ip address 10.102.0.2/30
ASBR2(config-if-gi)# ip ospf instance 1
```

```
ASBR2(config-if-gi)# ip ospf
ASBR2(config-if-gi)# exit
ASBR2(config)# interface loopback 1
ASBR2(config-loopback)# ip address 10.11.1.2/32
ASBR2(config-loopback)# ip ospf instance 1
ASBR2(config-loopback)# ip ospf
ASBR2(config-loopback)# exit
ASBR2(config)# mpls
ASBR2(config-mpls)# ldp
ASBR2(config-ldp)# router-id 10.11.1.2
ASBR2(config-ldp)# address-family ipv4
ASBR2(config-ldp-af-ipv4)# interface gigabitethernet 1/0/2
ASBR2(config-ldp-af-ipv4-if)# exit
ASBR2(config-ldp-af-ipv4)# exit
ASBR2(config-ldp)# enable
ASBR2(config-ldp)# exit
ASBR2(config-mpls)# forwarding interface gigabitethernet 1/0/1
ASBR2(config-mpls)# forwarding interface gigabitethernet 1/0/2
ASBR2(config-mpls)# exit
ASBR2(config)# do com
ASBR2(config)# do conf
```

После завершения настройки проверим распространение маршрутной информации и сетевую доступность узлов:

```
PE1# sh bgp vpnv4 unicast all
```

```
Status codes: * - valid, > - best, i - internal, S - stale
```

```
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Codes	Route	Distinguisher	IP Prefix	Next hop	Metric	Label	LocPrf	Weight	Path
*>i	65501:2		10.104.0.1/32	10.10.1.2	--	23	100	0	65500 65513 i
*>i	65501:1		10.103.0.1/32	10.10.1.2	--	19	100	0	65500 65512 i
*>	65501:2		10.101.0.1/32	--	--	29	100	--	65511 i
*>	65501:1		10.100.0.1/32	--	--	28	100	--	65510 i

```
ASBR1# sh bgp vpnv4 unicast all
```

```
Status codes: * - valid, > - best, i - internal, S - stale
```

```
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Codes	Route	Distinguisher	IP Prefix	Next hop	Metric	Label	LocPrf	Weight	Path
*>	65501:2		10.104.0.1/32	10.101.0.1	--	24	100	0	65500 65513 i
*>	65501:1		10.103.0.1/32	10.101.0.1	--	20	100	0	65500 65512 i
*>i	65501:2		10.101.0.1/32	10.10.1.1	--	29	100	0	65511 i
*>i	65501:1		10.100.0.1/32	10.10.1.1	--	28	100	0	65510 i


```
ASBR2# sh bgp vpnv4 unicast all
Status codes: * - valid, > - best, i - internal, S - stale
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Codes	Route	Distinguisher	IP Prefix	Next hop	Metric	Label	LocPrf	Weight	Path
*>i	65501:2		10.104.0.1/32	10.11.1.1	--	19	100	0	65513 i
*>i	65501:1		10.103.0.1/32	10.11.1.1	--	18	100	0	65512 i
*>	65501:2		10.101.0.1/32	10.101.0.2	--	30	100	0	65501 65511 i
*>	65501:1		10.100.0.1/32	10.101.0.2	--	31	100	0	65501 65510 i

```
PE2# sh bgp vpnv4 unicast all
Status codes: * - valid, > - best, i - internal, S - stale
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Codes	Route	Distinguisher	IP Prefix	Next hop	Metric	Label	LocPrf	Weight	Path
*>	65501:2		10.104.0.1/32	--	--	19	100	--	65513 i
*>	65501:1		10.103.0.1/32	--	--	18	100	--	65512 i
*>i	65501:2		10.101.0.1/32	10.11.1.2	--	29	100	0	65501 65511 i
*>i	65501:1		10.100.0.1/32	10.11.1.2	--	30	100	0	65501 65510 i

```
CE4# ping 10.104.0.1 source ip 10.101.0.1 detailed
PING 10.104.0.1 (10.104.0.1) from 10.101.0.1 : 56 bytes of data.
64 bytes from 10.104.0.1: icmp_seq=1 ttl=0 time=2.25 ms
64 bytes from 10.104.0.1: icmp_seq=2 ttl=0 time=2.08 ms
64 bytes from 10.104.0.1: icmp_seq=3 ttl=0 time=2.15 ms
64 bytes from 10.104.0.1: icmp_seq=4 ttl=0 time=2.12 ms
64 bytes from 10.104.0.1: icmp_seq=5 ttl=0 time=2.09 ms
```

```
CE1# ping 10.103.0.1 source ip 10.100.0.1 detailed
PING 10.103.0.1 (10.103.0.1) from 10.100.0.1 : 56 bytes of data.
64 bytes from 10.103.0.1: icmp_seq=1 ttl=0 time=2.22 ms
```

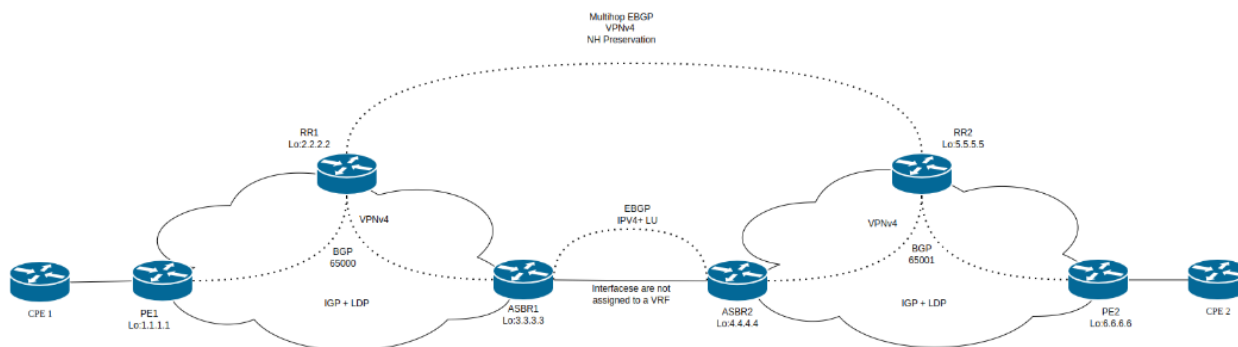
13.13. Inter-AS Option C

Inter-AS Option C является третьим сценарием для настройки связности различных автономных систем, описанным в RFC 4364. Данный сценарий является наиболее масштабируемым из описанных ранее, однако он имеет свои особенности, которые необходимо учитывать при построении сети.

В данной схеме ASBR-ы не хранят клиентские VPNv4-префиксы, а только распространяют маршрутную информацию и метки для PE-устройств в своей автономной системе.

Для распространения клиентских VPNv4-префиксов между различными автономными системами используется MP-EBGP-сессия между устройствами, выполняющими роль RR (route reflector), либо между PE-устройствами. В случае, если VPNv4-сессия настраивается между RR, то в передаваемых BGP update сообщениях не должен меняться атрибут next-hop.

В рамках EBGP-сессии между ASBR производится обмен маршрутной информацией о транспортных префиксах PE различных автономных систем. Эти маршруты отвечают за доступность next-hop для клиентских VPNv4-префиксов, передаваемых в рамках MP-EBGP-сессии между RR или PE. Данные префиксы также используются для установления MP-EBGP-сессии между устройствами, выполняющими роль RR либо роль PE в разных автономных системах.



Из плюсов данного решения можно отметить хорошую масштабируемость. ASBR-устройства не хранят данные клиентских префиксов, вся информация хранится на RR, что положительно сказывается на производительности.

Из недостатков можно отметить следующее:

- Безопасность. Передача транспортных префиксов PE из локальной AS во вне несет в себе потенциальные риски. Между AS должен быть установлен высокий уровень доверия.
- QoS. VPN-контексты отсутствуют на ASBR, соответственно нет возможности применить shaping/policing per VPN.

Для организации сквозных настроек QoS требуется согласование настроек на стыке ASBR↔ASBR.

13.13.1. L3VPN

Предварительная конфигурация:

- Внутри AS должен быть настроен IGP для распространения маршрутной информации для связности PE.
- Внутри AS должен быть настроен протокол LDP для распространения меток.
- На PE, к которым подключены абонентские CPE, должны быть настроены соответствующие VRF. Интерфейсы, к которым подключены CPE, должны быть помещены в соответствующий VRF.

Для настройки сервиса VPN приведем пример конфигурации устройств одной из AS (настройки в другой AS будут полностью зеркальные):

PE-1

```
RTT(config)# hostname PE1
RTT(config)#
RTT(config)# ip vrf vrf1
RTT(config-vrf)# rd 1.1.1.1:1
RTT(config-vrf)# route-target export 100:1
RTT(config-vrf)# route-target import 100:1
RTT(config-vrf)# exit
RTT(config)#
RTT(config)# router bgp 65000
```

```
RTT(config-bgp)# neighbor 2.2.2.2
RTT(config-bgp-neighbor)# remote-as 65000
RTT(config-bgp-neighbor)# update-source loopback 1
RTT(config-bgp-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af)# send-label
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# address-family vpnv4 unicast
RTT(config-bgp-neighbor-af)# next-hop-self
RTT(config-bgp-neighbor-af)# send-community extended
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# enable
RTT(config-bgp)# vrf vrf1
RTT(config-bgp-vrf)# address-family ipv4 unicast
RTT(config-bgp-vrf-af)# network 100.100.100.1/32
RTT(config-bgp-vrf-af)# exit
RTT(config-bgp-vrf)# exit
RTT(config-bgp)# exit
```

В примере конфигурации PE устройства префикс 100.100.100.1 является примером абонентской подсети.

RR-1

```
RTT(config)# hostname RR1
RTT(config)#
RTT(config)# route-map VPNv4_RM1
RTT(config-route-map)# rule 1
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
RTT(config)# router bgp 65000
RTT(config-bgp)# neighbor 3.3.3.3
RTT(config-bgp-neighbor)# remote-as 65000
RTT(config-bgp-neighbor)# route-reflector-client
RTT(config-bgp-neighbor)# update-source loopback 1
RTT(config-bgp-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af)# send-label
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# neighbor 1.1.1.1
RTT(config-bgp-neighbor)# remote-as 65000
RTT(config-bgp-neighbor)# route-reflector-client
RTT(config-bgp-neighbor)# update-source loopback 1
RTT(config-bgp-neighbor)# address-family ipv4 unicast
RTT(config-bgp-neighbor-af)# send-label
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
RTT(config-bgp-neighbor)# address-family vpnv4 unicast
RTT(config-bgp-neighbor-af)# send-community extended
RTT(config-bgp-neighbor-af)# enable
RTT(config-bgp-neighbor-af)# exit
```

```
RTT(config-bgp-neighbor) # enable
RTT(config-bgp-neighbor) # exit
RTT(config-bgp) # neighbor 5.5.5.5
RTT(config-bgp-neighbor) # remote-as 65001
RTT(config-bgp-neighbor) # ebgp-multihop 10
RTT(config-bgp-neighbor) # update-source loopback 1
RTT(config-bgp-neighbor) # address-family vpnv4 unicast
RTT(config-bgp-neighbor-af) # route-map VPNv4_RM1 out
RTT(config-bgp-neighbor-af) # next-hop-unchanged
RTT(config-bgp-neighbor-af) # send-community extended
RTT(config-bgp-neighbor-af) # enable
RTT(config-bgp-neighbor-af) # exit
RTT(config-bgp-neighbor) # enable
RTT(config-bgp-neighbor) # exit
RTT(config-bgp) # enable
RTT(config-bgp) # exit
```

ASBR-1

```
RTT(config) # hostname ASBR1
RTT(config) #
RTT(config) # route-map RM1
RTT(config-route-map) # rule 1
RTT(config-route-map-rule) # exit
RTT(config-route-map) # exit
RTT(config) # router bgp 65000
RTT(config-bgp) # neighbor 2.2.2.2
RTT(config-bgp-neighbor) # remote-as 65000
RTT(config-bgp-neighbor) # update-source loopback 1
RTT(config-bgp-neighbor) # address-family ipv4 unicast
RTT(config-bgp-neighbor-af) # next-hop-self
RTT(config-bgp-neighbor-af) # send-label
RTT(config-bgp-neighbor-af) # enable
RTT(config-bgp-neighbor-af) # exit
RTT(config-bgp-neighbor) # enable
RTT(config-bgp-neighbor) # exit
RTT(config-bgp) # neighbor 192.168.100.1
RTT(config-bgp-neighbor) # remote-as 65001
RTT(config-bgp-neighbor) # address-family ipv4 unicast
RTT(config-bgp-neighbor-af) # route-map RM1 out
RTT(config-bgp-neighbor-af) # send-label
RTT(config-bgp-neighbor-af) # enable
RTT(config-bgp-neighbor-af) # exit
RTT(config-bgp-neighbor) # enable
RTT(config-bgp-neighbor) # exit
RTT(config-bgp) # address-family ipv4 unicast
RTT(config-bgp-af) # network 1.1.1.1/32
RTT(config-bgp-af) # network 2.2.2.2/32
RTT(config-bgp-af) # network 3.3.3.3/32
RTT(config-bgp-af) # exit
RTT(config-bgp) # enable
RTT(config-bgp) # exit
```

13.14. MPLS over GRE

В этом разделе приведен пример настройки VPN сервисов, построенных через GRE-туннель.

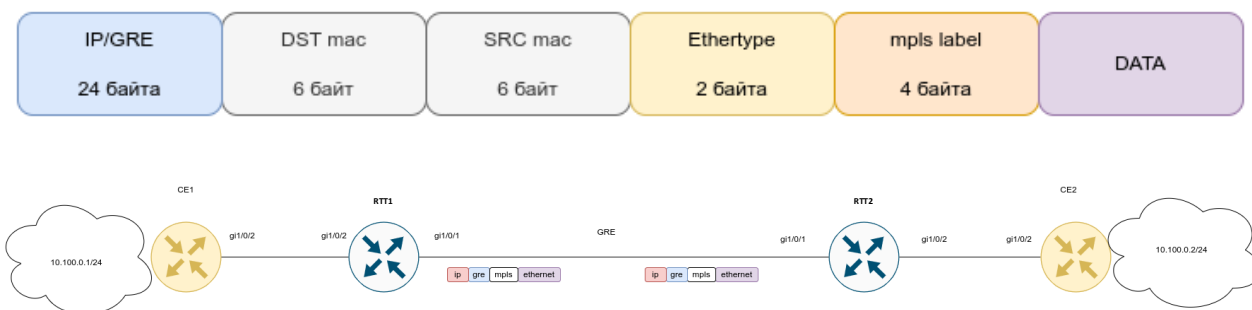
13.14.1. L2VPN

В качестве сервиса l2vpn произведем настройку EoMPLS over GRE. Также возможно построение VPLS over GRE (BGP или LDP signaling).

При настройке MTU на туннеле необходимо учитывать следующее:

- По крайней мере одна mpls-метка будет присутствовать при передаче через туннель. В учет стоит включать все метки в стеке, например, `explicit null` или `entropy label`;
- Необходимо учитывать vlan, q-in-q заголовки (если они имеются);
- При превышении MTU исходящего интерфейса пакет будет отброшен (если не включена безусловная фрагментация GRE-трафика);
- Control world не поддерживается;
- DF-бит будет выставлен в единицу.

Ниже представлена примерная структура пакета:



Настройки CE1 и CE2:

CE1

```
hostname CE1

interface gigabitethernet 1/0/2
 ip firewall disable
 ip address 10.100.0.1/24
exit
```

CE2

```
hostname CE2

interface gigabitethernet 1/0/2
 ip firewall disable
 ip address 10.100.0.2/24
exit
```

Конфигурация RTT1 и RTT2:

```
RTT1(config)# hostname RTT1
RTT1(config)#
RTT1(config)# system cpu load-balance mpls passenger ip
RTT1(config)# system cpu load-balance mpls passenger ipoe-pw-without-cw
RTT1(config)# security zone trusted
RTT1(config-zone)# exit
RTT1(config)# security zone untrusted
RTT1(config-zone)# exit
RTT1(config)#
RTT1(config)# router ospf 1
RTT1(config-ospf)# area 0.0.0.0
RTT1(config-ospf-area)# enable
RTT1(config-ospf-area)# exit
RTT1(config-ospf)# enable
RTT1(config-ospf)# exit
RTT1(config)#
RTT1(config)# interface gigabitethernet 1/0/1
RTT1(config-if-gi)# security-zone untrusted
RTT1(config-if-gi)# ip address 192.0.2.1/30
RTT1(config-if-gi)# exit
RTT1(config)# interface gigabitethernet 1/0/2
RTT1(config-if-gi)# description "From CE1"
RTT1(config-if-gi)# mode switchport
RTT1(config-if-gi)# exit
RTT1(config)# interface loopback 1
RTT1(config-loopback)# ip address 10.100.0.1/32
RTT1(config-loopback)# ip ospf instance 1
RTT1(config-loopback)# ip ospf
RTT1(config-loopback)# exit
RTT1(config)# tunnel gre 1
RTT1(config-gre)# key 60
RTT1(config-gre)# ttl 64
RTT1(config-gre)# mtu 1458
RTT1(config-gre)# ip firewall disable
RTT1(config-gre)# local address 192.0.2.1
RTT1(config-gre)# remote address 192.0.2.2
RTT1(config-gre)# ip address 10.0.0.1/30
RTT1(config-gre)# ip ospf instance 1
RTT1(config-gre)# ip ospf network point-to-point
RTT1(config-gre)# ip ospf
RTT1(config-gre)# enable
RTT1(config-gre)# exit
RTT1(config)#
RTT1(config)# mpls
RTT1(config-mpls)# ldp
RTT1(config-ldp)# router-id 10.100.0.1
RTT1(config-ldp)# address-family ipv4
RTT1(config-ldp-af-ipv4)# interface gre 1
RTT1(config-ldp-af-ipv4-if)# exit
RTT1(config-ldp-af-ipv4)# exit
RTT1(config-ldp)# enable
RTT1(config-ldp)# exit
RTT1(config-mpls)# l2vpn
RTT1(config-l2vpn)# pw-class VPWS
RTT1(config-l2vpn-pw-class)# exit
RTT1(config-l2vpn)# p2p EoMPLS
```

```
RTT1(config-l2vpn-p2p)# interface gigabitethernet 1/0/2
RTT1(config-l2vpn-p2p)# pw 100 10.100.0.2
RTT1(config-l2vpn-pw)# pw-class VPWS
RTT1(config-l2vpn-pw)# enable
RTT1(config-l2vpn-pw)# exit
RTT1(config-l2vpn-p2p)# enable
RTT1(config-l2vpn-p2p)# exit
RTT1(config-l2vpn)# exit
RTT1(config-mps)# forwarding interface gre 1
RTT1(config-mps)# exit
RTT1(config)# security zone-pair untrusted self
RTT1(config-zone-pair)# rule 1
RTT1(config-zone-pair-rule)# action permit
RTT1(config-zone-pair-rule)# match protocol gre
RTT1(config-zone-pair-rule)# enable
RTT1(config-zone-pair-rule)# exit
RTT1(config-zone-pair)# exit
RTT1(config)# do com
RTT1(config)# do conf
```

RTT2

```
RTT2(config)# hostname RTT2
RTT2(config)#
RTT2(config)# system cpu load-balance mpls passenger ip
RTT2(config)# system cpu load-balance mpls passenger ipoe-pw-without-cw
RTT2(config)# security zone trusted
RTT2(config-zone)# exit
RTT2(config)# security zone untrusted
RTT2(config-zone)# exit
RTT2(config)#
RTT2(config)# router ospf 1
RTT2(config-ospf)# area 0.0.0.0
RTT2(config-ospf-area)# enable
RTT2(config-ospf-area)# exit
RTT2(config-ospf)# enable
RTT2(config-ospf)# exit
RTT2(config)#
RTT2(config)# interface gigabitethernet 1/0/1
RTT2(config-if-gi)# security-zone untrusted
RTT2(config-if-gi)# ip address 192.0.2.2/30
RTT2(config-if-gi)# exit
RTT2(config)# interface gigabitethernet 1/0/2
RTT2(config-if-gi)# description "From CE2"
RTT2(config-if-gi)# mode switchport
RTT2(config-if-gi)# exit
RTT2(config)# interface loopback 1
RTT2(config-loopback)# ip address 10.100.0.2/32
RTT2(config-loopback)# ip ospf instance 1
RTT2(config-loopback)# ip ospf
RTT2(config-loopback)# exit
RTT2(config)# tunnel gre 1
RTT2(config-gre)# key 60
RTT2(config-gre)# ttl 64
RTT2(config-gre)# mtu 1458
RTT2(config-gre)# ip firewall disable
RTT2(config-gre)# local address 192.0.2.2
```

```
RTT2(config-gre)# remote address 192.0.2.1
RTT2(config-gre)# ip address 10.0.0.2/30
RTT2(config-gre)# ip ospf instance 1
RTT2(config-gre)# ip ospf network point-to-point
RTT2(config-gre)# ip ospf
RTT2(config-gre)# enable
RTT2(config-gre)# exit
RTT2(config)#
RTT2(config)# mpls
RTT2(config-mpls)# ldp
RTT2(config-ldp)# router-id 10.100.0.2
RTT2(config-ldp)# address-family ipv4
RTT2(config-ldp-af-ipv4)# interface gre 1
RTT2(config-ldp-af-ipv4-if)# exit
RTT2(config-ldp-af-ipv4)# exit
RTT2(config-ldp)# enable
RTT2(config-ldp)# exit
RTT2(config-mpls)# l2vpn
RTT2(config-l2vpn)# pw-class VPWS
RTT2(config-l2vpn-pw-class)# exit
RTT2(config-l2vpn)# p2p EoMPLS
RTT2(config-l2vpn-p2p)# interface gigabitethernet 1/0/2
RTT2(config-l2vpn-p2p)# pw 100 10.100.0.1
RTT2(config-l2vpn-pw)# pw-class VPWS
RTT2(config-l2vpn-pw)# enable
RTT2(config-l2vpn-pw)# exit
RTT2(config-l2vpn-p2p)# enable
RTT2(config-l2vpn-p2p)# exit
RTT2(config-l2vpn)# exit
RTT2(config-mpls)# forwarding interface gre 1
RTT2(config-mpls)# exit
RTT2(config)# security zone-pair untrusted self
RTT2(config-zone-pair)# rule 1
RTT2(config-zone-pair-rule)# action deny
RTT2(config-zone-pair-rule)# match protocol gre
RTT2(config-zone-pair-rule)# enable
RTT2(config-zone-pair-rule)# exit
RTT2(config-zone-pair)# exit
RTT2(config)# do com
RTT2(config)# do conf
```

Проверим состояние сервиса и доступность узлов:

* Конфигурация туннеля*

```
RTT2# sh tunnels configuration gre 1
State: Enabled
Description: --
Mode: ip
Bridge group: --
VRF: --
Local address: 192.0.2.2
Remote address: 192.0.2.1
Calculates checksums for outgoing GRE packets: No
Requires that all input GRE packets were checksum: No
key: 60
TTL: 64
DSCP: Inherit
MTU: 1458
```



```

Path MTU discovery:           Enabled
Don't fragment bit suppression: Disabled
Security zone:               --
Multipoint mode:             Disabled
Keepalive:
    State:                    Disabled
    Timeout:                  10
    Retries:                   6
    Destination address:      --

```

Статус сервиса и выделенные метки

```

sh mpls l2vpn p2p
P2P: EoMPLS
    gigabitethernet 1/0/2:
        MTU:      1500
        Status: Up
    PW ID 100, Neighbor 10.100.0.1:
        MTU:      1500
        Status TLV: Enable
        Last change: 00:14:27
        Status:    Up

```

RTT2# sh mpls forwarding-table

Local	Outgoing Prefix	Outgoing
Next Hop		
label	label or tunnel ID	Interface
17	imp-null 10.100.0.1/32	gre 1
10.0.0.1		
16	16 PW ID 100	--
10.100.0.1		

```

*Доступность*CE1# ping 10.100.0.2 detailed
PING 10.100.0.2 (10.100.0.2) 56 bytes of data.
64 bytes from 10.100.0.2: icmp_seq=1 ttl=0 time=1.38 ms
64 bytes from 10.100.0.2: icmp_seq=2 ttl=0 time=1.22 ms
64 bytes from 10.100.0.2: icmp_seq=3 ttl=0 time=1.33 ms
64 bytes from 10.100.0.2: icmp_seq=4 ttl=0 time=1.26 ms
64 bytes from 10.100.0.2: icmp_seq=5 ttl=0 time=1.17 ms

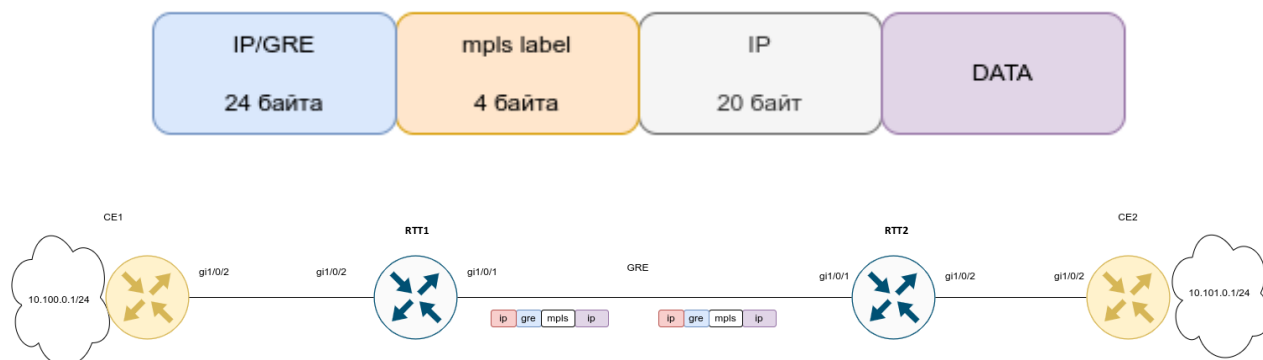
```

13.14.2. L3VPN

При настройке MTU на туннеле необходимо учитывать следующее:

- По крайней мере одна mpls-метка будет присутствовать при передаче через туннель. В учет стоит включать все метки в стеке, например, `explicit null` и/или `entropy label`;
- При превышении MTU исходящего интерфейса пакет будет отброшен (если не включена безусловная фрагментация GRE-трафика);
- Control world не поддержан;
- DF-бит будет выставлен в единицу.

Ниже представлена примерная структура пакета:



Настройки CE1 и CE2:

CE1

```
CE1(config)# hostname CE1
CE1(config)#
CE1(config)# route-map BGP_OUT
CE1(config-route-map)# rule 1
CE1(config-route-map-rule)# exit
CE1(config-route-map)# exit
CE1(config)# router bgp 65501
CE1(config-bgp)# neighbor 10.10.0.2
CE1(config-bgp-neighbor)# remote-as 65500
CE1(config-bgp-neighbor)# address-family ipv4 unicast
CE1(config-bgp-neighbor-af)# route-map BGP_OUT out
CE1(config-bgp-neighbor-af)# enable
CE1(config-bgp-neighbor-af)# exit
CE1(config-bgp-neighbor)# enable
CE1(config-bgp-neighbor)# exit
CE1(config-bgp)# address-family ipv4 unicast
CE1(config-bgp-af)# network 10.100.0.0/24
CE1(config-bgp-af)# exit
CE1(config-bgp)# enable
CE1(config-bgp)# exit
CE1(config)#
CE1(config)#
CE1(config)# interface gigabitethernet 1/0/2
CE1(config-if-gi)# description "to RTT1"
CE1(config-if-gi)# ip firewall disable
CE1(config-if-gi)# ip address 10.10.0.1/30
CE1(config-if-gi)# exit
CE1(config)# interface loopback 1
CE1(config-loopback)# ip address 10.100.0.1/24
CE1(config-loopback)# exit
```

CE2

```
CE2(config)# hostname CE2
CE2(config)#
CE2(config)# route-map BGP_OUT
CE2(config-route-map)# rule 1
CE2(config-route-map-rule)# exit
CE2(config-route-map)# exit
```

```
CE2(config)# router bgp 65502
CE2(config-bgp)# neighbor 10.10.0.5
CE2(config-bgp-neighbor)# remote-as 65500
CE2(config-bgp-neighbor)# address-family ipv4 unicast
CE2(config-bgp-neighbor-af)# route-map BGP_OUT out
CE2(config-bgp-neighbor-af)# enable
CE2(config-bgp-neighbor-af)# exit
CE2(config-bgp-neighbor)# enable
CE2(config-bgp-neighbor)# exit
CE2(config-bgp)# address-family ipv4 unicast
CE2(config-bgp-af)# network 10.101.0.0/24
CE2(config-bgp-af)# exit
CE2(config-bgp)# enable
CE2(config-bgp)# exit
CE2(config)#
CE2(config)#
CE2(config)# interface gigabitethernet 1/0/2
CE2(config-if-gi)# description "to RTT2"
CE2(config-if-gi)# ip firewall disable
CE2(config-if-gi)# ip address 10.10.0.6/30
CE2(config-if-gi)# exit
CE2(config)# interface loopback 1
CE2(config-loopback)# ip address 10.101.0.1/24
CE2(config-loopback)# exit
```

Конфигурация RTT1 и RTT2:

RTT1

```
RTT1(config)# hostname RTT1
RTT1(config)#
RTT1(config)# ip vrf l3vpn_service
RTT1(config-vrf)# ip protocols bgp max-routes 100
RTT1(config-vrf)# rd 65500:1
RTT1(config-vrf)# route-target export 65500:1
RTT1(config-vrf)# route-target import 65500:1
RTT1(config-vrf)# exit
RTT1(config)#
RTT1(config)#
RTT1(config)# system cpu load-balance mpls passenger ip
RTT1(config)# security zone untrusted
RTT1(config-zone)# exit
RTT1(config)# security zone trusted
RTT1(config-zone)# exit
RTT1(config)#
RTT1(config)# route-map BGP_OUT
RTT1(config-route-map)# rule 1
RTT1(config-route-map-rule)# exit
RTT1(config-route-map)# exit
RTT1(config)# router bgp 65500
RTT1(config-bgp)# router-id 10.12.0.1
RTT1(config-bgp)# neighbor 10.12.0.2
RTT1(config-bgp-neighbor)# remote-as 65500
RTT1(config-bgp-neighbor)# update-source 10.12.0.1
RTT1(config-bgp-neighbor)# address-family vpnv4 unicast
RTT1(config-bgp-neighbor-af)# send-community extended
```

```
RTT1(config-bgp-neighbor-af)# enable
RTT1(config-bgp-neighbor-af)# exit
RTT1(config-bgp-neighbor)# enable
RTT1(config-bgp-neighbor)# exit
RTT1(config-bgp)# enable
RTT1(config-bgp)# vrf l3vpn_service
RTT1(config-bgp-vrf)# neighbor 10.10.0.1
RTT1(config-bgp-vrf-neighbor)# remote-as 65501
RTT1(config-bgp-vrf-neighbor)# address-family ipv4 unicast
RTT1(config-bgp-neighbor-af-vrf)# route-map BGP_OUT out
RTT1(config-bgp-neighbor-af-vrf)# enable
RTT1(config-bgp-neighbor-af-vrf)# exit
RTT1(config-bgp-vrf-neighbor)# enable
RTT1(config-bgp-vrf-neighbor)# exit
RTT1(config-bgp-vrf)# address-family ipv4 unicast
RTT1(config-bgp-vrf-af)# redistribute bgp 65500 route-map BGP_OUT
RTT1(config-bgp-vrf-af)# exit
RTT1(config-bgp-vrf)# enable
RTT1(config-bgp-vrf)# exit
RTT1(config-bgp)# exit
RTT1(config)#
RTT1(config)# router ospf 1
RTT1(config-ospf)# router-id 10.12.0.1
RTT1(config-ospf)# area 0.0.0.0
RTT1(config-ospf-area)# enable
RTT1(config-ospf-area)# exit
RTT1(config-ospf)# enable
RTT1(config-ospf)# exit
RTT1(config)#
RTT1(config)# interface gigabitethernet 1/0/1
RTT1(config-if-gi)# security-zone untrusted
RTT1(config-if-gi)# ip address 192.0.2.1/30
RTT1(config-if-gi)# exit
RTT1(config)# interface gigabitethernet 1/0/2
RTT1(config-if-gi)# ip vrf forwarding l3vpn_service
RTT1(config-if-gi)# description "from CE1"
RTT1(config-if-gi)# ip firewall disable
RTT1(config-if-gi)# ip address 10.10.0.2/30
RTT1(config-if-gi)# exit
RTT1(config)# interface loopback 1
RTT1(config-loopback)# ip address 10.12.0.1/32
RTT1(config-loopback)# ip ospf instance 1
RTT1(config-loopback)# ip ospf
RTT1(config-loopback)# exit
RTT1(config)# tunnel gre 1
RTT1(config-gre)# key 60
RTT1(config-gre)# ttl 64
RTT1(config-gre)# mtu 1472
RTT1(config-gre)# ip firewall disable
RTT1(config-gre)# local address 192.0.2.1
RTT1(config-gre)# remote address 192.0.2.2
RTT1(config-gre)# ip address 10.11.0.1/30
RTT1(config-gre)# ip ospf instance 1
RTT1(config-gre)# ip ospf
RTT1(config-gre)# enable
RTT1(config-gre)# exit
RTT1(config)#
RTT1(config)# mpls
RTT1(config-mpls)# ldp
```

```
RTT1(config-ldp)#      router-id 10.12.0.1
RTT1(config-ldp)#      address-family ipv4
RTT1(config-ldp-af-ipv4)#      interface gre 1
RTT1(config-ldp-af-ipv4-if)#      exit
RTT1(config-ldp-af-ipv4)#      exit
RTT1(config-ldp)#      enable
RTT1(config-ldp)#      exit
RTT1(config-mpls)#      forwarding interface gre 1
RTT1(config-mpls)#      exit
RTT1(config)# security zone-pair untrusted self
RTT1(config-zone-pair)#      rule 1
RTT1(config-zone-pair-rule)#      action permit
RTT1(config-zone-pair-rule)#      match protocol gre
RTT1(config-zone-pair-rule)#      enable
RTT1(config-zone-pair-rule)#      exit
RTT1(config-zone-pair)#      exit
RTT2(config)# hostname RTT2
RTT2(config)#
RTT2(config)# ip vrf l3vpn_service
RTT2(config-vrf)#      ip protocols bgp max-routes 100
RTT2(config-vrf)#      rd 65500:1
RTT2(config-vrf)#      route-target export 65500:1
RTT2(config-vrf)#      route-target import 65500:1
RTT2(config-vrf)#      exit
RTT2(config)#
RTT2(config)#
RTT2(config)# system cpu load-balance mpls passenger ip
RTT2(config)# security zone untrusted
RTT2(config-zone)#      exit
RTT2(config)# security zone trusted
RTT2(config-zone)#      exit
RTT2(config)#
RTT2(config)# route-map BGP_OUT
RTT2(config-route-map)#      rule 1
RTT2(config-route-map-rule)#      exit
RTT2(config-route-map)#      exit
RTT2(config)# router bgp 65500
RTT2(config-bgp)#      router-id 10.12.0.2
RTT2(config-bgp)#      neighbor 10.12.0.1
RTT2(config-bgp-neighbor)#      remote-as 65500
RTT2(config-bgp-neighbor)#      update-source 10.12.0.2
RTT2(config-bgp-neighbor)#      address-family vpnv4 unicast
RTT2(config-bgp-neighbor-af)#      send-community extended
RTT2(config-bgp-neighbor-af)#      enable
RTT2(config-bgp-neighbor-af)#      exit
RTT2(config-bgp-neighbor)#      enable
RTT2(config-bgp-neighbor)#      exit
RTT2(config-bgp)#      enable
RTT2(config-bgp)#      vrf l3vpn_service
RTT2(config-bgp-vrf)#      neighbor 10.10.0.6
RTT2(config-bgp-vrf-neighbor)#      remote-as 65502
RTT2(config-bgp-vrf-neighbor)#      address-family ipv4 unicast
RTT2(config-bgp-neighbor-af-vrf)#      route-map BGP_OUT out
RTT2(config-bgp-neighbor-af-vrf)#      enable
RTT2(config-bgp-neighbor-af-vrf)#      exit
RTT2(config-bgp-vrf-neighbor)#      enable
RTT2(config-bgp-vrf-neighbor)#      exit
RTT2(config-bgp-vrf)#      address-family ipv4 unicast
```

```
RTT2(config-bgp-vrf-af)# redistribute bgp 65500 route-map BGP_OUT
RTT2(config-bgp-vrf-af)# exit
RTT2(config-bgp-vrf)# enable
RTT2(config-bgp-vrf)# exit
RTT2(config-bgp)# exit
RTT2(config)#
RTT2(config)# router ospf 1
RTT2(config-ospf)# router-id 10.12.0.2
RTT2(config-ospf)# area 0.0.0.0
RTT2(config-ospf-area)# enable
RTT2(config-ospf-area)# exit
RTT2(config-ospf)# enable
RTT2(config-ospf)# exit
RTT2(config)#
RTT2(config)# interface gigabitethernet 1/0/1
RTT2(config-if-gi)# security-zone untrusted
RTT2(config-if-gi)# ip address 192.0.2.2/30
RTT2(config-if-gi)# exit
RTT2(config)# interface gigabitethernet 1/0/2
RTT2(config-if-gi)# ip vrf forwarding l3vpn_service
RTT2(config-if-gi)# description "from CE2"
RTT2(config-if-gi)# ip firewall disable
RTT2(config-if-gi)# ip address 10.10.0.5/30
RTT2(config-if-gi)# exit
RTT2(config)# interface loopback 1
RTT2(config-loopback)# ip address 10.12.0.2/32
RTT2(config-loopback)# ip ospf instance 1
RTT2(config-loopback)# ip ospf
RTT2(config-loopback)# exit
RTT2(config)# tunnel gre 1
RTT2(config-gre)# key 60
RTT2(config-gre)# ttl 64
RTT2(config-gre)# mtu 1472
RTT2(config-gre)# ip firewall disable
RTT2(config-gre)# local address 192.0.2.2
RTT2(config-gre)# remote address 192.0.2.1
RTT2(config-gre)# ip address 10.11.0.2/30
RTT2(config-gre)# ip ospf instance 1
RTT2(config-gre)# ip ospf
RTT2(config-gre)# enable
RTT2(config-gre)# exit
RTT2(config)#
RTT2(config)# mpls
RTT2(config-mpls)# ldp
RTT2(config-ldp)# router-id 10.12.0.2
RTT2(config-ldp)# address-family ipv4
RTT2(config-ldp-af-ipv4)# interface gre 1
RTT2(config-ldp-af-ipv4-if)# exit
RTT2(config-ldp-af-ipv4)# exit
RTT2(config-ldp)# enable
RTT2(config-ldp)# exit
RTT2(config-mpls)# forwarding interface gre 1
RTT2(config-mpls)# exit
RTT2(config)# security zone-pair untrusted self
RTT2(config-zone-pair)# rule 1
RTT2(config-zone-pair-rule)# action permit
RTT2(config-zone-pair-rule)# match protocol gre
RTT2(config-zone-pair-rule)# enable
RTT2(config-zone-pair-rule)# exit
```

```
RTT2(config-zone-pair)# exit
```

После завершения настройки проверим статус сервиса и доступность узлов в сети:

Конфигурация туннеля GRE

```
RTT2# sh tunnels configuration
```

Tunnel	State	Description
gre 1	Enabled	--

```
RTT2# sh tunnels configuration gre 1
```

State:	Enabled
Description:	--
Mode:	ip
Bridge group:	--
VRF:	--
Local address:	192.0.2.2
Remote address:	192.0.2.1
Calculates checksums for outgoing GRE packets:	No
Requires that all input GRE packets were checksum:	No
key:	60
TTL:	64
DSCP:	Inherit
MTU:	1472
Path MTU discovery:	Enabled
Don't fragment bit suppression:	Disabled
Security zone:	--
Multipoint mode:	Disabled
Keepalive:	
State:	Disabled
Timeout:	10
Retries:	6
Destination address:	--

Наличие vpnv4-маршрутов

```
SR2# sh bgp vpnv4 unicast all
```

Status codes: * - valid, > - best, i - internal, S - stale

Origin codes: i - IGP, e - EGP, ? - incomplete

Codes	Route	Distinguisher	IP Prefix	Next hop	Metric	Label	LocPrf	Weight	Path
*>	65500:1		10.101.0.0/24	--	--	34	100	--	65502 i
*>i	65500:1		10.100.0.0/24	10.12.0.1	--	16	100	0	65501 i

Состояние протокола LDP

ESR2# sh mpls ldp neighbor

Peer LDP ID: 10.12.0.1; Local LDP ID 10.12.0.2

State: Operational

TCP connection: 10.12.0.1:646 - 10.12.0.2:46444

Messages sent/received: 60/60

Uptime: 00:53:59

LDP discovery sources:

gre 1

ESR2# sh mpls forwarding-table

Local	Outgoing	Prefix	Outgoing	Next Hop
label	label	or tunnel ID	Interface	
35	imp-null	10.12.0.1/32	gre 1	10.11.0.1

Доступность узлов в сети

CE2# ping 10.100.0.1 source ip 10.101.0.1 detailed

PING 10.100.0.1 (10.100.0.1) from 10.101.0.1 : 56 bytes of data.

64 bytes from 10.100.0.1: icmp_seq=1 ttl=0 time=1.32 ms

64 bytes from 10.100.0.1: icmp_seq=2 ttl=0 time=1.12 ms

64 bytes from 10.100.0.1: icmp_seq=3 ttl=0 time=1.14 ms

64 bytes from 10.100.0.1: icmp_seq=4 ttl=0 time=1.09 ms

64 bytes from 10.100.0.1: icmp_seq=5 ttl=0 time=1.15 ms

14. УПРАВЛЕНИЕ БЕЗОПАСНОСТЬЮ

14.1. Настройка AAA

AAA (Authentication, Authorization, Accounting) – используется для описания процесса предоставления доступа и контроля над ним.

- Authentication (аутентификация) – сопоставление персоны (запроса) существующей учётной записи в системе безопасности. Осуществляется по логину, паролю.
- Authorization (авторизация, проверка полномочий, проверка уровня доступа) – сопоставление учётной записи в системе и определённых полномочий.
- Accounting (учёт) – слежение за подключением пользователя или внесённым им изменениям.

14.1.1. Алгоритм настройки локальной аутентификации

Шаг	Описание	Команда	Ключи
1	Задать список методов аутентификации по умолчанию (default)/с именем <NAME> и указать local.	<pre> rtt(config)# aaa authentication login { default <NAME> } <METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>] </pre>	<p><NAME> – имя списка, задаётся строкой до 31 символа.</p> <p>Способы аутентификации:</p> <ul style="list-style-type: none"> • 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.
2	Задать список методов аутентификации повышения привилегий пользователей по умолчанию (default)/с именем <NAME> и указать enable.	<pre> rtt(config)# aaa authentication enable <NAME><METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>] </pre>	<p><NAME> – имя списка, задаётся строкой до 31 символа.</p> <p>Способы аутентификации:</p> <ul style="list-style-type: none"> • 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.

Шаг	Описание	Команда	Ключи
3	Указать способ перебора методов аутентификации в случае отказа (необязательно).	<code>rtt(config)# aaa authentication mode <MODE></code>	<p><MODE> – способы перебора методов:</p> <ul style="list-style-type: none"> • chain – если сервер вернул FAIL, перейти к следующему в цепочке методу аутентификации; • break – если сервер вернул FAIL, прекратить попытки аутентификации. Если сервер недоступен, продолжить попытки аутентификации следующими в цепочке методами. <p>Значение по умолчанию: chain.</p>
4	Указать количество неудачных попыток аутентификации для блокировки логина пользователя и время блокировки (необязательно).	<code>rtt(config)# aaa authentication attempts max-fail <COUNT> <TIME></code>	<p><COUNT> – количество неудачных попыток аутентификации, после которых произойдет блокировка пользователя, принимает значения [1..65535];</p> <p><TIME> – интервал времени в минутах, на который будет заблокирован пользователь, принимает значения [1..65535].</p> <p>Значение по умолчанию: <COUNT> – 5; <TIME> – 300.</p>
5	Включить запрос на смену пароля по умолчанию для пользователя admin (необязательно).	<code>rtt(config)# security passwords default- expired</code>	
6	Включить режим запрета на использование ранее установленных паролей локальных пользователей (необязательно).	<code>rtt(config)# security passwords history <COUNT></code>	<p><COUNT> – количество паролей, сохраняемых в памяти маршрутизатора. Принимает значение в диапазоне [1..15].</p> <p>Значение по умолчанию: 0.</p>
7	Установить время действия пароля локального пользователя (необязательно).	<code>rtt(config)# security passwords lifetime <TIME></code>	<p><TIME> – интервал времени действия пароля в днях. Принимает значение в диапазоне [1..365].</p> <p>По умолчанию: время действия пароля локального пользователя не ограничено.</p>
8	Установить ограничение на минимальную длину пароля локального пользователя и ENABLE-пароля (необязательно).	<code>rtt(config)# security passwords min-length <NUM></code>	<p><NUM> – минимальное количество символов в пароле. Принимает значение в диапазоне [8..128].</p> <p>Значение по умолчанию: 0.</p>

Шаг	Описание	Команда	Ключи
9	Установить ограничение на максимальную длину пароля локального пользователя и ENABLE-пароля (необязательно).	<code>rtt(config)# security passwords max-length <NUM></code>	<NUM> – максимальное количество символов в пароле. Принимает значение в диапазоне [8..128]. Значение по умолчанию: не ограничено.
10	Установить минимальное количество типов символов, которые должны присутствовать в пароле локального пользователя и ENABLE-пароле (необязательно).	<code>rtt(config)# security passwords symbol-types <COUNT></code>	<COUNT> – минимальное количество типов символов в пароле. Принимает значение в диапазоне [1..4]. Значение по умолчанию: 1.
11	Установить минимальное количество строчных букв в пароле локального пользователя и ENABLE-пароле (необязательно).	<code>rtt(config)# security passwords lower-case <COUNT></code>	<COUNT> – минимальное количество строчных букв в пароле локального пользователя и ENABLE-пароле. Принимает значение в диапазоне [0..128]. Значение по умолчанию: 0.
12	Установить минимальное количество прописных (заглавных) букв в пароле локального пользователя и ENABLE-пароле (необязательно).	<code>rtt(config)# security passwords upper-case <COUNT></code>	<COUNT> – минимальное количество прописных (заглавных) букв в пароле. Принимает значение в диапазоне [0..128]. Значение по умолчанию: 0.
13	Установить минимальное количество цифр в пароле локального пользователя и ENABLE-пароле (необязательно).	<code>rtt(config)# security passwords numeric-count <COUNT></code>	<COUNT> – минимальное количество цифр в пароле. Принимает значение в диапазоне [0..128]. Значение по умолчанию: 0.
14	Установить минимальное количество специальных символов в пароле локального пользователя и ENABLE-пароле (необязательно).	<code>rtt(config)# security passwords special-case <COUNT></code>	<COUNT> – минимальное количество специальных символов в пароле. Принимает значение в диапазоне [0..128]. Значение по умолчанию: 0.
15	Добавить пользователя в локальную базу и перейти в режим настройки параметров пользователя.	<code>rtt(config)# username <NAME></code>	<NAME> – имя пользователя, задаётся строкой до 31 символа.
16	Установить пароль пользователя.	<code>rtt(config-user)# password { <CLEAR-TEXT> encrypted <HASH_SHA512> }</code>	<CLEAR-TEXT> – пароль, задаётся строкой [8 .. 32] символов, принимает значения [0-9a-fA-F]; <HASH_SHA512> – хеш пароля по алгоритму sha512, задаётся строкой из 110 символов.

Шаг	Описание	Команда	Ключи
17	Установить уровень привилегий пользователя.	<code>rtt(config-user) # privilege <PRIV></code>	<PRIV> – необходимый уровень привилегий. Принимает значение [1..15].
18	Установить режим работы учетной записи пользователя (необязательно).	<code>rtt(config-user) # mode <MODE></code>	<p><MODE> – режим работы учетной записи пользователя. Может принимать значения:</p> <ul style="list-style-type: none"> • cli – режим работы по умолчанию, пользователь получает доступ к интерфейсу командной строки, предназначенному для управления, просмотра состояния и мониторинга устройства; • techsupport – пользователь получает доступ к командной оболочке, в которой выполняется процедура отладки устройства совместно с специалистами технической поддержки; • sftp – пользователь используется для организации доступа к встроенному SFTP-серверу, возможность работы в какой-либо командой оболочке при этом у пользователя отсутствует.
19	Указать метод аутентификации SSH-сессий для пользователя (необязательно).	<code>rtt(config-user) # ssh authentication method <METHOD></code>	<p><METHOD> – метод аутентификации SSH-сессий. Может принимать значения:</p> <ul style="list-style-type: none"> • password – аутентификация пользователя при открытии SSH-сессий может быть произведена только по паролю; • pubkey – аутентификация пользователя при открытии SSH-сессий может быть произведена только по публичному ключу; • both – аутентификация пользователя при открытии SSH-сессий может быть произведена как по паролю, так и по публичному ключу.
20	Указать имя файла публичного ключа, который будет использован при аутентификации SSH-сессии пользователя (необязательно).	<code>rtt(config-user) # ssh pubkey <NAME></code>	<NAME> – имя файла публичного ключа, расположенного в разделе crypto:public-key, задаётся строкой до 31 символа.

Шаг	Описание	Команда	Ключи
21	Отключить авторизацию для предустановленного пользователя admin (необязательно).	<code>rtt(config)# no admin login enable</code>	
22	Перейти в режим конфигурирования соответствующего терминала.	<code>rtt(config)# line <TYPE></code>	<p><TYPE> – тип консоли:</p> <ul style="list-style-type: none"> • console – локальная консоль; • telnet – удаленная консоль; • ssh – защищенная удаленная консоль.
23	Активировать список аутентификации входа пользователей в систему.	<code>rtt(config-line-console)# login authentication <NAME></code>	<NAME> – имя списка, задаётся строкой до 31 символа. Создан на шаге 1.
24	Активировать список аутентификации повышения привилегий пользователей.	<code>rtt(config-line-console)# enable authentication <NAME></code>	<NAME> – имя списка, задаётся строкой до 31 символа. Создан на шаге 2.
25	Задать интервал, по истечении которого будет разрываться бездействующая сессия.	<code>rtt(config-line-console)# exec-timeout <SEC></code>	<SEC> – период времени в минутах, принимает значения [1..65535].

14.1.2. Алгоритм настройки AAA по протоколу RADIUS

Шаг	Описание	Команда	Ключи
1	Задать глобальное значение кода DSCP для использования в IP-заголовках исходящих пакетов RADIUS-сервера (необязательно).	<code>rtt(config)# radius-server dscp <DSCP></code>	<p><DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].</p> <p>Значение по умолчанию: 63.</p>
2	Задать глобальное значение количества перезапросов к последнему активному RADIUS-серверу (необязательно).	<code>rtt(config)# radius-server retransmit <COUNT></code>	<p><COUNT> – количество перезапросов к RADIUS-серверу, принимает значения [1..10].</p> <p>Значение по умолчанию: 1.</p>
3	Задать глобальное значение интервала, по истечении которого маршрутизатор считает, что RADIUS-сервер недоступен (необязательно).	<code>rtt(config)# radius-server timeout <SEC></code>	<p><SEC> – период времени в секундах, принимает значения [1..30].</p> <p>Значение по умолчанию: 3 секунды.</p>

Шаг	Описание	Команда	Ключи
4	Добавить RADIUS-сервер в список используемых серверов и перейти в режим его конфигурирования.	<pre>rtt(config)# radius- server host { <IP-ADDR> <IPV6-ADDR> } [vrf <VRF>]</pre>	<p><IP-ADDR> – IP-адрес RADIUS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><IPV6-ADDR> – IPv6-адрес RADIUS-сервера, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]</p> <p><VRF> – имя экземпляра VRF, задается строкой до 31 символа.</p>
5	Задать описание конфигурируемого RADIUS-сервера (необязательно).	<pre>rtt(config-radius- server)# description <description></pre>	<description> – описание RADIUS-сервера, задается строкой до 255 символов.
6	Указать количество неудачных попыток аутентификации для блокировки логина пользователя и времени блокировки (необязательно).	<pre>rtt(config-radius- server)# aaa authentication attempts max-fail <COUNT> <TIME></pre>	<p><COUNT> – количество неудачных попыток аутентификации, после которых произойдет блокировка пользователя, принимает значения [1..65535];</p> <p><TIME> – интервал времени в секундах, на который будет заблокирован пользователь, принимает значения [1..65535].</p> <p>Значение по умолчанию:</p> <p><COUNT> – 5; <TIME> – 300.</p>
7	Задать пароль для аутентификации на удаленном RADIUS-сервере.	<pre>rtt(config-radius- server)# key ascii-text { <TEXT> encrypted <ENCRYPTED-TEXT> }</pre>	<p><TEXT> – строка [8..16] ASCII-символов;</p> <p><ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задаётся строкой [16..32] символов.</p>
8	Задать приоритет использования удаленного RADIUS-сервера (необязательно).	<pre>rtt(config-radius- server)# priority <PRIORITY></pre>	<p><PRIORITY> – приоритет использования удаленного сервера, принимает значения [1..65535].</p> <p>Чем ниже значение, тем приоритетнее сервер.</p> <p>Значение по умолчанию: 1.</p>
9	Задать интервал, по истечении которого маршрутизатор считает, что данный RADIUS-сервер недоступен (необязательно).	<pre>rtt(config-radius- server)# timeout <SEC></pre>	<p><SEC> – период времени в секундах, принимает значения [1..30].</p> <p>Значение по умолчанию: используется значение глобального таймера.</p>

Шаг	Описание	Команда	Ключи
10	Задать IPv4/IPv6-адрес, который будет использоваться в качестве IPv4/IPv6-адреса источника в отправляемых RADIUS-пакетах.	<pre>rtt(config-radius-server)# source-address { <ADDR> object-group <NETWORK_OBJ_GROUP_NAME> }</pre>	<p><ADDR> – IP-адрес источника, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве source address.</p>
11	Задать интерфейс или туннель маршрутизатора, IPv4/IPv6-адрес которого будет использоваться в качестве IPv4/IPv6-адреса источника в отправляемых RADIUS-пакетах.	<pre>rtt(config-radius-server)# source-interface { <IF> <TUN> }</pre>	<p><IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;</p> <p><TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.</p>
12	Задать список методов аутентификации по умолчанию (default)/с именем <NAME> и указать radius.	<pre>rtt(config)# aaa authentication login { default <NAME> } <METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]</pre>	<p><NAME> – имя списка, задаётся строкой до 31 символа.</p> <p>Способы аутентификации:</p> <ul style="list-style-type: none"> • 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.
13	Задать список методов аутентификации повышения привилегий пользователей по умолчанию (default)/с именем <NAME> и указать radius.	<pre>rtt(config)# aaa authentication enable <NAME><METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]</pre>	<p><NAME> – имя списка строка до 31 символа;</p> <ul style="list-style-type: none"> • default – имя списка по умолчанию. <p><METHOD> – способы аутентификации:</p> <ul style="list-style-type: none"> • enable – аутентификация с помощью enable-паролей; • tacacs – аутентификация по протоколу TACACS; • radius – аутентификация по протоколу RADIUS; • ldap – аутентификация по протоколу LDAP.

Шаг	Описание	Команда	Ключи
14	Указать способ перебора методов аутентификации в случае отказа (необязательно).	<code>rtt(config)# aaa authentication mode <MODE></code>	<p><MODE> – способы перебора методов:</p> <ul style="list-style-type: none"> chain – если сервер вернул FAIL, переход к следующему в цепочке методу аутентификации; break – если сервер вернул FAIL, прекратить попытки аутентификации. Если сервер недоступен, продолжить попытки аутентификации следующими в цепочке методами. <p>Значение по умолчанию: chain.</p>
15	Сконфигурировать RADIUS в списке способов учета сессий пользователей (необязательно).	<code>rtt(config)# aaa accounting login start-stop <METHOD 1> [<METHOD 2>]</code>	<p><METHOD> – способы учета:</p> <ul style="list-style-type: none"> tacacs – учет сессий по протоколу TACACS; radius – учет сессий по протоколу RADIUS.
16	Перейти в режим конфигурирования соответствующего терминала.	<code>rtt(config)# line <TYPE></code>	<p><TYPE> – тип консоли:</p> <ul style="list-style-type: none"> console – локальная консоль; telnet – удаленная консоль; ssh – защищенная удаленная консоль.
17	Активировать список аутентификации входа пользователей в систему.	<code>rtt(config-line-console)# login authentication <NAME></code>	<NAME> – имя списка, задаётся строкой до 31 символа. Создан на шаге 12.
18	Активировать список аутентификации повышения привилегий пользователей.	<code>rtt(config-line-console)# enable authentication <NAME></code>	<NAME> – имя списка, задаётся строкой до 31 символа. Создан на шаге 13.

14.1.3. Алгоритм настройки AAA по протоколу TACACS

Шаг	Описание	Команда	Ключи
1	Задать глобальное значение кода DSCP для использования в IP-заголовках исходящих пакетов TACACS-сервера (необязательно).	<code>rtt(config)# tacacs-server dscp <DSCP></code>	<p><DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].</p> <p>Значение по умолчанию: 63.</p>

Шаг	Описание	Команда	Ключи
2	Задать глобальное значение интервала, по истечении которого маршрутизатор считает, что TACACS-сервер недоступен (необязательно).	<code>rtt(config)# tacacs-server timeout <SEC></code>	<SEC> – период времени в секундах, принимает значения [1..30]. Значение по умолчанию: 3 секунды.
3	Добавить TACACS-сервер в список используемых серверов и перейти в режим его конфигурирования.	<code>rtt(config)# tacacs - server host { <IP-ADDR> <IPv6-ADDR> } [vrf <VRF>]</code>	<IP-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, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF] <VRF> – имя экземпляра VRF, задается строкой до 31 символа.
4	Задать описание конфигурируемого TACACS-сервера (необязательно).	<code>rtt(config-tacacs-server)# description <description></code>	<description> – описание TACACS-сервера, задается строкой до 255 символов.
5	Указать количество неудачных попыток аутентификации для блокировки логина пользователя и время блокировки (необязательно)	<code>rtt(config-tacacs-server)# aaa authentication attempts max-fail <COUNT> <TIME></code>	<COUNT> – количество неудачных попыток аутентификации, после которых произойдет блокировка пользователя, принимает значения [1..65535]; <TIME> – интервал времени в минутах, на который будет заблокирован пользователь, принимает значения [1..65535]. Значение по умолчанию: <COUNT> – 5; <TIME> – 300.
6	Задать пароль для аутентификации на удаленном TACACS-сервере	<code>rtt(config-tacacs-server)# key ascii-text { <TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<TEXT> – строка [8..16] ASCII-символов; <ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задаётся строкой [16..32] символов.
7	Задать номер порта для обмена данными с удаленным TACACS-сервером (необязательно).	<code>rtt(config-tacacs-server)# port <PORT></code>	<PORT> – номер TCP-порта для обмена данными с удаленным сервером, принимает значения [1..65535]. Значение по умолчанию: 49 для TACACS-сервера.

Шаг	Описание	Команда	Ключи
8	Задать приоритет использования удаленного TACACS сервера (необязательно).	<code>rtt(config-tacacs-server)# priority <PRIORITY></code>	<p><PRIORITY> – приоритет использования удаленного сервера, принимает значения [1..65535].</p> <p>Чем ниже значение, тем приоритетнее сервер.</p> <p>Значение по умолчанию: 1.</p>
9	Задать IPv4/IPv6-адрес, который будет использоваться в качестве IP/IPv6-адреса источника в отправляемых TACACS-пакетах.	<code>rtt(config-tacacs-server)# source-address { <ADDR> object-group <NETWORK_OBJ_GROUP_NAME> }</code>	<p><ADDR> – IP-адрес источника, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве source address.</p>
10	Задать интерфейс или туннель маршрутизатора, IPv4/IPv6-адрес которого будет использоваться в качестве IPv4/IPv6-адреса источника в отправляемых TACACS-пакетах.	<code>rtt(config-tacacs-server)# source-interface { <IF> <TUN> }</code>	<p><IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;</p> <p><TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.</p>
11	Задать список методов аутентификации по умолчанию (default)/с именем <NAME> и указать tacacs.	<code>rtt(config)# aaa authentication login { default <NAME> } <METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]</code>	<p><NAME> – имя списка, задаётся строкой до 31 символа.</p> <p>Способы аутентификации:</p> <ul style="list-style-type: none"> • 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.

Шаг	Описание	Команда	Ключи
12	Задать список методов аутентификации повышения привилегий пользователей по умолчанию (default)/с именем <NAME> и указать tacacs.	<pre> rtt(config)# aaa authentication enable <NAME><METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>] </pre>	<p><NAME> – имя списка строка до 31 символа;</p> <ul style="list-style-type: none"> default – имя списка по умолчанию. <p><METHOD> – способы аутентификации:</p> <ul style="list-style-type: none"> enable – аутентификация с помощью enable-паролей; tacacs – аутентификация по протоколу TACACS; radius – аутентификация по протоколу RADIUS; ldap – аутентификация по протоколу LDAP.
13	Задать список методов авторизации команд, вводимых пользователем в систему по умолчанию (default)/с именем <NAME> и указать tacacs.	<pre> rtt(config)# aaa authorization commands { default <NAME> } <METHOD 1>[<METHOD 2>] </pre>	<p><NAME> – имя списка, задаётся строкой до 31 символа.</p> <p>Способы аутентификации:</p> <p>local – авторизация с помощью локальной базы пользователей;</p> <p>tacacs – авторизация по списку TACACS-серверов;</p>
14	Указать способ перебора методов аутентификации в случае отказа (необязательно).	<pre> rtt(config)# aaa authentication mode <MODE> </pre>	<p><MODE> – способы перебора методов:</p> <ul style="list-style-type: none"> chain – если сервер вернул FAIL, переход к следующему в цепочке методу аутентификации; break – если сервер вернул FAIL, прекратить попытки аутентификации. Если сервер недоступен, продолжить попытки аутентификации следующими в цепочке методами. <p>Значение по умолчанию: chain.</p>
15	Сконфигурировать список способов учета команд, введенных в CLI (необязательно).	<pre> rtt(config)# aaa accounting commands stop-only <METHOD> </pre>	<p><METHOD> – способы учета:</p> <p>tacacs – учет введенных команд по протоколу TACACS.</p>

Шаг	Описание	Команда	Ключи
16	Сконфигурировать tacacs в списке способов учета сессий пользователей (необязательно).	<code>rtt(config)# aaa accounting login start- stop <METHOD 1> [<METHOD 2>]</code>	<p><METHOD> – способы учета:</p> <ul style="list-style-type: none"> • tacacs – учет сессий по протоколу TACACS; • radius – учет сессий по протоколу RADIUS.
17	Перейти в режим конфигурирования соответствующего терминала.	<code>rtt(config)# line <TYPE></code>	<p><TYPE> – тип консоли:</p> <ul style="list-style-type: none"> • console – локальная консоль; • telnet – удаленная консоль; • ssh – защищенная удаленная консоль.
18	Активировать список аутентификации входа пользователей в систему.	<code>rtt(config-line- console)# login authentication <NAME></code>	<NAME> – имя списка, задаётся строкой до 31 символа. Создан на шаге 11.
19	Активировать список аутентификации повышения привилегий пользователей.	<code>rtt(config-line- console)# enable authentication <NAME></code>	<NAME> – имя списка, задаётся строкой до 31 символа. Создан на шаге 12.
20	Активировать список авторизации команд вводимых пользователем в систему.	<code>rtt(config-line- console)# commands authorization <NAME></code>	<NAME> – имя списка, задаётся строкой до 31 символа. Создан на шаге 13.

14.1.4. Алгоритм настройки AAA по протоколу LDAP

Шаг	Описание	Команда	Ключи
1	Задать базовый DN (Distinguished name), который будет использоваться при поиске пользователей.	<code>rtt(config)# ldap-server base-dn <NAME></code>	<NAME> – базовый DN, задается строкой до 255 символов.
2	Задать интервал, по истечении которого устройство считает, что LDAP-сервер недоступен (необязательно).	<code>rtt(config)# ldap-server bind timeout <SEC></code>	<p><SEC> – период времени в секундах, принимает значения [1..30].</p> <p>Значение по умолчанию: 3 секунды.</p>
3	Задать DN (Distinguished name) пользователя с правами администратора, под которым будет происходить авторизация на LDAP-сервере при поиске пользователей.	<code>rtt(config)# ldap-server bind authenticate root-dn <NAME></code>	<NAME> – DN пользователя с правами администратора, задается строкой до 255 символов.

Шаг	Описание	Команда	Ключи
4	Задать пароль пользователя с правами администратора, под которым будет происходить авторизация на LDAP-сервере при поиске пользователей.	<code>rtt(config)# ldap-server bind authenticate root-password ascii-text { <TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<TEXT> – строка [8..16] ASCII-символов; <ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задаётся строкой [16..32] символов.
5	Задать имя класса объектов, среди которых необходимо выполнять поиск пользователей на LDAP-сервере (необязательно).	<code>rtt(config)# ldap-server search filter user-object-class <NAME></code>	<NAME> – имя класса объектов, задаётся строкой до 127 символов. Значение по умолчанию: posixAccount.
6	Задать область поиска пользователей в дереве LDAP-сервера (необязательно).	<code>rtt(config)# ldap-server search scope <SCOPE></code>	<SCOPE> – область поиска пользователей на LDAP-сервере, принимает следующие значения: <ul style="list-style-type: none"> • 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.
7	Задать интервал, по истечении которого устройство считает, что LDAP-сервер не нашел записей пользователей, подходящих под условие поиска (необязательно).	<code>rtt(config)# ldap-server search timeout <SEC></code>	<SEC> – период времени в секундах, принимает значения [0..30] Значение по умолчанию: 0 – устройство ожидает завершения поиска и получения ответа от LDAP-сервера.
8	Задать имя атрибута объекта, со значением которого идет сравнение имени искомого пользователя на LDAP-сервере (необязательно).	<code>rtt(config)# ldap-server naming-attribute <NAME></code>	<NAME> – имя атрибута объекта, задаётся строкой до 127 символов. Значение по умолчанию: uid.
9	Задать имя атрибута объекта, значение которого будет определять начальные привилегии пользователя на устройстве (необязательно).	<code>rtt(config)# ldap-server privilege-level-attribute <NAME></code>	<NAME> – имя атрибута объекта, задаётся строкой до 127 символов. Значение по умолчанию: priv-lvl.

Шаг	Описание	Команда	Ключи
10	Задать глобальное значение кода DSCP для использования в IP-заголовках исходящих пакетов LDAP-сервера (необязательно).	<code>rtt(config)# ldap-server dscp <DSCP></code>	<p><DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].</p> <p>Значение по умолчанию: 63.</p>
11	Добавить LDAP-сервер в список используемых серверов и перейти в режим его конфигурирования.	<code>rtt(config)# ldap-server</code> <code>host { <IP-ADDR> <IPv6-ADDR> }</code> <code>[vrf <VRF>]</code>	<p><IP-ADDR> – IP-адрес LDAP-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]</p> <p><IPv6-ADDR> – IPv6-адрес LDAP-сервера, задаётся в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]</p> <p><VRF> – имя экземпляра VRF, задается строкой до 31 символа.</p>
12	Задать описание конфигурируемого LDAP-сервера (необязательно).	<code>rtt(config-ldap-server)# description <description></code>	<description> – описание LDAP-сервера, задается строкой до 255 символов.
13	Указать количество неудачных попыток аутентификации для блокировки логина пользователя и время блокировки (необязательно).	<code>rtt(config-ldap-server)#</code> <code>aaa authentication</code> <code>attempts max-fail <COUNT></code> <code><TIME></code>	<p><COUNT> – количество неудачных попыток аутентификации, после которых произойдет блокировка пользователя, принимает значения [1..65535];</p> <p><TIME> – интервал времени в минутах, на который будет заблокирован пользователь, принимает значения [1..65535].</p> <p>Значение по умолчанию:</p> <p><COUNT> – 5; <TIME> – 300.</p>
14	Задать номер порта для обмена данными с удаленным LDAP-сервером (необязательно).	<code>rtt(config-ldap-server)# port <PORT></code>	<p><PORT> – номер TCP-порта для обмена данными с удаленным сервером, принимает значения [1..65535].</p> <p>Значение по умолчанию: 389 для LDAP-сервера.</p>

Шаг	Описание	Команда	Ключи
15	Задать приоритет использования удаленного LDAP-сервера (необязательно).	<code>rtt(config-ldap-server)# priority <PRIORITY></code>	<p><PRIORITY> – приоритет использования удаленного сервера, принимает значения [1..65535].</p> <p>Чем ниже значение, тем приоритетнее сервер.</p> <p>Значение по умолчанию: 1.</p>
16	Задать IPv4/IPv6-адрес, который будет использоваться в качестве IP/IPv6-адреса источника в отправляемых LDAP-пакетах.	<code>rtt(config-ldap-server)# source-address { <ADDR> object-group <NETWORK_OBJ_GROUP_NAME> }</code>	<p><ADDR> – IP-адрес источника, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве source address.</p>
17	Задать интерфейс или туннель маршрутизатора, IPv4/IPv6-адрес которого будет использоваться в качестве IPv4/IPv6-адреса источника в отправляемых LDAP-пакетах.	<code>rtt(config-ldap-server)# source-interface { <IF> <TUN> }</code>	<p><IF> – имя интерфейса устройства, задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;</p> <p><TUN> – имя туннеля устройства, задается в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.</p>
18	Задать список методов аутентификации по умолчанию (default)/с именем <NAME> и указать ldap.	<code>rtt(config)# aaa authentication login { default <NAME> } <METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]</code>	<p><NAME> – имя списка, задается строкой до 31 символа.</p> <p>Способы аутентификации:</p> <ul style="list-style-type: none"> • 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.

Шаг	Описание	Команда	Ключи
19	Задать список методов аутентификации повышения привилегий пользователей по умолчанию (default)/с именем <NAME> и указать ldap.	<pre> rtt(config)# aaa authentication enable <NAME> <METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>] </pre>	<p><NAME> – имя списка строка до 31 символа;</p> <ul style="list-style-type: none"> default – имя списка по умолчанию. <p><METHOD> – способы аутентификации:</p> <ul style="list-style-type: none"> enable – аутентификация с помощью enable-паролей; tacacs – аутентификация по протоколу TACACS; radius – аутентификация по протоколу RADIUS; ldap – аутентификация по протоколу LDAP.
20	Указать способ перебора методов аутентификации в случае отказа.	<pre> rtt(config)# aaa authentication mode <MODE> </pre>	<p><MODE> – способы перебора методов:</p> <ul style="list-style-type: none"> chain – если сервер вернул FAIL, переход к следующему в цепочке методу аутентификации; break – если сервер вернул FAIL, прекратить попытки аутентификации. Если сервер недоступен, продолжить попытки аутентификации следующими в цепочке методами. <p>Значение по умолчанию: chain.</p>
21	Перейти в режим конфигурирования соответствующего терминала.	<pre> rtt(config)# line <TYPE> </pre>	<p><TYPE> – тип консоли:</p> <ul style="list-style-type: none"> console – локальная консоль; telnet – удаленная консоль; ssh – защищенная удаленная консоль.
22	Активировать список аутентификации входа пользователей в систему.	<pre> rtt(config-line-console)# login authentication <NAME> </pre>	<p><NAME> – имя списка, задается строкой до 31 символа. Создан на шаге 17.</p>
23	Активировать список аутентификации повышения привилегий пользователей.	<pre> rtt(config-line-console)# enable authentication <NAME> </pre>	<p><NAME> – имя списка, задается строкой до 31 символа. Создан на шаге 18.</p>

14.1.5. Пример настройки аутентификации по Telnet через RADIUS-сервер

Задача:

Настроить аутентификацию пользователей, подключающихся по Telnet, через RADIUS (192.168.16.1/24).

Решение:

Настроим подключение к RADIUS-серверу и укажем ключ (password):

```
rtt# configure
rtt(config)# radius-server host 192.168.16.1
rtt(config-radius-server)# key ascii-text encrypted 8CB5107EA7005AFF
rtt(config-radius-server)# exit
```

Создадим профиль аутентификации:

```
rtt(config)# aaa authentication login log radius
```

Укажем режим аутентификации, используемый при подключении по Telnet-протоколу:

```
rtt(config)# line telnet
rtt(config-line-telnet)# login authentication log
rtt(config-line-telnet)# exit
rtt(config)# exit
```

Просмотреть информацию по настройкам подключения к RADIUS-серверу можно командой:

```
rtt# show aaa radius-servers
```

Посмотреть профили аутентификации можно командой:

```
rtt# show aaa authentication
```

14.2. Настройка привилегий команд

Настройка привилегий команд является гибким инструментом, который позволяет назначить набору команд минимально необходимый уровень пользовательских привилегий (1-15). В дальнейшем при создании пользователя можно задать уровень привилегий, определяя ему доступный набор команд.

- *1-9 уровни* – позволяют использовать все команды мониторинга (show ...);
- *10-14 уровни* – позволяют использовать все команды кроме команд перезагрузки устройства, управления пользователями и ряда других;
- *15 уровень* – позволяет использовать все команды.

14.2.1. Алгоритм настройки

Для изменения минимального уровня привилегий необходимого для выполнения команды CLI используется команда:

```
rtt(config)# privilege <COMMAND-MODE> level <PRIV><COMMAND>
```

<COMMAND-MODE> – командный режим;

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

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

14.2.2. Пример настройки привилегий команд

Задача:

Перевести все команды просмотра информации об интерфейсах на уровень привилегий 10, кроме команды «show interfaces bridges». Команду «show interfaces bridges» перевести на уровень привилегий 3.

Решение:

В режиме конфигурирования определим команды, разрешенные на использование с уровнем привилегий 10 и уровнем привилегий 3:

```
rtt(config)# privilege root level 3 "show interfaces bridge"
rtt(config)# privilege root level 10 "show interfaces"
```

14.3. Настройка логирования и защиты от сетевых атак

14.3.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Включить защиту от ICMP flood-атак.	<code>rtt(config)# ip firewall screen dos-defense icmp-threshold { <NUM> }</code>	<NUM> – количество ICMP-пакетов в секунду, задается в диапазоне [1..10000].
2	Включить защиту от land-атак.	<code>rtt(config)# firewall screen dos-defense land</code>	
3	Включить ограничение числа пакетов, отправляемых за одну секунду на один адрес назначения	<code>rtt(config)# ip firewall screen dos-defense limit-session-destination { <NUM> }</code>	<NUM> – ограничение числа IP-пакетов в секунду, задается в диапазоне [1..10000].
4	Включить ограничение числа пакетов, отправляемых за одну секунду с единого адреса источника	<code>rtt(config)# ip firewall screen dos-defense limit-session-source { <NUM> }</code>	<NUM> – ограничение числа IP-пакетов в секунду, задается в диапазоне [1..10000].

Шаг	Описание	Команда	Ключи
5	Включить защиту от SYN flood-атак.	<code>rtt(config)# ip firewall screen dos-defense syn-flood { <NUM> } [src- dsr]</code>	<NUM> – максимальное количество TCP-пакетов с установленным флагом SYN в секунду, задается в диапазоне [1..10000]. src-dst – ограничение количества TCP-пакетов с установленным флагом SYN на основании адреса источника и адреса назначения.
6	Включить защиту от UDP flood-атак.	<code>rtt(config)# ip firewall screen dos-defense udp-threshold { <NUM> }</code>	<NUM> – максимальное количество UDP-пакетов в секунду, задается в диапазоне [1..10000].
7	Включить защиту от winnuke-атак.	<code>rtt(config)# ip firewall screen dos-defense winnuke</code>	
8	Включить блокировку TCP-пакетов с установленным флагом FIN и не установленным флагом ACK.	<code>rtt(config)# ip firewall screen spy-blocking fin- no-ack</code>	
9	Включить блокировку ICMP-пакетов различных типов.	<code>rtt(config)# ip firewall screen spy-blocking icmp- type</code>	<TYPE> – тип ICMP, может принимать значения: <ul style="list-style-type: none">• destination-unreachable• echo-request• reserved• source-quench• time-exceeded
10	Включить защиту от IP sweep-атак.	<code>rtt(config)# ip firewall screen spy-blocking ip- sweep { <NUM> }</code>	<NUM> – интервал выявления ip sweep атаки, задается в миллисекундах [1..1000000].
11	Включить защиту от port scan-атак.	<code>rtt(config)# ip firewall screen spy-blocking port- scan { <threshold> } [<TIME>]</code>	<threshold> – интервал в секундах, в течение которого будет фиксироваться port scan-атака [1..10000]. <TIME> – время блокировки в миллисекундах [1..1000000].
12	Включить защиту от IP spoofing-атак.	<code>rtt(config)# ip firewall screen spy-blocking spoofing</code>	
13	Исключить из защиты от IP-spoofing атак указанную Object Group.	<code>rtt(config)# ip firewall screen spy-blocking spoofing exclude <object- group></code>	<object-group> – список разрешённых для spoofing подсетей.

Шаг	Описание	Команда	Ключи
14	Включить блокировку TCP-пакетов, с установленными флагами SYN и FIN.	<code>rtt(config)# ip firewall screen spy-blocking syn-fin</code>	
15	Включить блокировку TCP-пакетов, со всеми флагами или с набором флагов: FIN, PSH, URG. Данной командой обеспечивается защита от атаки XMAS.	<code>rtt(config)# ip firewall screen spy-blocking tcp-all-flag</code>	
16	Включить блокировку TCP-пакетов, с нулевым полем flags.	<code>rtt(config)# ip firewall screen spy-blocking tcp-no-flag</code>	
17	Включить блокировку фрагментированных ICMP-пакетов.	<code>rtt(config)# ip firewall screen suspicious-packets icmp-fragment</code>	
18	Включить блокировку фрагментированных IP-пакетов.	<code>rtt(config)# ip firewall screen suspicious-packets ip-fragment</code>	
19	Включить блокировку ICMP-пакетов длиной более 1024 байт.	<code>rtt(config)# ip firewall screen suspicious-packets icmp-fragment</code>	
20	Включить блокировку фрагментированных TCP-пакетов, с флагом SYN.	<code>rtt(config)# ip firewall screen suspicious-packets syn-fragment</code>	
21	Включить блокировку фрагментированных UDP-пакетов.	<code>rtt(config)# ip firewall screen suspicious-packets udp-fragment</code>	
22	Включить блокировку пакетов, с ID протокола в заголовке IP равном 137 и более.	<code>rtt(config)# ip firewall screen suspicious-packets unknown-protocols</code>	
23	Установить частоту оповещения (по SNMP, syslog и в CLI) об обнаруженных и отраженных сетевых атаках.	<code>rtt(config)# ip firewall logging interval <NUM></code>	<NUM> – интервал времени в секундах [30 .. 2147483647].
24	Включить механизм обнаружения и логирования DoS-атак через CLI, Syslog и по SNMP.	<code>rtt(config)# logging firewall screen dos-defense <ATTACK_TYPE></code>	<ATTACK_TYPE> – тип DoS-атаки, принимает значения: icmp-threshold, land, limit-session-destination, limit-session-source, syn-flood, udp-threshold, winnuke.

Шаг	Описание	Команда	Ключи
25	Включить механизм обнаружения и логирования шпионской активности через CLI, Syslog и по SNMP.	<pre>rtt(config)# logging firewall screen spy- blocking { <ATAK_TYPE> icmp-type <ICMP_TYPE> }</pre>	<p><ATAK_TYPE> – тип шпионской активности, принимает значения: fin-no-ack, ip-sweep, port-scan, spoofing, syn-fin, tcp-all-flag, tcp-no-flag.</p> <p><ICMP_TYPE> – тип ICMP, принимает значения: destination-unreachable, echo-request, reserved, source-quench, time-exceeded.</p>
26	Включить механизм обнаружения нестандартных пакетов и логирования через CLI, Syslog и по SNMP.	<pre>rtt(config)# logging firewall screen suspicious-packets <PACKET_TYPE></pre>	<PACKET_TYPE> – тип нестандартных пакетов, принимает значения: icmp-fragment, ip-fragment, large-icmp, syn-fragment, udp-fragment, unknown-protocols.

14.3.2. Описание механизмов защиты от атак

Команда	Описание
ip firewall screen dos-defense icmp-threshold	Данная команда включает защиту от ICMP flood-атак. При включенной защите ограничивается количество ICMP-пакетов всех типов в секунду для одного source-адреса. Атака приводит к перегрузке хоста и выводу его из строя из-за необходимости обрабатывать каждый запрос и отвечать на него.
firewall screen dos-defense land	Данная команда включает защиту от land-атак. При включенной защите блокируются пакеты с одинаковыми source и destination IP-адресами и флагом SYN в заголовке TCP. Атака приводит к перегрузке хоста и выводу его из строя из-за необходимости обрабатывать каждый TCP SYN пакет и попыток хоста установить TCP-сессию с самим собой.
ip firewall screen dos-defense limit-session-destination	Когда таблица IP-сессий хоста переполняется, он больше не в состоянии организовывать новые сессии и отбрасывает запросы (такое может происходить при различных DoS-атаках: SYN flood, UDP flood, ICMP flood, и т.д.). Команда включает ограничение числа пакетов, передаваемых за секунду на один адреса назначения, которое смягчает DoS-атаки.
ip firewall screen dos-defense limit-session-source	Когда таблица IP-сессий хоста переполняется, он больше не в состоянии организовывать новые сессии и отбрасывает запросы (такое может происходить при различных DoS-атаках: SYN flood, UDP flood, ICMP flood, и т.д.). Команда включает ограничение числа пакетов, передаваемых за секунду с одного адреса источника, которое смягчает DoS-атаки.
ip firewall screen dos-defense syn-flood	Данная команда включает защиту от SYN flood-атак. При включенной защите ограничивается количество TCP-пакетов с установленным флагом SYN в секунду для одного адреса назначения. Атака приводит к перегрузке хоста и выводу его из строя из-за необходимости обрабатывать каждый TCP SYN пакет и попыток установить TCP-сессию.

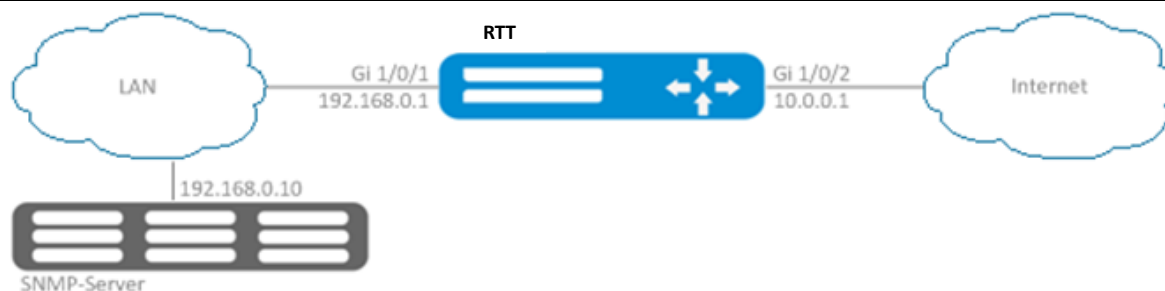
Команда	Описание
ip firewall screen dos-defense udp-threshold	Данная команда включает защиту от UDP flood-атак. При включенной защите ограничивается количество UDP-пакетов в секунду для одного адреса назначения. Атака приводит к перегрузке хоста и выводу его из строя из-за массивного UDP-трафика.
ip firewall screen dos-defense winnuke	Данная команда включает защиту от winnuke-атак. При включенной защите блокируются TCP-пакеты с установленным флагом URG и 139 портом назначения. Атака приводит к выходу из строя старых версий Windows (до 95 версии).
ip firewall screen spy-blocking fin-no-ack	Данная команда включает блокировку TCP-пакетов с установленным флагом FIN и не установленным флагом ACK. Такие пакеты являются нестандартными, и по ответу можно определить операционную систему жертвы.
ip firewall screen spy-blocking icmp-type destination-unreachable	Данная команда включает блокировку всех ICMP-пакетов 3 типа (destination-unreachable), включая пакеты, сгенерированные самим маршрутизатором. Защита не дает злоумышленнику узнать о топологии сети и доступности хостов.
ip firewall screen spy-blocking icmp-type echo-request	Данная команда включает блокировку всех ICMP-пакетов 8 типа (echo-request), включая пакеты, сгенерированные самим маршрутизатором. Защита не дает злоумышленнику узнать о топологии сети и доступности хостов.
ip firewall screen spy-blocking icmp-type reserved	Данная команда включает блокировку всех ICMP-пакетов 2 и 7 типов (reserved), включая пакеты, сгенерированные самим маршрутизатором. Защита не дает злоумышленнику узнать о топологии сети и доступности хостов.
ip firewall screen spy-blocking icmp-type source-quench	Данная команда включает блокировку всех ICMP-пакетов 4 типа (source quench), включая пакеты, сгенерированные самим маршрутизатором. Защита не дает злоумышленнику узнать о топологии сети и доступности хостов.
ip firewall screen spy-blocking icmp-type time-exceeded	Данная команда включает блокировку всех ICMP-пакетов 11 типа (time exceeded), включая пакеты, сгенерированные самим маршрутизатором. Защита не дает злоумышленнику узнать о топологии сети и доступности хостов.
ip firewall screen spy-blocking ip-sweep	Данная команда включает защиту от IP sweep-атак. При включенной защите, если в течение заданного в параметрах интервала приходит более 10 ICMP-запросов от одного источника, первые 10 запросов пропускаются маршрутизатором, а 11 и последующие отбрасываются на оставшееся время интервала. Защита не дает злоумышленнику узнать о топологии сети и доступности хостов.
ip firewall screen spy-blocking port-scan	Данная команда включает защиту от port scan-атак. Если в течение первого заданного интервала времени (<threshold>) на один источник приходит более 10 TCP-пакетов с флагом SYN на разные TCP-порты или более 10 UDP-пакетов на разные UDP-порты, то такое поведение фиксируется как port scan-атака, и все последующие пакеты такого рода от источника блокируются на второй заданный интервал времени (<TIME>). Злоумышленник не сможет быстро просканировать открытые порты на устройстве.
ip firewall screen spy-blocking spoofing	Данная команда включает защиту от ip spoofing-атак. При включенной защите маршрутизатор проверяет пакеты на соответствие адреса источника и записей в таблице маршрутизации, и в случае несоответствия пакет отбрасывается. Например, если пакет с адресом источника 10.0.0.1/24 приходит на интерфейс Gi1/0/1, а в таблице маршрутизации данная подсеть располагается за интерфейсом Gi1/0/2, то считается, что адрес источника был подменен. Защищает от вторжений в сеть с подмененными source IP-адресами.

Команда	Описание
ip firewall screen spy-blocking spoofing exclude <object-group>	Данная команда исключает из защиты от IP-spoofing атак указанную Object Group. В Object Group помещается список из допустимых адресов, которые будут игнорироваться. Команда используется вместе с основной ip firewall screen spy-blocking spoofing, которая включает защиту, иначе она не будет иметь эффекта. В случае, если на маршрутизатор приходит spoofing от разрешённых подсетей (например, частый опрос устройств в сети), пакеты пропускаются.
ip firewall screen spy-blocking syn-fin	Данная команда включает блокировку TCP-пакетов с установленными флагами SYN и FIN. Такие пакеты являются нестандартными, и по ответу можно определить операционную систему жертвы.
ip firewall screen spy-blocking tcp-all-flag	Данная команда включает блокировку TCP-пакетов со всеми флагами или с набором флагов: FIN, PSH, URG. Обеспечивается защита от атаки XMAS.
ip firewall screen spy-blocking tcp-no-flag	Данная команда включает блокировку TCP-пакетов с нулевым полем flags. Такие пакеты являются нестандартными, и по ответу можно определить операционную систему жертвы.
ip firewall screen suspicious-packets icmp-fragment	Данная команда включает блокировку фрагментированных ICMP-пакетов. ICMP-пакеты обычно небольшого размера и необходимости в их фрагментировании нет.
ip firewall screen suspicious-packets ip-fragment	Данная команда включает блокировку фрагментированных пакетов.
ip firewall screen suspicious-packets large-icmp	Данная команда включает блокировку ICMP-пакетов длиной более 1024 байт.
ip firewall screen suspicious-packets syn-fragment	Данная команда включает блокировку фрагментированных TCP-пакетов с флагом SYN. TCP-пакеты с SYN-флагом обычно небольшого размера и необходимости в их фрагментировании нет. Защита предотвращает накопление фрагментированных пакетов в буфере.
ip firewall screen suspicious-packets udp-fragment	Данная команда включает блокировку фрагментированных UDP-пакетов.
ip firewall screen suspicious-packets unknown-protocols	Данная команда включает блокировку пакетов, с ID протокола в заголовке IP равном 137 и более.

14.3.3. Пример настройки логирования и защиты от сетевых атак

Задача:

Необходимо защитить LAN-сеть и маршрутизатор RTT от сетевых атак land, syn-flood, ICMP flood и настроить оповещение об атаках по SNMP на SNMP-сервер 192.168.0.10.



Решение:

Предварительно необходимо настроить интерфейсы и firewall (настройка firewall или ее отсутствие не повлияют на работу защиты от сетевых атак):

```

rtt(config)# security zone LAN
rtt(config-security-zone)# exit
rtt(config)# security zone WAN
rtt(config-security-zone)# exit
rtt(config)# security zone-pair LAN WAN
rtt(config-security-zone-pair)# rule 100
rtt(config-security-zone-pair-rule)# action permit
rtt(config-security-zone-pair-rule)# enable
rtt(config-security-zone-pair-rule)# exit
rtt(config-security-zone-pair)# exit
rtt(config)# security zone-pair WAN LAN
rtt(config-security-zone-pair)# rule 100
rtt(config-security-zone-pair-rule)# action permit
rtt(config-security-zone-pair-rule)# enable
rtt(config-security-zone-pair-rule)# exit
rtt(config-security-zone-pair)# exit
rtt(config)# exit
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# security-zone LAN
rtt(config-if-gi)# ip address 192.168.0.1/24
rtt(config-if-gi)# exit
rtt(config)# interface gigabitethernet 1/0/2
rtt(config-if-gi)# security-zone WAN
rtt(config-if-gi)# ip address 10.0.0.1/24
rtt(config-if-gi)# exit
  
```

Настроим защиту от land, syn-flood, ICMP flood-атак:

```

rtt(config)# ip firewall screen dos-defense land
rtt(config)# ip firewall screen dos-defense syn-flood 100 src-dst
rtt(config)# ip firewall screen dos-defense icmp-threshold 100
  
```

Настроим логирование обнаруженных атак:

```

rtt(config)# logging firewall screen dos-defense land
rtt(config)# logging firewall screen dos-defense syn-flood
rtt(config)# logging firewall screen dos-defense icmp-threshold
  
```

Настроим SNMP-сервер, на который будут отправляться трапы:


```

rtt(config)# snmp-server
rtt(config)# snmp-server host 192.168.0.10
rtt(config)# snmp-server enable traps screen land
rtt(config)# snmp-server enable traps screen syn-flood
rtt(config)# snmp-server enable traps screen icmp-threshold

```

Посмотреть статистику по зафиксированным сетевым атакам можно командой:

```

rtt# show ip firewall screen counters

```

14.4. Конфигурирование Firewall

Firewall – комплекс аппаратных или программных средств, осуществляющий контроль и фильтрацию проходящих через него сетевых пакетов в соответствии с заданными правилами.

14.4.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать зоны безопасности.	<pre> rtt(config)# security zone <zone-name1> rtt(config)# security zone <zone-name2> </pre>	<p><zone-name> – до 12 символов.</p> <p>Имена all, any и self зарезервированы.</p>
2	Задать описание зоны безопасности.	<pre> rtt(config-security- zone)# description <description> </pre>	<description> – до 255 символов.
3	Указать экземпляр VRF, в котором будет работать данная зона безопасности (необязательно).	<pre> rtt(config- security- zone)# ip vrf forwarding <VRF> </pre>	<VRF> – имя VRF, задается строкой до 31 символа.
4	Включить счетчики сессий для NAT и Firewall (необязательно, снижает производительность).	<pre> rtt(config)# ip firewall sessions counters </pre>	
5	Отключить фильтрацию пакетов, для которых не удалось определить принадлежность к какому-либо известному соединению и которые не являются началом нового соединения (необязательно, снижает производительность).	<pre> rtt(config)# ip firewall sessions unknown <ACTION> </pre>	<p><ACTION> – правило обработки неизвестных сессий для межсетевого экрана:</p> <p>permit – разрешить неизвестные сессии;</p> <p>deny – отбрасывать неизвестные сессии;</p> <p>reject – отбрасывать неизвестные сессии и отправлять обратно пакет с ошибкой.</p>

Шаг	Описание	Команда	Ключи
6	<p>Выбрать режим работы межсетевого экрана (необязательно).</p> <p>В режиме stateful проверяется только первый пакет сессии, и если «прямой» трафик разрешён, «ответный» трафик разрешается автоматически.</p> <p>В режиме stateless происходит проверка каждого пакета. «Прямой» и «ответный» трафики требуется разрешать в соответствующих zone-pair (см. шаг 29).</p> <p>Работа межсетевого экрана по списку приложений возможна только в режиме stateless.</p>	<pre>rtt(config)# ip firewall mode <MODE></pre>	<p><MODE> – режим работы межсетевого экрана, может принимать значения: stateful, stateless.</p> <p>Значение по умолчанию: stateful.</p>
7	<p>Определить время жизни сессии для неподдерживаемых протоколов (необязательно).</p>	<pre>rtt(config)# ip firewall sessions generic-timeout <TIME></pre>	<p><TIME> – время жизни сессии для неподдерживаемых протоколов, принимает значения в секундах [1..8553600].</p> <p>По умолчанию: 60 секунд.</p>
8	<p>Определить время жизни ICMP-сессии, по истечении которого она считается устаревшей (необязательно).</p>	<pre>rtt(config)# ip firewall sessions icmp- timeout <TIME></pre>	<p><TIME> – время жизни ICMP-сессии, принимает значения в секундах [1..8553600].</p> <p>По умолчанию: 30 секунд.</p>
9	<p>Определить время жизни ICMPv6-сессии, по истечении которого она считается устаревшей (необязательно).</p>	<pre>rtt(config)# ip firewall sessions icmpv6-timeout <TIME></pre>	<p><TIME> – время жизни ICMP-сессии, принимает значения в секундах [1..8553600].</p> <p>По умолчанию: 30 секунд.</p>
10	<p>Определить размер таблицы сессий, ожидающих обработки (необязательно).</p>	<pre>rtt(config)# ip firewall sessions max- expect <COUNT></pre>	<p><COUNT> – размер таблицы, принимает значения [1..8553600].</p> <p>По умолчанию: 256.</p>
11	<p>Определить размер таблицы отслеживаемых сессий (необязательно).</p>	<pre>rtt(config)# ip firewall sessions max- tracking <COUNT></pre>	<p><COUNT> – размер таблицы, принимает значения [1..8553600].</p> <p>По умолчанию: 512000.</p>

Шаг	Описание	Команда	Ключи
12	Определить время жизни TCP-сессии в состоянии «соединение устанавливается», по истечении которого она считается устаревшей (необязательно).	<code>rtt(config)# ip firewall sessions tcp- connect-timeout <TIME></code>	<TIME> – время жизни TCP-сессии в состоянии «соединение устанавливается», принимает значения в секундах [1..8553600]. По умолчанию: 60 секунд.
13	Определить время жизни TCP-сессии в состоянии «соединение закрывается», по истечении которого она считается устаревшей (необязательно).	<code>rtt(config)# ip firewall sessions tcp- disconnect-timeout <TIME></code>	<TIME> – время жизни TCP-сессии в состоянии «соединение закрывается» принимает значения в секундах [1..8553600]. По умолчанию: 30 секунд.
14	Определить время жизни TCP-сессии в состоянии «соединение установлено», по истечении которого она считается устаревшей (необязательно).	<code>rtt(config)# ip firewall sessions tcp- established-timeout <TIME></code>	<TIME> – время жизни TCP-сессии в состоянии «соединение установлено», принимает значения в секундах [1..8553600]. По умолчанию: 120 секунд.
15	Определить время ожидания, по истечении которого происходит фактическое удаление закрытой TCP-сессии из таблицы отслеживаемых сессий (необязательно).	<code>rtt(config)# ip firewall sessions tcp- latecome-timeout <TIME></code>	<TIME> – время ожидания, принимает значения в секундах [1..8553600]. По умолчанию: 120 секунд.
16	Включить функцию отслеживания сессий уровня приложений для отдельных протоколов (необязательно).	<code>rtt(config)# ip firewall sessions tracking</code>	<PROTOCOL> – протокол уровня приложений [ftp, h323, pptp, netbios-ns, tftp], сессии которого должны отслеживаться. <OBJECT-GROUP-SERVICE> – имя профиля TCP/UDP-портов sip-сессии, задаётся строкой до 31 символа. Если группа не указана, то отслеживание сессий sip будет осуществляться для порта 5060. Вместо имени отдельного протокола можно использовать ключ «all», который включает функцию отслеживания сессий уровня приложений для всех доступных протоколов. По умолчанию – отключено для всех протоколов.

Шаг	Описание	Команда	Ключи
17	Определить время жизни UDP-сессии в состоянии «соединение подтверждено», по истечении которого она считается устаревшей (необязательно).	<code>rtt(config)# ip firewall sessions udp- assured-timeout <TIME></code>	<TIME> – время жизни UDP-сессии в состоянии «соединение подтверждено», принимает значения в секундах [1..8553600]. По умолчанию: 180 секунд.
18	Определить время жизни UDP-сессии в состоянии «соединение не подтверждено», по истечении которого она считается устаревшей.	<code>rtt(config)# ip firewall sessions udp- wait-timeout <TIME></code>	<TIME> – время жизни UDP-сессии в состоянии «соединение не подтверждено», принимает значения в секундах [1..8553600]. По умолчанию: 30 секунд.
19	Создать списки MAC-адресов, которые будут использоваться при фильтрации.	<code>rtt(config)# object- group mac <obj-group- name></code>	<obj-group-name> – до 31 символа.
20	Задать описание списка MAC-адресов (необязательно).	<code>rtt(config-object- group-mac)# description <description></code>	<description> – описание профиля, задается строкой до 255 символов.
21	Внести необходимые MAC-адреса в список.	<code>rtt(config-object- group-mac)# mac address <ADDR> <WILDCARD></code>	<ADDR> – MAC-адрес, 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.
22	Создать списки IP-адресов, которые будут использоваться при фильтрации.	<code>rtt(config)# object- group network <obj- group-name></code>	<obj-group-name> – до 31 символа.
23	Задать описание списка IP-адресов (необязательно).	<code>rtt(config-object- group-network)# description <description></code>	<description> – описание профиля, задается строкой до 255 символов.
24	Внести необходимые IPv4/IPv6-адреса в список.	<code>rtt(config-object- group-network)# ip prefix <ADDR/LEN> [unit <ID>]</code>	<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]. <ID> – номер юнита, принимает значения [1..4].

Шаг	Описание	Команда	Ключи
		<pre>rtt(config-object-group-network)# ip address-range <FROM-ADDR>-<TO-ADDR> [unit <ID>]</pre>	<p><FROM-ADDR> – начальный IP-адрес диапазона адресов;</p> <p><TO-ADDR> – конечный IP-адрес диапазона адресов, опциональный параметр. Если параметр не указан, то командой задаётся одиночный IP-адрес.</p> <p>Адреса задаются в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p> <p><ID> – номер юнита, принимает значения [1..4].</p>
		<pre>rtt(config-object-group-network)# ipv6 prefix <IPv6-ADDR/LEN> [unit <ID>]</pre>	<p><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].</p> <p><ID> – номер юнита, принимает значения [1..4].</p>
		<pre>rtt(config-object-group-network)# ipv6 address-range <FROM-ADDR>-<TO-ADDR> [unit <ID>]</pre>	<p><FROM-ADDR> – начальный IPv6-адрес диапазона адресов;</p> <p><TO-ADDR> – конечный IPv6-адрес диапазона адресов, опциональный параметр. Если параметр не указан, то командой задаётся одиночный IPv6-адрес.</p> <p>Адреса задаются в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].</p> <p><ID> – номер юнита, принимает значения [1..4].</p>
25	Создать списки сервисов, которые будут использоваться при фильтрации.	<pre>rtt(config)# object- group service <obj- group-name></pre>	<p><obj-group-name> – имя профиля сервисов, задается строкой до 31 символа.</p>
26	Задать описание списка сервисов (необязательно).	<pre>rtt(config-object- group-service)# description <description></pre>	<p><description> – описание профиля, задается строкой до 255 символов.</p>

Шаг	Описание	Команда	Ключи
27	Внести необходимые сервисы (tcp/udp-порты) в список.	<code>rtt (config-object-group-service) # port-range <port></code>	<port> – принимает значение [1..65535]. Можно указать несколько портов перечислением через запятую «,» либо указать диапазон портов через «-».
28	Создать списки приложений, которые будут использоваться в механизме DPI.	<code>rtt (config) # object-group application <NAME></code>	<NAME> – имя профиля приложений, задается строкой до 31 символа.
29	Задать описание списка приложений (необязательно).	<code>rtt (config-object-group-application) # description <description></code>	<description> – описание профиля, задается строкой до 255 символов.
30	Внести необходимые приложения в списки.	<code>rtt (config-object-group-application) # application <APPLICATION ></code>	<APPLICATION> – указывает приложение, попадающее под действие данного профиля.
31	Создать список доменных имен, которые будут использоваться при фильтрации.	<code>rtt (config) # object-group domain-name <NAME></code>	<NAME> – имя профиля доменных имен, задается строкой до 31 символа.
32	Задать описание списка доменных имен (необязательно).	<code>rtt (config-object-group-domain-name) # description <description></code>	<description> – описание профиля, задается строкой до 255 символов.
33	Внести необходимые доменные имена в списки.	<code>rtt (config-object-group-domain-name) # domain <DOMAIN></code>	<DOMAIN> – доменное имя, строка длиной от 1 до 253 символов.
34	Включить интерфейсы (физические, логические, E1/Multilink и подключаемые), сервер удаленного доступа (l2tp, openvpn, pptp) или туннели (gre, ip4ip4, l2tp, lt, rppoe, pptp) в зоны безопасности (если необходимо).	<code>rtt (config-if-gi) # security-zone <zone-name></code>	<zone-name> – до 12 символов.
	Отключить функции Firewall на сетевом интерфейсе (физические, логические, E1/Multilink и подключаемые), сервере удаленного доступа (l2tp, openvpn, pptp) или туннели (gre, ip4ip4, l2tp, lt, rppoe, pptp) (если необходимо).	<code>rtt (config-if-gi) # ip firewall disable</code>	

Шаг	Описание	Команда	Ключи
	Отключить функции Firewall глобально на всех сетевых сущностях (если необходимо).	<code>rtt(config)# ip firewall disable</code>	
35	Создать набор правил межзонового взаимодействия. 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». Очерёдность обработки трафика для разных zone-pair описана в примечании.	<code>rtt(config)# security zone-pair <src-zone- name1> <dst-zone-name2></code>	<src-zone-name> – до 12 символов. <dst-zone-name> – до 12 символов.
36	Создать правило межзонового взаимодействия.	<code>rtt(config-security- zone-pair)# rule <rule- number></code>	<rule-number> – 1..10000.
37	Задать описание правила (необязательно).	<code>rtt(config-security- zone-pair)# description <description></code>	<description> – до 255 символов.
38	Установить имя или номер IP-протокола, для которого должно срабатывать правило (необязательно).	<code>rtt(config-security- zone-pair-rule)# match [not] protocol <protocol-type></code>	<protocol-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.
		<code>rtt(config-security- zone-pair-rule)# match [not] protocol-id <protocol-id></code>	<protocol-id> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].

Шаг	Описание	Команда	Ключи
39	Установить IP-адрес отправителя, для которых должно срабатывать правило (необязательно).	<pre>rtt(config-security-zone-pair-rule)# match [not] source-address { address-range { <ADDR>[-<ADDR>] <IPV6-ADDR>[-<IPV6- ADDR>] } prefix { <ADDR/LEN> <IPv6- ADDR/LEN> } object-group { network <OBJ-GROUP-NETWORK- NAME> domain-name <OBJ-GROUP-DOMAIN-NAME> } any }</pre>	<p>address-range <ADDR>[-<ADDR>] – диапазон IP-адресов для правил firewall. Если не указывать IP-адрес конца диапазона, то в качестве IP-адреса для срабатывания правила используется только IP-адрес начала диапазона.</p> <p>Параметр задаётся в виде A.B.C.D, 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];</p>
40	Установить IP-адрес получателя, для которых должно срабатывать правило (необязательно).	<pre>rtt(config-security-zone-pair-rule)# match [not] destination- address { address-range { <ADDR>[-<ADDR>] <IPV6-ADDR>[-<IPV6- ADDR>] } prefix { <ADDR/LEN> <IPv6- ADDR/LEN> } object-group { network <OBJ-GROUP-NETWORK- NAME> domain-name <OBJ-GROUP-DOMAIN-NAME> } any }</pre>	<p>prefix <ADDR/LEN> – IP-подсеть, используемая для срабатывания правила фильтрации firewall. Параметр задаётся в виде A.B.C.D/E, где каждая часть A – D принимает значения [0..255] и E принимает значения [1..32]; <IPv6-ADDR/LEN> – IPv6-адрес, задаётся в виде X:X:X:X::X/E, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и E принимает значения [1..128];</p> <p>object-group network <OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа;</p> <p>object-group domain-name <OBJ-GROUP-DOMAIN-NAME> – имя профиля доменных имен, задаётся строкой до 31 символа.</p> <p>При указании значения any правило будет срабатывать для любого IP-адреса получателя.</p>
41	Установить MAC-адрес отправителя, для которого должно срабатывать правило (необязательно).	<pre>rtt(config-security-zone-pair-rule)# match [not] source-mac {<mac- addr> <OBJ-GROUP- NAME>}</pre>	<p><mac-addr> – задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].</p>
42	Установить MAC-адрес получателя, для которого должно срабатывать правило (необязательно).	<pre>rtt(config-security-zone-pair-rule)# match [not] destination- mac {<mac-addr> <OBJ- GROUP-NAME>}</pre>	<p><OBJ-GROUP-NAME> – имя профиля MAC-адресов, задаётся строкой до 31 символа.</p>

Шаг	Описание	Команда	Ключи
43	Установить TCP/UDP-порт отправителя, для которого должно срабатывать правило (если указан протокол).	<code>rtt (config-security-zone-pair-rule)# match [not] source-port <TYPE> {<PORT-SET-NAME> <FROM-PORT> - <TO-PORT>}</code>	<p><TYPE> – тип аргумента, устанавливаемый в качестве порта:</p> <ul style="list-style-type: none"> • object-group – указать имя профиля; • port-range – указать диапазон портов; • any – установить в качестве порта любой порт.
44	Установить TCP/UDP-порт получателя, для которого должно срабатывать правило (если указан протокол).	<code>rtt (config-security-zone-pair-rule)# match [not] destination-port <TYPE> {<PORT-SET-NAME> <FROM-PORT> - <TO-PORT>}</code>	<p><PORT-SET-NAME> – задаётся строкой до 31 символа;</p> <p><FROM-PORT> – начальный порт диапазона;</p> <p><TO-PORT> – конечный порт диапазона.</p>
45	Установить профиль приложений, который будет использоваться в механизме DPI.	<code>rtt (config-security-zone-pair-rule)# match [not] application <OBJ-GROUP-NAME></code>	<OBJ-GROUP-NAME> – имя профиля приложений, задаётся строкой до 31 символа.
46	Установить тип и код сообщений протокола ICMP, для которых должно срабатывать правило (если в качестве протокола выбран ICMP) (необязательно).	<code>rtt (config-security-zone-pair-rule)# match [not] icmp <ICMP_TYPE> <ICMP_CODE></code>	<p><ICMP_TYPE> – тип сообщения протокола ICMP, принимает значения [0..255];</p> <p><ICMP_CODE> – код сообщения протокола ICMP, принимает значения [0..255]. При указании значения «any» правило будет срабатывать для любого кода сообщения протокола ICMP.</p>
47	Установить тип и код сообщений протокола ICMPv6, для которых должно срабатывать правило (если в качестве протокола выбран ICMP) (необязательно).	<code>rtt (config-security-zone-pair-rule)# match [not] icmpv6 <ICMP_TYPE> <ICMP_CODE></code>	<p><ICMP_TYPE> – тип сообщения протокола ICMPv6, принимает значения [0..255];</p> <p><ICMP_CODE> – код сообщения протокола ICMPv6, принимает значения [0..255]. При указании значения «any» правило будет срабатывать для любого кода сообщения протокола ICMP.</p>
48	Установить ограничение, при котором правило будет срабатывать только для трафика, измененного сервисом трансляции IP-адресов и портов получателя.	<code>rtt (config-security-zone-pair-rule)# match [not] destination-nat</code>	

Шаг	Описание	Команда	Ключи
49	Установить фильтрацию только для фрагментированных IP-пакетов (необязательно, доступно только для zone-pair any self и zone-pair <zone-name> any).	<code>rtt(config-security-zone-pair-rule)# match [not] fragment</code>	
50	Установить фильтрацию для IP-пакетов, содержащих ip-option (необязательно, доступно только для zone-pair any self и zone-pair <zone-name> any).	<code>rtt(config-security-zone-pair-rule)# match [not] ip-option</code>	
51	Включить правило межзонового взаимодействия.	<code>rtt(config-security-zone-pair-rule)# enable</code>	
52	Активировать фильтрацию и режим отслеживания сессий при прохождении пакетов между участниками одной Bridge-группы (необязательно).	<code>rtt(config-bridge)# ports firewall enable</code>	

Порядок обработки транзитного трафика правилами firewall

1. Трафик проверяется правилами zone-pair user-zone any.
Если трафик не попал ни под одно из правил текущей zone-pair, переходим к следующему шагу.
2. Если трафик передаётся с одного интерфейса на другой в пределах одной зоны (user-zone), то он проверяется правилами zone-pair user-zone user-zone.
Если трафик не попал ни под одно из правил текущей zone-pair, переходим к следующему шагу.
3. Если трафик передаётся с одного интерфейса на другой в разных зонах, то он проверяется правилами zone-pair user-zone1 user-zone2.
Если трафик не попал ни под одно из правил текущей zone-pair, переходим к следующему шагу.
4. Трафик проверяется правилами zone-pair any any.
Если трафик не попал ни под одно из правил текущей zone-pair, он отбрасывается.

Порядок обработки трафика, терминируемого на маршрутизаторе

1. Трафик проверяется правилами zone-pair any self.
Если трафик не попал ни под одно из правил текущей zone-pair, переходим к следующему шагу.
2. Трафик проверяется правилами zone-pair user-zone self.
Если трафик не попал ни под одно из правил текущей zone-pair, он отбрасывается.

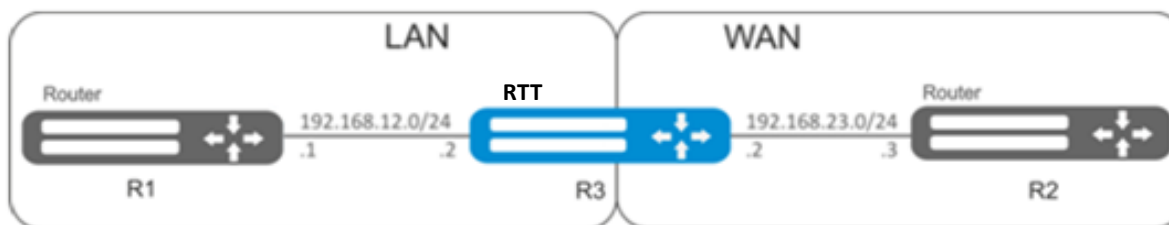
Каждая команда «match» может содержать ключ «not». При использовании данного ключа под правило будут подпадать пакеты, не удовлетворяющие заданному критерию.

Более подробная информация о командах для настройки межсетевого экрана содержится в справочнике команд CLI.

14.4.2. Пример настройки Firewall

Задача:

Разрешить обмен сообщениями по протоколу ICMP между устройствами R1, R2 и маршрутизатором RTT.



Решение:

Для каждой сети RTT создадим свою зону безопасности:

```
rtt# configure
rtt(config)# security zone LAN
rtt(config-security-zone)# exit
rtt(config)# security zone WAN
rtt(config-security-zone)# exit
```

Настроим сетевые интерфейсы и определим их принадлежность к зонам безопасности:

```
rtt(config)# interface gi1/0/2
rtt(config-if-gi)# ip address 192.168.12.2/24
rtt(config-if-gi)# security-zone LAN
rtt(config-if-gi)# exit
rtt(config)# interface gi1/0/3
rtt(config-if-gi)# ip address 192.168.23.2/24
rtt(config-if-gi)# security-zone WAN
rtt(config-if-gi)# exit
```

Для настройки правил зон безопасности потребуется создать профиль адресов сети «LAN», включающий адреса, которым разрешен выход в сеть «WAN», и профиль адресов сети «WAN».

```
rtt(config)# object-group network WAN
rtt(config-object-group-network)# ip address-range 192.168.23.2
rtt(config-object-group-network)# exit
rtt(config)# object-group network LAN
rtt(config-object-group-network)# ip address-range 192.168.12.2
rtt(config-object-group-network)# exit
rtt(config)# object-group network LAN_GATEWAY
rtt(config-object-group-network)# ip address-range 192.168.12.1
rtt(config-object-group-network)# exit
```

```
rtt(config)# object-group network WAN_GATEWAY
rtt(config-object-group-network)# ip address-range 192.168.23.3
rtt(config-object-group-network)# exit
```

Для пропуска трафика из зоны «LAN» в зону «WAN» создадим пару зон и добавим правило, разрешающее проходить ICMP-трафику от R1 к R2. Действие правил разрешается командой **enable**:

```
rtt(config)# security zone-pair LAN WAN
rtt(config-security-zone-pair)# rule 1
rtt(config-security-zone-pair-rule)# action permit
rtt(config-security-zone-pair-rule)# match protocol icmp
rtt(config-security-zone-pair-rule)# match destination-address object-group
network WAN_GATEWAY
rtt(config-security-zone-pair-rule)# match source-address object-group network
LAN_GATEWAY
rtt(config-security-zone-pair-rule)# enable
rtt(config-security-zone-pair-rule)# exit
rtt(config-security-zone-pair)# exit
```

Для пропуска трафика из зоны «WAN» в зону «LAN» создадим пару зон и добавим правило, разрешающее проходить ICMP-трафику от R2 к R1. Действие правил разрешается командой **enable**:

```
rtt(config)# security zone-pair WAN LAN
rtt(config-security-zone-pair)# rule 1
rtt(config-security-zone-pair-rule)# action permit
rtt(config-security-zone-pair-rule)# match protocol icmp
rtt(config-security-zone-pair-rule)# match destination-address object-group
network LAN_GATEWAY
rtt(config-security-zone-pair-rule)# match source-address object-group network
WAN_GATEWAY
rtt(config-security-zone-pair-rule)# enable
rtt(config-security-zone-pair-rule)# exit
rtt(config-security-zone-pair)# exit
```

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». Создадим пару зон для трафика, идущего из зоны «WAN» в зону «self». Добавим правило, разрешающее проходить ICMP-трафику между R2 и маршрутизатором RTT, для того чтобы маршрутизатор начал отвечать на ICMP-запросы из зоны «WAN»:

```
rtt(config)# security zone-pair WAN self
rtt(config-security-zone-pair)# rule 1
rtt(config-security-zone-pair-rule)# action permit
rtt(config-security-zone-pair-rule)# match protocol icmp
rtt(config-security-zone-pair-rule)# match destination-address object-group
network WAN
rtt(config-security-zone-pair-rule)# match source-address object-group network
WAN_GATEWAY
rtt(config-security-zone-pair-rule)# enable
rtt(config-security-zone-pair-rule)# exit
rtt(config-security-zone-pair)# exit
```

Создадим пару зон для трафика, идущего из зоны «LAN» в зону «self». Добавим правило, разрешающее проходить ICMP-трафику между R1 и RTT, для того чтобы маршрутизатор начал отвечать на ICMP-запросы из зоны «LAN»:

```

rtt(config)# security zone-pair LAN self
rtt(config-security-zone-pair)# rule 1
rtt(config-security-zone-pair-rule)# action permit
rtt(config-security-zone-pair-rule)# match protocol icmp
rtt(config-security-zone-pair-rule)# match destination-address network object-
group LAN
rtt(config-security-zone-pair-rule)# match source-address object-group network
LAN_GATEWAY
rtt(config-security-zone-pair-rule)# enable
rtt(config-security-zone-pair-rule)# exit
rtt(config-security-zone-pair)# exit
rtt(config)# exit

```

Посмотреть членство портов в зонах можно с помощью команды:

```
rtt# show security zone
```

Посмотреть пары зон и их конфигурацию можно с помощью команд:

```

rtt# show security zone-pair
rtt# show security zone-pair configuration

```

Посмотреть активные сессии можно с помощью команд:

```
rtt# show ip firewall sessions
```

14.4.3. Пример настройки Firewall по доменным именам

Задача:

Фиксировать обращения на ресурсы компании Yandex из локальной сети в Syslog:



Решение:

Создадим зоны безопасности для локальной сети и uplink-интерфейса в сторону интернет-провайдера:

```

rtt# configure
rtt(config)# security zone LAN
rtt(config-security-zone)# exit
rtt(config)# security zone WAN
rtt(config-security-zone)# exit

```

Настроим сетевые интерфейсы и определим их принадлежность к зонам безопасности:

```
rtt(config)# interface gil/0/1
rtt(config-if-gi)# ip address 10.0.0.1/24
rtt(config-if-gi)# security-zone WAN
rtt(config-if-gi)# exit
rtt(config)# interface gil/0/2
rtt(config-if-te)# ip address 192.168.0.1/24
rtt(config-if-te)# security-zone LAN
rtt(config-if-te)# exit
```

Настроим маршрут по умолчанию в сторону интернет-провайдера:

```
rtt(config)# ip route 0.0.0.0/0 10.0.0.254
```

Настроим систему разрешения доменных имен, её настройка необходима для работы Firewall по доменным именам:

```
rtt(config)# domain lookup enable
rtt(config)# domain nameserver 10.0.0.254
```

Настроим простую конфигурацию Source-NAT для любого транзитного через RTT трафика:

```
rtt(config)# nat source
rtt(config-snat)# ruleset SNAT
rtt(config-snat-ruleset)# to zone WAN
rtt(config-snat-ruleset)# rule 1
rtt(config-snat-rule)# action source-nat interface
rtt(config-snat-rule)# enable
rtt(config-snat-rule)# exit
rtt(config-snat-ruleset)# exit
rtt(config-snat)# exit
```

Создадим профиль доменных имен, включающий в себя популярные домены компании "Яндекс":

RTT также позволяет работать с интернационализированными доменными именами (домены, использующие символы национальных алфавитов). Для использования интернационализованного доменного имени в конфигурации RTT его нужно преобразовать к виду ASCII-compatible encoding при помощи любого Punycode-преобразователя. Т. е. получим:

яндекс.рф → xn--d1acpјx3f.xn--p1ai

президент.рф → xn--d1abbgf6aiіy.xn--p1ai

И уже ASCII-compatible encoding вариант доменного имени можно указывать в конфигурации RTT.

```
rtt(config)# object-group domain-name YANDEX
rtt(config-object-group-domain-name)# domain ya.ru
rtt(config-object-group-domain-name)# domain yandex.ru
rtt(config-object-group-domain-name)# domain dzen.ru
rtt(config-object-group-domain-name)# domain xn--d1acpјx3f.xn--p1ai
rtt(config-object-group-domain-name)# exit
```

Разрешим прохождение трафика из зоны «LAN» в зону «WAN» и отдельно создадим правило, которое будет фиксировать в Syslog обращения на домены компании "Яндекс":

```
rtt(config)# security zone-pair LAN WAN
rtt(config-security-zone-pair)# rule 1
rtt(config-security-zone-pair-rule)# action permit log
rtt(config-security-zone-pair-rule)# match destination-address object-group
domain-name YANDEX
rtt(config-security-zone-pair-rule)# enable
rtt(config-security-zone-pair-rule)# exit
rtt(config-security-zone-pair)# rule 2
rtt(config-security-zone-pair-rule)# action permit
rtt(config-security-zone-pair-rule)# enable
rtt(config-security-zone-pair-rule)# exit
rtt(config-security-zone-pair)# exit
```

Посмотреть членство портов в зонах можно с помощью команды:

```
rtt# show security zone
```

Посмотреть пары зон и их конфигурацию можно с помощью команд:

```
rtt# show security zone-pair
rtt# show security zone-pair configuration
```

Посмотреть активные сессии можно с помощью команд:

```
rtt# show ip firewall sessions
```

Посмотреть кэш DNS можно с помощью команд:

```
rtt# show dns records
rtt# show dns records negative
```

Пример сообщения Syslog при срабатывании правила:

```
<190>1 2025-07-07T17:40:10+07:00 rtt firewalld - - - %FIREWALL-I-LOG: zone-pair
'LAN WAN' rule 1 permitted tcp 192.168.0.21:45574 (gil/0/2) -> 77.88.44.55:443
dscp 48
```

14.4.4. Пример настройки фильтрации приложений (DPI)



Использование механизма фильтрации приложений многократно снижает производительность маршрутизатора из-за необходимости проверки каждого пакета. Производительность снижается с ростом количества выбранных приложений для фильтрации.

Задача:

Блокировать доступ пользователей в локальной сети к Telegram, Facebook Messenger и Skype.



Решение:

Для каждой сети RTT создадим свою зону безопасности:

```
rtt# configure
rtt(config)# security zone LAN
rtt(config-security-zone)# exit
rtt(config)# security zone WAN
rtt(config-security-zone)# exit
```

Настроим сетевые интерфейсы и определим их принадлежность к зонам безопасности:

```
rtt(config)# interface gi1/0/1
rtt(config-if-gi)# ip address 10.0.0.1/24
rtt(config-if-gi)# security-zone WAN
rtt(config-if-gi)# exit
rtt(config)# interface gi1/0/2
rtt(config-if-te)# ip address 192.168.0.1/24
rtt(config-if-te)# security-zone LAN
rtt(config-if-te)# exit
```

Для настройки правил зон безопасности потребуется создать профиль приложений, которые необходимо будет блокировать:

```
rtt(config)# object-group application BLACKLIST
rtt(config-object-group-application)# application telegram
rtt(config-object-group-application)# application facebook-messenger
rtt(config-object-group-application)# application skype-teams
rtt(config-object-group-application)# exit
```

Для установки правил прохождения трафика из зоны «WAN» в зону «LAN» создадим пару зон и добавим правило, запрещающее проходить трафику приложений, и правило, разрешающее проходить остальному трафику. Действие правил разрешается командой **enable**:

```
rtt(config)# security zone-pair WAN LAN
rtt(config-security-zone-pair)# rule 1
rtt(config-security-zone-pair-rule)# action deny
rtt(config-security-zone-pair-rule)# match application BLACKLIST
rtt(config-security-zone-pair-rule)# enable
rtt(config-security-zone-pair-rule)# exit
rtt(config-security-zone-pair)# rule 2
rtt(config-security-zone-pair-rule)# action permit
rtt(config-security-zone-pair-rule)# enable
rtt(config-security-zone-pair-rule)# exit
rtt(config-security-zone-pair)# exit
```


Для установки правил прохождения трафика из зоны «LAN» в зону «WAN» создадим пару зон и добавим правило, запрещающее прохождение трафика приложений, и правило, разрешающее прохождение всего остального трафика. Действие правил разрешается командой **enable**:

```
rtt(config)# security zone-pair LAN WAN
rtt(config-security-zone-pair)# rule 1
rtt(config-security-zone-pair-rule)# action deny
rtt(config-security-zone-pair-rule)# match application BLACKLIST
rtt(config-security-zone-pair-rule)# enable
rtt(config-security-zone-pair-rule)# exit
rtt(config-security-zone-pair)# rule 2
rtt(config-security-zone-pair-rule)# action permit
rtt(config-security-zone-pair-rule)# enable
rtt(config-security-zone-pair-rule)# exit
rtt(config-security-zone-pair)# exit
```

Посмотреть членство портов в зонах можно с помощью команды:

```
rtt# show security zone
```

Посмотреть пары зон и их конфигурацию можно с помощью команд:

```
rtt# show security zone-pair
rtt# show security zone-pair configuration
```

Посмотреть активные сессии можно с помощью команд:

```
rtt# show ip firewall sessions
```

14.5. Настройка списков доступа (ACL)

Access Control List или ACL — список контроля доступа, содержит правила, определяющие прохождение трафика через интерфейс.

14.5.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать список контроля доступа и перейти в режим его конфигурирования.	<code>rtt(config)# ip access-list extended <NAME></code>	<NAME> — имя создаваемого списка контроля доступа, задаётся строкой до 31 символа.
2	Указать описание конфигурируемого списка контроля доступа (необязательно).	<code>rtt(config-acl)# description <DESCRIPTION></code>	<DESCRIPTION> — описание списка контроля доступа, задаётся строкой до 255 символов.

Шаг	Описание	Команда	Ключи
3	Создать правило и перейти в режим его конфигурирования. Правила обрабатываются маршрутизатором в порядке возрастания их номеров.	<code>rtt(config-acl)# rule <ORDER></code>	<ORDER> – номер правила, принимает значения [1...4094].
4	Указать действие, которое должно быть применено для трафика, удовлетворяющего заданным критериям.	<code>rtt(config-acl-rule)# action <ACT></code>	<ACT> – назначаемое действие: <ul style="list-style-type: none"> • permit – прохождение трафика разрешается; • deny – прохождение трафика запрещается.
5	Установить имя/номер протокола, для которого должно срабатывать правило (необязательно).	<code>rtt(config-acl-rule)# match protocol <TYPE></code>	<TYPE> – тип протокола, принимает значения: esp, icmp, icmp6, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rsvp, l2tp, gre. При указании значения «any» правило будет срабатывать для любых протоколов.
		<code>rtt(config-acl-rule)# match protocol-id <ID></code>	<ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].
6	Установить IP-адреса отправителя, для которых должно срабатывать правило (необязательно).	<code>rtt(config-acl-rule)# match source-address { <ADDR> <MASK> any }</code>	<ADDR> – IP-адрес отправителя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];
7	Установить IP-адреса получателя, для которых должно срабатывать правило (необязательно).	<code>rtt(config-acl-rule)# match destination-address { <ADDR> <MASK> any }</code>	<MASK> – 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/получателя.
8	Установить MAC-адреса отправителя, для которых должно срабатывать правило (необязательно).	<code>rtt(config-acl-rule)# match source-mac <ADDR><WILDCARD></code>	<ADDR> – MAC-адрес отправителя, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];
9	Установить MAC-адреса получателя, для которых должно срабатывать правило (необязательно).	<code>rtt(config-acl-rule)# match destination-mac <ADDR><WILDCARD></code>	<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.

Шаг	Описание	Команда	Ключи
10	Установить номер TCP/UDP-порта отправителя, для которого должно срабатывать правило (если указан протокол).	<code>rtt(config-acl-rule)# match source-port <TYPE> {<FROM-PORT> - <TO-PORT> <PORT>}</code>	<p><TYPE> – тип аргумента, устанавливаемый в качестве порта:</p> <ul style="list-style-type: none"> port-range – указать диапазон портов; any – установить в качестве порта любой порт. <p><FROM-PORT> – начальный порт диапазона;</p> <p><TO-PORT> – конечный порт диапазона;</p> <p><PORT> – указание единичного порта.</p>
11	Установить номер TCP/UDP-порта получателя, для которого должно срабатывать правило (если указан протокол).	<code>rtt(config-acl-rule)# match destination-port <TYPE> {<FROM-PORT> - <TO-PORT> <PORT>}</code>	
12	Установить значение 802.1p приоритета, для которого должно срабатывать правило (необязательно).	<code>rtt(config-acl-rule)# match cos <COS></code>	<COS> – значение 802.1p приоритета, принимает значения [0..7].
13	Установить значение кода DSCP, для которого должно срабатывать правило (необязательно). Невозможно использовать совместно с IP Precedence.	<code>rtt(config-acl-rule)# match dscp <DSCP></code>	<DSCP> – значение кода DSCP, принимает значения [0..63].
14	Установить значение кода IP Precedence, для которого должно срабатывать правило (необязательно). Невозможно использовать совместно с DSCP.	<code>rtt(config-acl-rule)# match ip-precedence <IPP></code>	<IPP> – значение кода IP Precedence, принимает значения [0..7].
15	Установить значение идентификационного номера VLAN, для которого должно срабатывать правило (необязательно).	<code>rtt(config-acl-rule)# match vlan <VID></code>	<VID> – идентификационный номер VLAN, принимает значения [1..4094].
16	Активировать правило.	<code>rtt(config-acl-rule)# enable</code>	
17	Указать список контроля доступа к конфигурируемому интерфейсу для фильтрации входящего трафика.	<code>rtt(config-if-gi)# service-acl input <NAME></code>	<NAME> – имя списка контроля доступа, задаётся строкой до 31 символа.
18	Указать список контроля доступа к конфигурируемому интерфейсу для фильтрации исходящего трафика.	<code>rtt(config-if-gi)# service-acl output <NAME></code>	<NAME> – имя списка контроля доступа, задаётся строкой до 31 символа.

Также списки доступа могут использоваться для организации политик QoS.

14.5.2. Пример настройки списка доступа

Задача:

Разрешить прохождения трафика только из подсети 192.168.20.0/24.

Решение:

Настроим список доступа для фильтрации по подсетям:

```
rtt# configure
rtt(config)# ip access-list extended white
rtt(config-acl)# rule 1
rtt(config-acl-rule)# action permit
rtt(config-acl-rule)# match source-address 192.168.20.0 255.255.255.0
rtt(config-acl-rule)# enable
rtt(config-acl-rule)# exit
rtt(config-acl)# exit
```

Применим список доступа на интерфейс Gi1/0/19 для входящего трафика:

```
rtt(config)# interface gigabitethernet 1/0/19
rtt(config-if-gi)# service-acl input white
```

Просмотреть детальную информацию о списке доступа возможно через команду:

```
rtt# show ip access-list white
```

14.6. Проксирование HTTP/HTTPS-трафика

14.6.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать объект с URL.	<code>rtt(config)# object-group url <NAME></code>	
2	Указать набор.	<code>rtt(config-object-group-url)# url <URL></code>	<URL> – адрес веб-страницы, сайта.
3	Создать профиль проксирования.	<code>rtt(config)# ip http profile <NAME></code>	<NAME> – название профиля.
4	Выбрать действие по умолчанию.	<code>rtt(config-profile)# default action { deny permit redirect } [redirect-url <URL>]</code>	<URL> – адрес хоста, на который будут передаваться запросы.
5	Указать описание (необязательно).	<code>rtt(config-profile)# description <description></code>	<description> – до 255 символов.

Шаг	Описание	Команда	Ключи
6	Указать режим фильтрации данных (необязательно).	<code>rtt(config-profile)# filter <DATA-TYPE></code>	<p><DATA-TYPE> – тип данных, подлежащих фильтрации. Может принимать значения (как одно, так и несколько):</p> <ul style="list-style-type: none"> • activex – заблокировать все приложения ActiveX; • cookie – запретить веб-сайтам размещать cookie на пользовательских компьютерах; • js – заблокировать все страницы или приложения на основе Javascript.
7	Указать удаленный или локальный список URL и тип операции (блокировка/пропуск трафика/перенаправление) (необязательно).	<code>rtt(config-profile)# urls { local remote } <URL_OBJ_GROUP_NAME> action { deny permit redirect } [redirect- url <URL>]</code>	<URL_OBJ_GROUP_NAME> – указать название объекта, содержащего набор URL.
8	Указать удаленный сервер, где лежат необходимые списки URL (необязательно).	<code>rtt(config)# ip http proxy server-url <URL></code>	<URL> – адрес сервера, откуда будут брать удалённые списки url.
9	Указать прослушиваемый порт для проксирования http (необязательно).	<code>rtt(config)# ip http proxy listen-ports <OBJ_GROUP_NAME></code>	<p><OBJ_GROUP_NAME> – имя профиля порта, задаётся строкой до 31 символа.</p> <p>По умолчанию прослушиваются порты 80 и 8080</p>
10	Указать прослушиваемый порт для проксирования (необязательно).	<code>rtt(config)# ip https proxy listen-ports <OBJ_GROUP_NAME></code>	<p><OBJ_GROUP_NAME> – имя профиля порта, задаётся строкой до 31 символа.</p> <p>По умолчанию прослушивается порт 443</p>
11	Указать базовый порт для проксирования http (необязательно).	<code>rtt(config)# ip http proxy redirect-port <PORT></code>	<p><PORT> – номер порта, указывается в диапазоне [1..65535].</p> <p>Значение по умолчанию 3128.</p>
12	Указать базовый порт для проксирования https (необязательно).	<code>rtt(config)# ip http proxy redirect-port <PORT></code>	<p><PORT> – номер порта, указывается в диапазоне [1..65535].</p> <p>Значение по умолчанию 3128.</p>
13	Включить проксирование на интерфейсе на основе выбранного HTTP-профиля.	<code>rtt(config-if)# ip http proxy <PROFILE_NAME></code>	<PROFILE_NAME> – название профиля.

Шаг	Описание	Команда	Ключи
14	Включить проксирование на интерфейсе на основе выбранного HTTPS-профиля.	<code>rtt(config-if)# ip https proxy <PROFILE_NAME></code>	<PROFILE_NAME> – название профиля.
15	Создать списки сервисов, которые будут использоваться при фильтрации.	<code>rtt(config)# object-group service <obj-group-name></code>	<obj-group-name> – имя профиля сервисов, задается строкой до 31 символа.
16	Задать описание списка сервисов (необязательно).	<code>rtt(config-object-group-service)# description <description></code>	<description> – описание профиля, задается строкой до 255 символов.
17	Внести необходимые сервисы (TCP/UDP-порты) в список.	<code>rtt(config-object-group-service)# port-range 3128-3135</code>	<p>Прокси-сервер RTT использует для своей работы порты, начиная с базового порта, определённого на 10 шаге.</p> <p>Для http проху используются порты, начиная с базового порта по базовый порт + количество сри данной модели RTT - 1.</p> <p>Для https проху используются порты, начиная с базового порта + количество сри данной модели RTT по базовый порт + количество сри данной модели RTT * 2 - 1.</p> <p>Количество CPU можно посмотреть с помощью команды show cpu utilization.</p>
18	Создать набор правил межзонового взаимодействия.	<code>rtt(config)# security zone-pair <src-zone-name> self</code>	<p><src-zone-name> – зона безопасности, в которой находятся интерфейсы с функцией ip http проху или ip https проху.</p> <p>self – предопределенная зона безопасности для трафика, поступающего на сам RTT.</p>
19	Создать правило межзонового взаимодействия.	<code>rtt(config-zone-pair)# rule <rule-number></code>	<rule-number> – 1..10000.
20	Задать описание правила (необязательно).	<code>rtt(config-zone-rule)# description <description></code>	<description> – до 255 символов.
21	Указать действие данного правила.	<code>rtt(config-zone-rule)# action <action> [log]</code>	<p><action> – permit.</p> <p>log – ключ для активации логирования сессий, которые устанавливаются согласно данному правилу.</p>

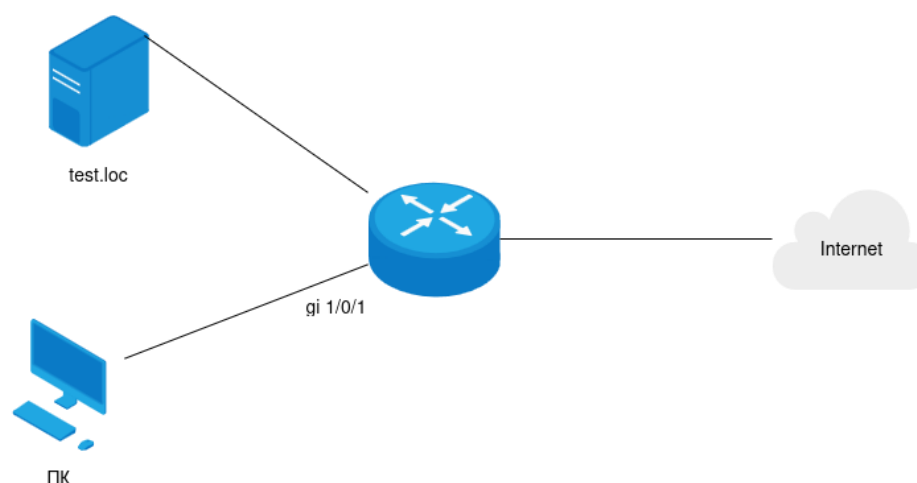
Шаг	Описание	Команда	Ключи
22	Установить имя IP-протокола, для которого должно срабатывать правило.	<code>rtt(config-zone-rule) # match protocol <protocol-type></code>	<protocol-type> – tcp. Прокси-сервер RTT работает по протоколу RTT.
23	Установить профиль TCP/UDP-портов получателя, для которых должно срабатывать правило (если указан протокол).	<code>rtt(config-zone-rule) # match [not] destination-port <obj-group-name></code>	<obj-group-name> – имя профиля сервисов, созданного на шаге 12.
24	Включить правило межзонового взаимодействия.	<code>rtt(config-zone-rule) # enable</code>	

Если функция Firewall на RTT принудительно не отключена, необходимо создать разрешающее правило для зоны Self.

14.6.2. Пример настройки HTTP-прокси

Задача №1:

Организовать фильтрацию по URL для ряда адресов посредством прокси.



Решение:

Создадим набор URL, по которым будет осуществляться фильтрация. Настроим прокси-фильтр и укажем действия для созданного набора URL:

```

rtt# configure
rtt(config)# object-group url FILTER_OG
rtt(config-object-group-url) # regexp *speedtest.net/
rtt(config-object-group-url) # url http://ya.ru/
rtt(config-object-group-url) # url https://ya.ru/
rtt(config-object-group-url) # exit
  
```

Создаем профиль, где указываем действие для всех URL и действие для созданной группы URL:

```
rtt(config)# ip http profile PROXY_LIST
rtt(config-profile)# default action permit
rtt(config-profile)# urls local FILTER_OG action redirect redirect-url
http://test.loc
rtt(config-profile)# exit
```

Включим проксирование на интерфейсе по профилю 'PROXY_LIST':

```
rtt(config)# interface gi 1/0/1
rtt(config-if)# ip http proxy PROXY_LIST
rtt(config-if)# ip https proxy PROXY_LIST
```

Если используется Firewall, создадим для него разрешающие правила. Для этого:

Определим число CPU, доступных для данной модели:

```
rtt(config)# do show cpu utilization
```

CPU	Last 5 sec	Last 1 min	Last 5 min
0	3.79%	1.61%	1.55%
1	0.00%	0.00%	0.01%
2	0.00%	0.00%	0.01%
3	0.00%	0.02%	0.01%

Соответственно по формуле, описанной на 17 шаге алгоритма настройки http/https-прокси, получаем:

Для http проху необходимо открыть порты с 3128 по 3131 ($3128+4-1=3131$).

Для https проху необходимо открыть порты с 3132 по 3135 ($3128+4=3132$, $3128+2*4-1=3135$).

Создаем профиль портов прокси-сервера:

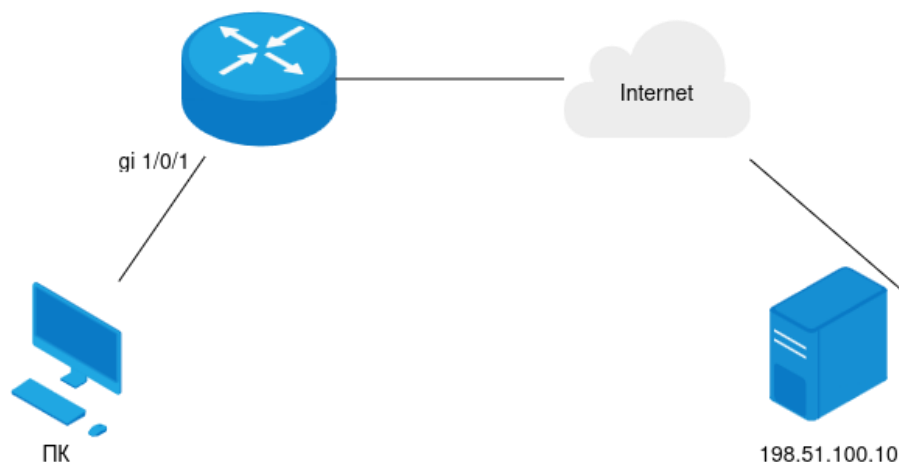
```
rtt(config)# object-group service PROXY_PORTS
rtt(config-object-group-service)# port-range 3128-3135
rtt(config-object-group-service)# exit
```

Создаем разрешающее правило межзонового взаимодействия:

```
rtt(config)# security zone-pair LAN self
rtt(config-zone-pair)# rule 50
rtt(config-zone-pair-rule)# action permit
rtt(config-zone-pair-rule)# match protocol tcp
rtt(config-zone-pair-rule)# match destination-port object-group PROXY_PORTS
rtt(config-zone-pair-rule)# enable
rtt(config-zone-pair-rule)# exit
rtt(config-zone-pair)# exit
```

Задача №2:

Изменить локальный список с URL для фильтрации проху-сервером из задачи №1 на список, получаемый с удалённого сервера.



Решение:

Для использования remote-списка необходимо в конфигурации прописать адрес сервера, а в ip **http profile** изменить **urls local** на **urls remote <list>** – название списка, лежащего на сервере:

```

rtt(config)# ip http proxy server-url http://198.51.100.10
rtt(config)# ip http profile PROXY_LIST
rtt(config-profile)# default action permit
rtt(config-profile)# urls remote URLS_PROXY action deny
rtt(config-profile)# exit
  
```

Задача №3:

Организовать фильтрацию по веб-скриптам ActiveX URL для ряда адресов посредством прокси и настроить логирование событий в консоль.

Решение:

Вы можете настроить Proxy-сервер для фильтрации определённых веб-скриптов. Для фильтрации доступны ActiveX, cookie, JavaScript.

Для блокировки сайтов, использующих ActiveX, настроим фильтрацию и включим логирование событий проху:

```

rtt(config)# ip http profile PROXY_LIST
rtt(config-profile)# default action permit
rtt(config-profile)# urls local FILTER_OG action redirect redirect-url
http://test.loc
rtt(config-profile)# filter activex
rtt(config-profile)# log enable
rtt(config-profile)# exit
  
```

Теперь при срабатывании фильтрации, а также указанных действий для URL, в консоль будут попадать логи вида:

```
%FIREWALL-I-LOG: http proxy 'Filter' (QQ) denied (ActiveX)
%FIREWALL-I-LOG: http proxy 'PROXY_LIST' permitted
```

14.7. Настройка IPS/IDS

Данная функция активируется только при наличии лицензии.

IPS/IDS (Intrusion Prevention System/Intrusion Detection System) – система предотвращения вторжений – программная система сетевой и компьютерной безопасности, обнаруживающая вторжения или нарушения безопасности и автоматически защищающая от них.

Работа системы основана на сигнатурном анализе трафика. Сигнатуры для систем IPS/IDS принято называть правилами. Устройства RTT позволяют скачивать актуальные правила с открытых источников в сети Интернет или с корпоративного сервера. Также с помощью CLI можно создавать свои специфические правила.

14.7.1. Алгоритм базовой настройки

Шаг	Описание	Команда	Ключи
1	Создать политику безопасности IPS/IDS.	<code>rtt(config)# security ips policy <NAME></code>	<NAME> – имя политики безопасности, задаётся строкой до 32 символов.
2	Задать описание политики (необязательно).	<code>rtt(config-ips-policy)# description <DESCRIPTION></code>	<DESCRIPTION> – описание задаётся строкой до 255 символов.
3	Задать профиль IP-адресов, которые будет защищать IPS/IDS.	<code>rtt(config-ips-policy)# protect network-group <OBJ-GROUP-NETWORK_NAME></code>	<OBJ-GROUP-NETWORK-NAME> – имя профиля защищаемых IP-адресов, задается строкой до 32 символов.
4	Задать профиль IP-адресов, внешних для IPS/IDS (необязательно).	<code>rtt(config-ips-policy)# external network-group <OBJ-GROUP-NETWORK_NAME></code>	<OBJ-GROUP-NETWORK-NAME> – имя профиля внешних IP-адресов, задается строкой до 32 символов.
5	Перейти в режим конфигурирования IPS/IDS.	<code>rtt(config)# security ips</code>	
6	Назначить политику безопасности IPS/IDS.	<code>rtt(config-ips)# policy <NAME></code>	<NAME> – имя политики безопасности, задаётся строкой до 32 символов.
7	Использовать все ресурсы RTT для IPS/IDS (необязательно).	<code>rtt(config-ips)# perfomance max</code>	По умолчанию для IPS/IDS отдается половина доступных ядер процессора.

Шаг	Описание	Команда	Ключи
8	Задать параметры удаленного сервера для отправки статистики работы сервиса IPS/IDS в формате EVE (elasticsearch) (необязательно).	<pre> rtt(config-ips)# logging remote-server { <ADDR> <IPV6-ADDR> } [<TRANSPORT>] [<PORT>] [source- address { <SRC-ADDR> <IPV6-SRC-ADDR> object- group <NETWORK_OBJ_GROUP_NAME> }] </pre>	<p><ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];</p> <p><TRANSPORT> – протокол передачи данных, по умолчанию – UDP, принимает значения:</p> <ul style="list-style-type: none"> • TCP – передача данных осуществляется по протоколу TCP; • UDP – передача данных осуществляется по протоколу UDP. <p><PORT> – номер TCP/UDP-порта, опциональный параметр, принимает значения [1..65535], по умолчанию 514;</p> <p><SRC-ADDR> – IPv4-адрес маршрутизатора, который будет использоваться в качестве IP-адреса источника в отправляемых syslog-пакетах, по умолчанию – IPv4-адрес интерфейса, с которого отправляются пакеты;</p> <p><IPV6-SRC-ADDR> – IPv6-адрес маршрутизатора, который будет использоваться в качестве IPv6-адреса источника в отправляемых syslog-пакетах, по умолчанию – IPv6-адрес интерфейса, с которого отправляются пакеты;</p> <p><NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве source address.</p>
9	Установить интервал отправки статистики работы сервиса IPS/IDS в формате EVE (elasticsearch) (необязательно).	<pre> rtt(config-ips)# logging update-interval <INTERVAL> </pre>	<p><INTERVAL> – интервал отправки статистики работы сервиса IPS/IDS, задаётся в минутах.</p>

Шаг	Описание	Команда	Ключи
10	Заблокировать передачу трафика при начальной загрузке до запуска сервиса IPS/IDS и загрузки хотя бы одного настроенного или существующего правила (необязательно).	<code>rtt(config-ips)# fail-close enable</code>	
11	Установить размер виртуальных очередей (необязательно).	<code>rtt(config-ips)# queue-limit <QUEUE-LIMIT></code>	<p><QUEUE-LIMIT> – предельное количество пакетов в виртуальной очереди, принимает значения в диапазоне [32..4096]</p> <p>Размер очереди по умолчанию 1024</p>
12	Активировать сервис IPS.	<code>rtt(config-ips)# enable</code>	
13	Активировать IPS/IDS на интерфейсе.	<code>rtt(config-if-gi)# service-ips { inline monitor }</code>	<p>inline – этот режим устанавливается, когда RTT с сервисом IPS/IDS ставится в разрыв сети.</p> <p>monitor – этот режим устанавливается, когда RTT с сервисом IPS/IDS мониторит зеркалируемый трафик.</p>

14.7.2. Алгоритм настройки автообновления правил IPS/IDS из внешних источников

Шаг	Описание	Команда	Ключи
1	Задать имя внешнего хранилища скачиваемых правил (необязательно).	<code>rtt(config-ips)# storage-path { usb://<USB-NAME>:/ mmc://<MMC-NAME>:/ }</code>	<p><USB-NAME> – имя подключенного USB-носителя. Имя можно узнать в выводе команды show storage-devices usb;</p> <p><MMC-NAME> – имя подключенного MMC-носителя. Имя можно узнать в выводе команды show storage-devices mmc.</p> <p>Для использования с системой IPS/IDS на внешнем носителе должен быть создан раздел файловой системы в формате exFAT.</p>
2	Перейти в режим конфигурирования автообновлений.	<code>rtt(config-ips)# auto-upgrade</code>	

3	Задать имя и перейти в режим конфигурирования пользовательского сервера обновлений.	<code>rtt(config-ips-auto-upgrade) # user-server <WORD></code>	<WORD> – имя сервера, задаётся строкой до 32 символов.
4	Задать описание пользовательского сервера обновлений (необязательно).	<code>rtt(config-ips-upgrade-user-server) # description <DESCRIPTION></code>	<DESCRIPTION> – описание задаётся строкой до 255 символов.
5	Задать URL.	<code>rtt(config-ips-upgrade-user-server) # url <URL></code>	<p><URL> – текстовое поле, содержащее URL-ссылку длиной от 8 до 255 символов.</p> <p>В качестве URL-ссылки может быть указан:</p> <ul style="list-style-type: none"> • файл правил с расширением .rule; • файл классификатора правил с именем classification.config; • каталог на сервере, содержащий файлы правил и/или файл классификатора правил.
6	Задать частоту проверки обновлений (необязательно).	<code>rtt(config-ips-upgrade-user-server) # upgrade interval <HOURS></code>	<p><HOURS> – интервал обновлений в часах, от 1 до 240.</p> <p>Значение по умолчанию: 24 часа.</p>
7	Активизировать пользовательский сервер обновлений.	<code>rtt(config-ips-upgrade-user-server) # enable</code>	

Для правил IPS/IDS, загружаемых из внешних источников, на маршрутизаторах RTT выделена отдельная область энергозависимой памяти.

Если настроить слишком много источников правил или загружать правила, превышающие указанные лимиты, то маршрутизатор будет выдавать сообщения об ошибке %STORAGE_IPS_MGR-I-ERR: There no free space in rules directory.

В этом случае стоит уменьшить объем запрашиваемых правил или использовать внешнее хранилище.

14.7.3. Рекомендуемые открытые источники обновления правил

14.7.3.1. SSL Blacklist

Чёрный список SSL (SSLBL) – это проект abuse.ch, целью которого является обнаружение вредоносных SSL-соединений путём идентификации и внесения в чёрный список SSL-сертификатов, используемых серверами управления ботнетами.

https://sslbl.abuse.ch/blacklist/sslblacklist_tls_cert.rules – набор правил SSL-сертификатов от SSLBL используется для обнаружения и/или блокировки вредоносных SSL-соединений в вашей сети на основе отпечатка SSL-сертификата. Набор правил SSL-сертификатов генерируется каждые 5 минут. Рекомендуется запрашивать его не чаще, чем раз в 5 минут.

https://sslbl.abuse.ch/blacklist/ja3_fingerprints.rules – набор правил JA3 FingerprintRuleset от SSLBL используется для обнаружения и/или блокировки вредоносных SSL-соединений в вашей сети на основе отпечатка JA3. Набор правил для отпечатков пальцев Suricata JA3 генерируется каждые 5 минут. Рекомендуется запрашивать его не чаще, чем раз в 5 минут.

Отпечатки JA3, внесённые в чёрный список SSLBL, были собраны путём анализа более 25 000 000 PCAP-файлов с образцами вредоносного ПО. Эти отпечатки ещё не были протестированы на известном безопасном трафике и могут привести к значительному количеству ложных срабатываний.

14.7.3.2. *Feodo Tracker*

Feodo Tracker – это проект abuse.ch, целью которого является обмен информацией о серверах управления ботнетами, связанными с Dridex, Emotet (также известным как Heodo), TrickBot, QakBot (также известным как QuakBot/Qbot) и BazarLoader (также известным как BazarBackdoor).

<https://feodotracker.abuse.ch/downloads/feodotracker.rules> – набор правил используется для обнаружения и/или блокировки сетевых подключений к хост-серверам (комбинация IP-адреса и порта). Набор правил генерируется каждые 5 минут. Рекомендуется обновлять набор правил IDS каждые 5–15 минут, чтобы обеспечить лучшую защиту от Dridex, Emotet, TrickBot, QakBot и BazarLoader.

Поскольку IP-адреса перерабатываются и используются повторно, в этот список блокировки входят только C2-серверы ботнетов, которые либо активны, либо в последний раз использовались в течение последних 30 дней. Таким образом, процент ложных срабатываний в этом списке блокировки должен быть низким.

https://feodotracker.abuse.ch/downloads/feodotracker_aggressive.rules – набор правил IDS с полным списком всех C2-серверов ботнетов. Однако, поскольку IP-адреса используются повторно, количество ложных срабатываний в этом наборе правил намного выше.

Не рекомендуется использовать агрессивную версию индикаторов компрометации ботнета C2 (IOC), так как она определённо вызовет ложные срабатывания.

14.7.3.3. *Travis Green*

Travis Green – набор правил для поиска угроз от специалиста по кибербезопасности Тревиса Грина.

<https://raw.githubusercontent.com/travisbgreen/hunting-rules/master/hunting.rules>

14.7.3.4. *Etnetera Core*

Набор правил с «агрессивными» IP-адресами от центра кибербезопасности компании Etnetera Core.

https://security.etnetera.cz/feeds/etn_aggressive.rules

14.7.4. Пример настройки IPS/IDS с автообновлением правил

Задача:

Организовать защиту локальной сети с автообновлением правил из открытых источников.

192.168.1.0/24 – локальная сеть.

Решение:

Создадим профиль адресов защищаемой локальной сети:

```
rtt(config)# object-group network LAN
rtt(config-object-group-network)# ip prefix 192.168.1.0/24
rtt(config-object-group-network)# exit
```

Настроим на RTT DNS-клиента для разрешения имен источников обновления правил IPS/IDS:

```
rtt(config)# domain lookup enable
rtt(config)# domain nameserver 8.8.8.8
```

Создадим политику безопасности IPS/IDS:

```
rtt(config)# security ips policy OFFICE
rtt(config-ips-policy)# description "My Policy"
rtt(config-ips-policy)# protect network-group LAN
```

Разрешим работу IPS/IDS на интерфейсе локальной сети bridge 1:

```
rtt(config)# bridge 1
rtt(config-bridge)# service-ips inline
```

Настроим параметры IPS/IDS:

```
rtt(config)# security ips
rtt(config-ips)# logging remote-server 192.168.10.1
rtt(config-ips)# logging update-interval 15
rtt(config-ips)# policy OFFICE
rtt(config-ips)# enable
```

Устройство будет использоваться только как шлюз безопасности, поэтому отдадим сервису IPS/IDS все доступные ресурсы:

```
rtt(config-ips)# perfomance max
```

Настроим автообновление правил с сайтов etnetera.cz и Abuse.ch:

```
rtt(config-ips)# auto-upgrade
```

```

rtt(config-auto-upgrade)# user-server Aggressive
rtt(config-ips-upgrade-user-server)# description "Etnetera aggressive IP
blacklist"
rtt(config-ips-upgrade-user-server)# url
https://security.etnetera.cz/feeds/etn_aggressive.rules
rtt(config-ips-upgrade-user-server)# upgrade interval 4
rtt(config-ips-upgrade-user-server)# enable
rtt(config-ips-upgrade-user-server)# exit
rtt(config-auto-upgrade)# user-server SSL-BlackList
rtt(config-ips-upgrade-user-server)# description "Abuse.ch SSL Blacklist"
rtt(config-ips-upgrade-user-server)# url
https://sslbl.abuse.ch/blacklist/sslblacklist_tls_cert.rules
rtt(config-ips-upgrade-user-server)# upgrade interval 4
rtt(config-ips-upgrade-user-server)# enable
rtt(config-ips-upgrade-user-server)# exit

```

14.7.5. Алгоритм настройки базовых пользовательских правил

Шаг	Описание	Команда	Ключи
1	Задать имя и перейти в режим конфигурирования набора пользовательских правил.	<code>rtt(config)# security ips-category user-defined <WORD></code>	<WORD> – имя набора пользовательских правил, задаётся строкой до 32 символов.
2	Задать описание набора пользовательских правил (не обязательно).	<code>rtt(config-ips-category)# description <DESCRIPTION></code>	<DESCRIPTION> – описание задаётся строкой до 255 символов.
3	Создать правило и перейти в режим конфигурирования правила.	<code>rtt(config-ips-category)# rule <ORDER></code>	<ORDER> – номер правила, принимает значения [1..512].
4	Задать описание правила (не обязательно).	<code>rtt(config-ips-category-rule)# description <DESCRIPTION></code>	<DESCRIPTION> – описание задаётся строкой до 255 символов.
5	Указать действие данного правила.	<code>rtt(config-ips-category-rule)# action { alert reject pass drop }</code>	<ul style="list-style-type: none"> • alert – прохождение трафика разрешается, и сервис IPS/IDS генерирует сообщение; • reject – прохождение трафика запрещается. Если это TCP-трафик, отправителю и получателю посылается пакет TCP-RESET, для остального типа трафика посылается пакет ICMP-ERROR. Сервис IPS/IDS генерирует сообщение; • pass – прохождение трафика разрешается; • drop – прохождение трафика запрещается, и сервис IPS/IDS генерирует сообщение.

Шаг	Описание	Команда	Ключи
6	Установить имя IP-протокола, для которого должно срабатывать правило.	<code>rtt (config-ips-category-rule) # protocol <PROTOCOL></code>	<p><PROTOCOL> – принимает значения any/ip/icmp/http/tcp/udp.</p> <p>При указании значения «any» правило будет срабатывать для любых протоколов.</p>
7	Установить IP-адреса отправителя, для которых должно срабатывать правило.	<pre>rtt (config-ips-category-rule) # source-address { ip <ADDR> ip-prefix <ADDR/LEN> object-group <OBJ_GR_NAME> policy-object-group { protect external } any }</pre>	<p><ADDR> – IP-адрес отправителя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><ADDR/LEN> – IP-подсеть отправителя, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и LEN принимает значения [1..32].</p> <p><OBJ_GR_NAME> – имя профиля IP-адресов, который содержит IP-адреса отправителя, задаётся строкой до 31 символа.</p> <ul style="list-style-type: none"> protect – устанавливает в качестве адресов отправителя и protect-адреса определенные адреса в политике IPS/IDS; external – устанавливает в качестве адресов отправителя и external-адреса определенные адреса в политике IPS/IDS. <p>При указании значения «any» правило будет срабатывать для любого IP-адреса отправителя.</p>
8	<p>Установить номера TCP/UDP-портов отправителя, для которых должно срабатывать правило.</p> <p>Для значения protocol icmp, значение source-port может быть только any.</p>	<pre>rtt (config-ips-category-rule) # source-port { any <PORT> object-group <OBJ-GR-NAME> }</pre>	<p><PORT> – номер TCP/UDP-порта отправителя, принимает значения [1..65535].</p> <p><OBJ_GR_NAME> – имя профиля TCP/UDP-портов отправителя, задаётся строкой до 31 символа.</p> <p>При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта отправителя.</p>

Шаг	Описание	Команда	Ключи
9	Установить IP-адреса получателя, для которых должно срабатывать правило.	<pre> rtt (config-ips- category-rule) # destination-address {ip <ADDR> ip-prefix <ADDR/LEN> object- group <OBJ_GR_NAME> policy-object-group { protect external } any } </pre>	<p><ADDR> – IP-адрес получателя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><ADDR/LEN> – IP-подсеть получателя, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и LEN принимает значения [1..32];</p> <p><OBJ_GR_NAME> – имя профиля IP-адресов, который содержит IP-адреса получателя, задаётся строкой до 31 символа.</p> <ul style="list-style-type: none"> • protect – устанавливает в качестве адресов получателя protect-адреса, определенные в политике IPS/IDS; • external – устанавливает в качестве адресов получателя external-адреса, определенные в политике IPS/IDS. <p>При указании значения «any» правило будет срабатывать для любого IP-адреса получателя.</p>
10	<p>Установить номера TCP/UDP-портов получателя, для которых должно срабатывать правило.</p> <p>Для значения protocol icmp, значение destination-port может быть только any.</p>	<pre> rtt (config-ips- category-rule) # destination-port {any <PORT> object-group <OBJ-GR- NAME> } </pre>	<p><PORT> – номер TCP/UDP-порта получателя, принимает значения [1..65535];</p> <p><OBJ_GR_NAME> – имя профиля TCP/UDP-портов получателя, задаётся строкой до 31 символа.</p> <p>При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта получателя.</p>
11	Установить направление потока трафика, для которого должно срабатывать правило.	<pre> rtt (config-ips- category-rule) # direction { one-way round-trip } </pre>	<ul style="list-style-type: none"> • one-way – трафик передаётся в одну сторону. • round-trip – трафик передаётся в обе стороны.
12	Определить сообщение, которое IPS/IDS будет записывать в лог при срабатывании этого правила.	<pre> rtt (config-ips- category-rule) # meta log-message <MESSAGE> </pre>	<p><MESSAGE> – текстовое сообщение, задаётся строкой до 129 символов.</p>

13	<p>Определить классификацию трафика, которая будет записываться в лог при срабатывании этого правила (необязательно).</p>	<pre> rtt (config-ips- category-rule) # meta classification-type { not-suspicious unknown bad-unknown attempted-recon successful-recon- limited successful- recon-largescale attempted-dos successful-dos attempted-user unsuccessful-user successful-user attempted-admin successful-admin rpc-portmap-decode shellcode-detect string-detect suspicious-filename- detect suspicious- login system-call- detect tcp- connection trojan- activity unusual-client-port- connection network- scan denial-of-service non-standard-protocol protocol-command- decode web- application-activity web-application-attack misc-activity misc-attack icmp-event inappropriate-content policy-violation default-login-attempt } </pre>	<ul style="list-style-type: none"> • not-suspicious – неподозрительный трафик. • unknown – неизвестный трафик. • bad-unknown – потенциально плохой трафик. • attempted-recon – попытка утечки информации. • successful-recon-limited – утечка информации. • successful-recon-largescale – масштабная утечка информации. • attempted-dos – попытка отказа в обслуживании. • successful-dos – отказ в обслуживании. • attempted-user – попытка получения привилегий пользователя. • unsuccessful-user – безуспешная попытка получения привилегий пользователя. • successful-user – успешная попытка получения привилегий пользователя. • attempted-admin – попытка получения привилегий администратора. • successful-admin – успешная попытка получения привилегий администратора. • rpc-portmap-decode – декодирование запроса RPC. • shellcode-detect – обнаружен исполняемый код. • string-detect – обнаружена подозрительная строка. • suspicious-filename-detect – было обнаружено подозрительное имя-файла. • suspicious-login – была обнаружена попытка входа с использованием подозрительного имени пользователя. • system-call-detect – обнаружен системный вызов. • tcp-connection – обнаружено TCP-соединение. • trojan-activity – был обнаружен сетевой троян. • unusual-client-port-connection – клиент использовал необычный порт.
----	---	---	--

Шаг	Описание	Команда	Ключи
			<ul style="list-style-type: none"> • network-scan – обнаружение сетевого сканирования. • denial-of-service – обнаружение атаки отказа в обслуживании. • non-standard-protocol – обнаружение нестандартного протокола или события. • protocol-command-decode – обнаружена попытка шифрования. • web-application-activity – доступ к потенциально уязвимому веб-приложению. • web-application-attack – атака на веб-приложение. • misc-activity – прочая активность. • misc-attack – прочие атаки. • icmp-event – общее событие ICMP. • inappropriate-content – обнаружено неприемлемое содержание. • policy-violation – потенциальное нарушение корпоративной конфиденциальности. • default-login-attempt – попытка входа с помощью стандартного логина/пароля.
14	Установить значение кода DSCP, для которого должно срабатывать правило (необязательно).	<code>rtt(config-ips-category-rule)# ip dscp <DSCP></code>	<DSCP> – значение кода DSCP, принимает значения [0..63].
15	Установить значение времени жизни пакета (TTL), для которого должно срабатывать правило (необязательно).	<code>rtt(config-ips-category-rule)# ip ttl <TTL></code>	<TTL> – значение TTL, принимает значения в диапазоне [1..255].
16	Установить номер IP-протокола, для которого должно срабатывать правило (необязательно). Применимо только для значения protocol any.	<code>rtt(config-ips-category-rule)# ip protocol-id <ID></code>	<ID> – идентификационный номер IP-протокола, принимает значения [1..255].
17	Установить значения ICMP CODE, для которого должно	<code>rtt(config-ips-category-rule)# ip icmp code <CODE></code>	<CODE> – значение CODE протокола ICMP, принимает значение в диапазоне [0..255].

Шаг	Описание	Команда	Ключи
	срабатывать правило (необязательно). Применимо только для значения protocol icmp.	<code>rtt(config-ips-category-rule)# ip icmp code comparison-operator { greater-than less-than }</code>	Оператор сравнения для значения ip icmp code: <ul style="list-style-type: none">greater-than – больше чем..less-than – меньше чем..
18	Установить значения ICMP ID, для которого должно срабатывать правило (необязательно). Применимо только для значения protocol icmp.	<code>rtt(config-ips-category-rule)# ip icmp id <ID></code>	<ID> – значение ID протокола ICMP, принимает значение в диапазоне [0.. 65535].
19	Установить значения ICMP Sequence-ID, для которого должно срабатывать правило (необязательно). Применимо только для значения protocol icmp.	<code>rtt(config-ips-category-rule)# ip icmp sequence-id <SEQ-ID></code>	<SEQ-ID> – значение Sequence-ID протокола ICMP, принимает значение в диапазоне [0.. 4294967295].
20	Установить значения ICMP TYPE, для которого должно срабатывать правило (необязательно). Применимо только для значения protocol icmp.	<code>rtt(config-ips-category-rule)# ip icmp type <TYPE></code>	<TYPE> – значение TYPE протокола ICMP, принимает значение в диапазоне [0..255].
		<code>rtt(config-ips-category-rule)# ip icmp type comparison-operator { greater-than less-than }</code>	Оператор сравнения для значения ip icmp type: <ul style="list-style-type: none">greater-than – больше чем..less-than – меньше чем..
21	Установить значения TCP Acknowledgment-Number, для которого должно срабатывать правило (необязательно). Применимо только для значения protocol tcp.	<code>rtt(config-ips-category-rule)# ip tcp acknowledgment-number <ACK-NUM></code>	<ACK-NUM> – значение Acknowledgment-Number протокола TCP, принимает значение в диапазоне [0.. 4294967295].
22	Установить значения TCP Sequence-ID, для которого должно срабатывать правило (необязательно). Применимо только для значения protocol tcp.	<code>rtt(config-ips-category-rule)# ip tcp sequence-id <SEQ-ID></code>	<SEQ-ID> – значение Sequence-ID протокола TCP, принимает значение в диапазоне [0.. 4294967295].

Шаг	Описание	Команда	Ключи
23	<p>Установить значения TCP Window-Size, для которого должно срабатывать правило (необязательно).</p> <p>Применимо только для значения protocol tcp.</p>	<code>rtt(config-ips-category-rule)# ip tcp window-size <SIZE></code>	<SIZE> – значение Window-Size протокола TCP, принимает значение в диапазоне [0.. 65535].
24	<p>Установить ключевые слова протокола HTTP, для которых должно срабатывать правило (необязательно).</p> <p>Применимо только для значения protocol http.</p>	<code>rtt(config-ips-category-rule)# ip http { accept accept-enc accept-lang client-body connection content-type cookie file-data header header-names host method protocol referer request-line response-line server-body start start-code start-msg uri user-agent }</code>	<p>Значение ключевых слов см. в документации Suricata 4.X.</p> <p>https://suricata.readthedocs.io/en/suricata-4.1.4/rules/http-keywords.html</p>
25	<p>Установить значение ключевого слова URI LEN протокола HTTP, для которых должно срабатывать правило (необязательно).</p> <p>Применимо только для значения protocol http.</p>	<code>rtt(config-ips-category-rule)# ip http urilen <LEN></code>	<LEN> – принимает значение в диапазоне [0.. 65535].
		<code>rtt(config-ips-category-rule)# ip http urilen comparison-operator { greater-than less-than }</code>	<p>Оператор сравнения для значения ip http urilen:</p> <ul style="list-style-type: none"> • greater-than – больше чем.. • less-than – меньше чем..
26	<p>Установить значение содержимого пакетов (Payload content), для которых должно срабатывать правило (необязательно).</p>	<code>rtt(config-ips-category-rule)# payload content <CONTENT></code>	<CONTENT> – текстовое сообщение, задаётся строкой до 1024 символов.
27	<p>Не различать прописные и заглавные буквы в описании содержимого пакетов (необязательно).</p> <p>Применимо только совместно с командой payload content.</p>	<code>rtt(config-ips-category-rule)# payload no-case</code>	

Шаг	Описание	Команда	Ключи
28	Установить, сколько байтов с начала содержимого пакета будет проверено (необязательно). Применимо только совместно с командой <code>payload content</code> .	<code>rtt (config-ips-category-rule) # payload depth <DEPTH></code>	<DEPTH> – число байт с начала содержимого пакета, принимает значение в диапазоне [1.. 65535]. По умолчанию проверяется все содержимое пакета.
29	Установить число байт смещения от начала содержимого пакета для проверки (необязательно). Применимо только совместно с командой <code>payload content</code> .	<code>rtt (config-ips-category-rule) # payload offset <OFFSET></code>	<OFFSET> – число байт смещения от начала содержимого пакета, принимает значение в диапазоне [1.. 65535]. По умолчанию проверяется с начала содержимого.
30	Установить размер содержимого пакетов, для которых должно срабатывать правило (необязательно).	<code>rtt (config-ips-category-rule) # payload data-size <SIZE></code>	<SIZE> – размер содержимого пакетов, принимает значение в диапазоне [0.. 65535].
		<code>rtt (config-ips-category-rule) # payload data-size comparison-operator { greater-than less-than }</code>	Оператор сравнения для значения <code>payload data-size</code> : <ul style="list-style-type: none">• <code>greater-than</code> – больше чем..• <code>less-than</code> – меньше чем.
31	Указать пороговое значение количества пакетов, при котором сработает правило (необязательно).	<code>rtt (config-ips-category-rule) # threshold count <COUNT></code>	<COUNT> – число пакетов, принимает значение в диапазоне [1.. 65535].
32	Указать интервал времени, для которого считается пороговое количество пакетов. (Обязательно, если включен <code>threshold count</code>).	<code>rtt (config-ips-category-rule) # threshold second <SECOND></code>	<SECOND> – интервал времени в секундах, принимает значение в диапазоне [1.. 65535].
33	Указать по адресу отправителя или получателя будут считаться пороги. (Обязательно, если включен <code>threshold count</code>).	<code>rtt (config-ips-category-rule) # threshold track { by-src by-dst }</code>	<ul style="list-style-type: none">• <code>by-src</code> – считать пороговое значение для пакетов с одинаковым IP-отправителя.• <code>by-dst</code> – считать пороговое значение для пакетов с одинаковым IP-получателя.

Шаг	Описание	Команда	Ключи
34	Указать метод обработки пороговых значений.	<code>rtt(config-ips-category-rule)# threshold type {threshold limit both }</code>	<ul style="list-style-type: none"> • <code>threshold</code> – выдавать сообщение каждый раз по достижении порога. • <code>limit</code> – выдавать сообщение не чаще <code><COUNT></code> раз за интервал времени <code><SECOND></code>. • <code>both</code> – комбинация <code>threshold</code> и <code>limit</code>. <p>Сообщение будет генерироваться, если в течении интервала времени <code><SECOND></code> было <code><COUNT></code> или более пакетов подходящих под условия правила, и сообщение будет отправлено только один раз в течении интервала времени <code><SECOND></code>.</p>
35	Активировать правило.	<code>rtt(config-ips-category-rule)# enable</code>	

14.7.6. Пример настройки базовых пользовательских правил

Задача:

Написать правило для защиты сервера с IP 192.168.1.10 от DOS-атаки ICMP-пакетами большого размера.

Решение:

Создадим набор пользовательских правил:

```
rtt(config)# security ips-category user-defined USER
```

Создадим правило для защиты от атаки:

```
rtt(config-ips-category)# rule 10
rtt(config-ips-category-rule)# description "Big ICMP DoS"
```

Будем отбрасывать пакеты:

```
rtt(config-ips-category-rule)# action drop
```

Настроим сообщение об атаке:

```
rtt(config-ips-category-rule)# meta log-message "Big ICMP DoS"
rtt(config-ips-category-rule)# meta classification-type successful-dos
```

Укажем тип протокола для правила:

```
rtt(config-ips-category-rule)# protocol icmp
```


Так как был указан протокол `icmp`, то в качестве порта отправителя и получателя требуется указать `any`:

```
rtt(config-ips-category-rule)# source-port any
rtt(config-ips-category-rule)# destination-port any
```

В качестве адреса получателя укажем используемый сервер:

```
rtt(config-ips-category-rule)# destination-address ip 192.168.1.10
```

Атакующий может отправлять пакеты с любого адреса:

```
rtt(config-ips-category-rule)# source-address any
```

Зададим направление трафика:

```
rtt(config-ips-category-rule)# direction one-way
```

Правило будет срабатывать на пакеты размером больше 1024 байт:

```
rtt(config-ips-category-rule)# payload data-size 1024
rtt(config-ips-category-rule)# payload data-size comparison-operator greater-than
```

Правило будет срабатывать, если нагрузка на сервер будет превышать 3 Мбит/с, при этом сообщение об атаке будет генерироваться не чаще одного раза в минуту:

```
3 Мб/с = 3145728 бит в сек
Пакет размером 1Кбайт = 8192 бита
3145728 / 8192 = 384 пакета в сек
384 * 60 = 23040 пакетов в минуту
rtt(config-ips-category-rule)# threshold count 23040
rtt(config-ips-category-rule)# threshold second 60
rtt(config-ips-category-rule)# threshold track by-dst
rtt(config-ips-category-rule)# threshold type both
```

Активируем правило:

```
rtt(config-ips-category-rule)# enable
```

14.7.7. Алгоритм настройки расширенных пользовательских правил

Шаг	Описание	Команда	Ключи
1	Задать имя и перейти в режим конфигурирования набора пользовательских правил.	<code>rtt(config)# security ips-category user-defined <WORD></code>	<WORD> – имя набора пользовательских правил, задаётся строкой до 32 символов.

Шаг	Описание	Команда	Ключи
2	Задать описание набора пользовательских правил (необязательно).	<code>rtt(config-ips-category) # description <DESCRIPTION></code>	<DESCRIPTION> – описание задаётся строкой до 255 символов.
3	Создать расширенное правило и перейти в режим его конфигурирования.	<code>rtt(config-ips-category) # rule-advanced <SID></code>	<SID> – номер правила, принимает значения [1.. 4294967295].
4	Задать описание правила (необязательно).	<code>rtt(config-ips-category-rule-advanced) # description <DESCRIPTION></code>	<DESCRIPTION> – описание задаётся строкой до 255 символов.
5	Указать действие данного правила.	<code>rtt(config-ips-category-rule-advanced) # rule-text <LINE></code>	<p><CONTENT> – текстовое сообщение в формате SNORT 2.X / Suricata 4.X, задаётся строкой до 1024 символов.</p> <p>При написании правил в тексте правила необходимо использовать только двойные кавычки (символ <code>"</code>), а само правило необходимо заключать в одинарные кавычки (символ <code>'</code>).</p>
6	Активировать правило.	<code>rtt(config-ips-category-rule-advanced) # enable</code>	

14.7.8. Пример настройки расширенных пользовательских правил

Задача:

Написать правило, детектирующее атаку типа Slowloris.

Решение:

Создадим набор пользовательских правил:

```
rtt(config)# security ips-category user-defined ADV
```

Создадим расширенное правило:

```
rtt(config-ips-category) # rule-advanced 1
rtt(config-ips-category-rule-advanced) # description "Slow Loris rule 1"
rtt(config-ips-category-rule-advanced) # rule-text 'alert tcp any any -> any 80
(msg:"Possible Slowloris Attack Detected"; flow:to_server,established;
content:"X-a|3a|"; distance:0; pcre:"/\d\d\d\d/"; distance:0; content:"|0d 0a|";
sid:10000001;)'
rtt(config-ips-category-rule-advanced) # enable
rtt(config-ips-category-rule-advanced) # exit
```

Создадим ещё одно расширенное правило, работающее по схожему алгоритму, чтобы определить, какое из правил будет эффективнее:

```
rtt(config-ips-category)# rule-advanced 2
rtt(config-ips-category-rule-advanced)# description "Slow Loris rule 2"
rtt(config-ips-category-rule-advanced)# rule-text 'alert tcp $EXTERNAL_NET any -
> $HOME_NET $HTTP_PORTS (msg:"SlowLoris.py DoS attempt";
flow:established,to_server,no_stream; content:"X-a: "; dsize:<15;
detection_filter:track by_dst, count 3, seconds 30; classtype:denial-of-service;
sid: 10000002; rev:1; )'
rtt(config-ips-category-rule-advanced)#
```

15.УПРАВЛЕНИЕ СЕРТИФИКАТАМИ И КЛЮЧАМИ

15.1. Автоматическое распространение ключей и сертификатов X.509

15.1.1. *Общее описание инфраструктуры открытых ключей*

Инфраструктура открытых ключей (Public Key Infrastructure, PKI) – комплекс средств, мер и политик, обеспечивающих работу систем на базе алгоритмов шифрования с открытым ключом.

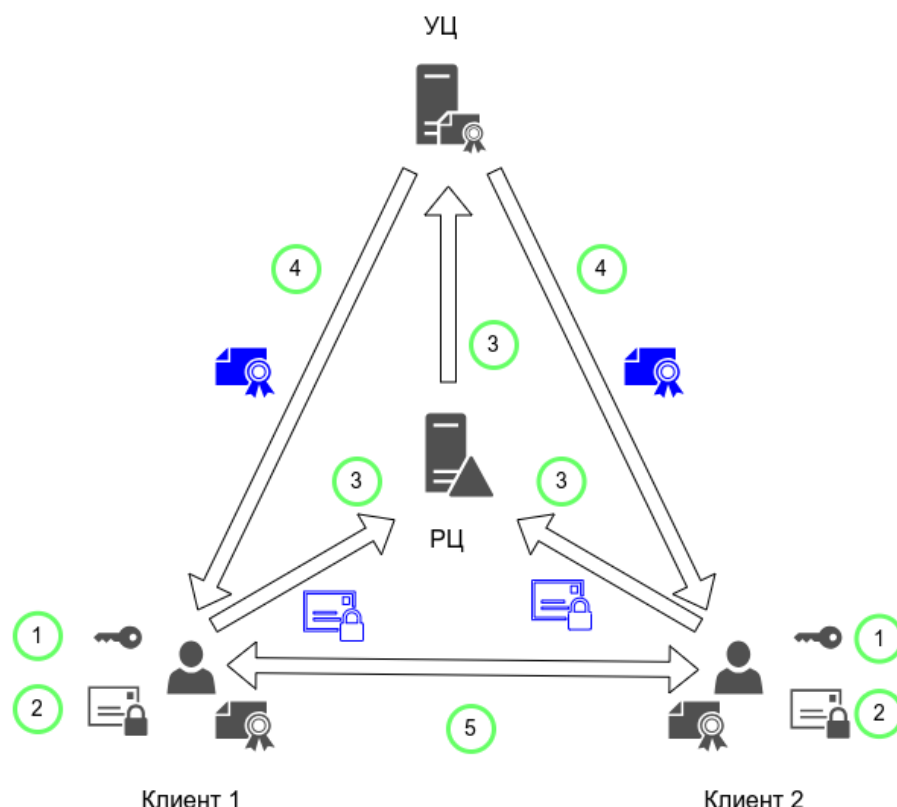
К участникам PKI обычно относятся:

- Удостоверяющий центр (УЦ),
- Регистрационный центр (РЦ),
- Клиент.

В процессе взаимодействия участников PKI появляются и используются следующие объекты:

- Приватный ключ,
- Публичный ключ,
- Запрос на сертификацию,
- Сертификат,
- Список отозванных сертификатов.

Схематично основные процедуры в инфраструктуре открытых ключей можно описать следующим образом:



1. Клиент генерирует у себя приватный ключ.
2. Клиент формирует запрос на сертификацию, включающий в себя публичный ключ, парный к ранее сгенерированному приватному ключу, информацию о владельце приватного ключа и запрашиваемые опции, которые затем могут быть добавлены в выпускаемый сертификат.
3. Клиент доставляет запрос на сертификацию в удостоверяющий центр. При наличии в структуре PKI регистрационного центра запрос от клиента будет приходить на него и после валидации будет уходить в удостоверяющий центр.
4. Удостоверяющий центр на основании клиентского запроса на сертификацию выпишет сертификат, содержащий всю клиентскую информацию и информацию о самом УЦ. Сам сертификат будет подписан приватным ключом УЦ, тем самым данные в клиентском сертификате не смогут быть изменены третьей стороной. От УЦ сертификат передается клиенту.
5. Теперь клиенты могут, используя свой приватный ключ, подписывать данные в рамках используемых процессов (в нашем случае – сетевых протоколов) при отправке и валидировать данные при помощи публичного сертификата на приёме.

15.1.2. Планирование инфраструктуры открытых ключей

Сервисные маршрутизаторы RTT могут принимать участие в структуре PKI в качестве корневого удостоверяющего центра и клиента PKI. Перед развертыванием инфраструктуры PKI крайне важно её спланировать для обеспечения безопасности и стабильной работы.

Основные рекомендации по планированию инфраструктуры открытых ключей:

1. Определите роли устройств в структуре PKI.
2. Определите схему именования сертификатов.
3. Определите политику доступа к центру сертификации и механизмы ограничения доступа к нему от недоверенных клиентов.
4. Обеспечьте актуальность времени на всех хостах, используемых в структуре PKI.

15.1.3. Настройка PKI-сервера в роли корневого удостоверяющего центра

Базовый вариант настройки PKI-сервера – корневой удостоверяющий центр, работающий на самоподписанных сертификатах, который напрямую обслуживает запросы конечных клиентов.



Изменение отличительного имени сертификата удостоверяющего центра приводит к регенерации приватного ключа и сертификата PKI-сервера и очистке базы выписанных сертификатов. Требуется заранее, на этапе планирования структуры PKI, определиться с идентификацией владельца удостоверяющего центра и после ввода удостоверяющего центра в эксплуатацию не менять эти настройки.

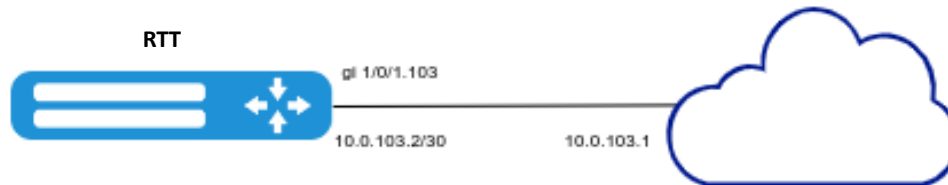
15.1.3.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Настроить NTP-клиент согласно разделу Настройка NTP . Маршрутизатор должен получать точное время от NTP-сервера или NTP-пира, либо доверять локально настроенному времени с включенным режимом NTP Master.		
2	Перейти в режим настройки PKI-сервера.	<code>rtt(config)# crypto pki server</code>	
3	Перейти в режим настройки отличительного имени сертификата – набора атрибутов, уникально описывающих удостоверяющий центр.	<code>rtt(config-pki- server)# subject-name</code>	
4	Указать код страны (необязательно).	<code>rtt(config-pki-server- subject-name)# country <COUNTRY></code>	<COUNTRY> – код страны, задаётся строкой длиной 2 символа. Рекомендуется использовать двухбуквенные обозначения стран "alpha-2" из стандарта ISO 3166-1.
5	Указать название штата, области или провинции (необязательно).	<code>rtt(config-pki-server- subject-name)# state <STATE></code>	<STATE> – название штата, области или провинции, задаётся строкой от 1 до 128 символов.

Шаг	Описание	Команда	Ключи
6	Указать название населенного пункта или его территориальной единицы (необязательно).	<code>rtt(config-pki-server-subject-name) # locality <LOCATION></code>	<LOCATION> – название населенного пункта или его территориальной единицы, задаётся строкой от 1 до 128 символов.
7	Указать название организации (необязательно).	<code>rtt(config-pki-server-subject-name) # organization <ORGANIZATION></code>	<ORGANIZATION> – название организации, задаётся строкой от 1 до 64 символов.
8	Указать подразделение организации (необязательно).	<code>rtt(config-pki-server-subject-name) # organization-unit <ORGANIZATION-UNIT></code>	<ORGANIZATION-UNIT> – название подразделения организации, задаётся строкой от 1 до 64 символов.
9	Указать общее имя сертификата.	<code>rtt(config-pki-server-subject-name) # common-name <COMMON-NAME></code>	<COMMON-NAME> – общее имя, задаётся строкой от 1 до 64 символов. Чаще всего в качестве общего имени сертификата удостоверяющего центра используется имя домена, который обслуживает удостоверяющий центр или юридическое название удостоверяющего центра.
10	Указать IP-адрес или сетевой интерфейс, который будет прослушиваться PKI-сервером для обработки входящих запросов (необязательно).	<code>rtt(config-pki-server) # source-address <ADDR></code>	<ADDR> – IP-адрес, назначенный на локальном сетевом интерфейсе маршрутизатора, на котором PKI-сервер будет слушать входящие подключения, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
	В случае, если настройка не будет указана PKI-сервер будет принимать запросы на всех настроенных IP-интерфейсах маршрутизатора.	<code>rtt(config-pki-server) # source-interface <IF></code>	<IF> – интерфейс или туннель, на котором PKI-сервер будет слушать входящие подключения.
11	Указать challenge-password, который будет использован для аутентификации PKI-клиентов, желающих выписать сертификат (необязательно).	<code>rtt(config-pki-server) # challenge-password { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 32 символов; <ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 32 байт (от 16 до 64 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).
12	Указать время жизни выписываемых клиентам сертификатов в днях.	<code>rtt(config-pki-server) # lifetime <DAYS></code>	<DAYS> – количество дней, принимает значения в диапазоне [1..3650]. Значение по умолчанию: 365.
13	Включить PKI-сервер.	<code>rtt(config-pki-server) # enable</code>	

Задача:

Настроить на маршрутизаторе PKI-сервер в роли корневого удостоверяющего центра. Клиентские сертификаты должны иметь срок жизни – две недели с момента выпуска.



В качестве начальной конфигурации на маршрутизаторе уже настроен сетевой интерфейс в сторону Интернет-провайдера, прописан шлюз по умолчанию и настроена зона безопасности. Шлюз Интернет-провайдера также может служить источником синхронизации времени по протоколу NTP.

```

hostname RTT.CA

security zone WAN
exit

interface gigabitethernet 1/0/1.103
    security-zone WAN
    ip address 10.0.103.2/30
exit

ip route 0.0.0.0/0 10.0.103.1
    
```

Решение:

Настроим NTP-клиент на получение точного времени от шлюза Интернет-провайдера:

```

RTT.CA(config)# ntp enable
RTT.CA(config)# ntp server 10.0.103.1
RTT.CA(config-ntp-server)# exit
RTT.CA(config)#
    
```

Перейдем к настройке PKI-сервера:

```

RTT.CA(config)# crypto pki server
RTT.CA(config-pki-server)#
    
```

Перейдем в раздел настройки отличительного имени сертификата удостоверяющего центра. Настроим те атрибуты, которые позволят однозначно идентифицировать удостоверяющий центр:

```

RTT.CA(config-pki-server)# subject-name
RTT.CA(config-pki-server-subject-name)# country RU
RTT.CA(config-pki-server-subject-name)# state Moscow
RTT.CA(config-pki-server-subject-name)# locality Moscow
    
```



```
RTT.CA(config-pki-server-subject-name)# organization Company
RTT.CA(config-pki-server-subject-name)# common-name ca.company.loc
RTT.CA(config-pki-server-subject-name)# exit
RTT.CA(config-pki-server)#
```

Привяжем PKI-сервер к адресу сетевого интерфейса, смотрящего в сторону Интернет-провайдера:

```
RTT.CA(config-pki-server)# source-interface gi 1/0/1.103
RTT.CA(config-pki-server)#
```

Зададим challenge-password, для корректного обращения к удостоверяющему центру PKI-клиенты должны использовать правильный challenge-password.

```
RTT.CA(config-pki-server)# challenge-password StR0nnGP+ss
RTT.CA(config-pki-server)#
```

Зададим время жизни выдаваемых клиентам сертификатов:

```
RTT.CA(config-pki-server)# lifetime 14
RTT.CA(config-pki-server)#
```

Включим PKI-сервер:

```
RTT.CA(config-pki-server)# enable
RTT.CA(config-pki-server)# exit
RTT.CA(config)#
```

Добавим пару зон безопасности и правило, разрешающее прохождение входящего на PKI-сервер трафика:

```
RTT.CA(config)# security zone-pair WAN self
RTT.CA(config-security-zone-pair)# rule 10
RTT.CA(config-security-zone-pair-rule)# description "Allow access to PKI-server
from WAN"
RTT.CA(config-security-zone-pair-rule)# match protocol tcp
RTT.CA(config-security-zone-pair-rule)# match destination-port port-range 80
RTT.CA(config-security-zone-pair-rule)# match destination-address address-range
10.0.103.2
RTT.CA(config-security-zone-pair-rule)# action permit
RTT.CA(config-security-zone-pair-rule)# enable
RTT.CA(config-security-zone-pair-rule)# exit
RTT.CA(config-security-zone-pair)# exit
RTT.CA(config)#
```

Применим конфигурацию на маршрутизаторе:

```
RTT.CA(config)# end
Warning: you have uncommitted configuration changes.
RTT.CA# commit
Configuration has been successfully applied and saved to flash. Commit timer
started, changes will be reverted in 600 seconds.
RTT.CA# confirm
```

Configuration has been confirmed. Commit timer canceled.

RTT.CA#

В результате получим запущенный корневой удостоверяющий центр, готовый к обслуживанию клиентских запросов. В команде **show crypto pki server** можно увидеть **fingerprint** сертификата удостоверяющего центра, который необходимо использовать в клиентах PKI совместно с установленным **challenge-password** для корректной авторизации:

```
RTT.CA# show crypto pki server
Status: Enabled
Lifetime days: 14
Certificate fingerprint: 79:D2:B6:7E:DF:77:2D:C5:27:68:99:10:BA:EC:D2:47
Source: gigabitethernet 1/0/1.103
Last issued serial number: --
Challenge password: Active
RTT.CA#
```

15.1.4. Настройка PKI-клиента



Изменение отличительного имени клиентского сертификата или URL подключения к PKI-серверу приводит к немедленному перезапросу нового сертификата у удостоверяющего центра.

15.1.4.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Перейти в режим настройки PKI-клиента.	<code>rtt(config)# crypto pki trustpoint <TRUSTPOINT></code>	<TRUSTPOINT> – имя PKI-клиента, задаётся строкой от 1 до 31 символа.
2	Перейти в режим настройки отличительного имени сертификата – набора атрибутов, уникально описывающих владельца сертификата.	<code>rtt(config-trustpoint)# subject-name</code>	
3	Указать код страны (необязательно).	<code>rtt(config-trustpoint-subject-name)# country <COUNTRY></code>	<COUNTRY> – код страны, задаётся строкой длиной 2 символа. Рекомендуется использовать двухбуквенные обозначения стран "alpha-2" из стандарта ISO 3166-1.
4	Указать название штата, области или провинции (необязательно).	<code>rtt(config-trustpoint-subject-name)# state <STATE></code>	<STATE> – название штата, области или провинции, задаётся строкой от 1 до 128 символов.
5	Указать название населенного пункта или его территориальной единицы (необязательно).	<code>rtt(config-trustpoint-subject-name)# locality <LOCATION></code>	<LOCATION> – название населенного пункта или его территориальной единицы, задаётся строкой от 1 до 128 символов.

Шаг	Описание	Команда	Ключи
6	Указать название организации (необязательно).	<code>rtt(config-server-subject-name) # organization <ORGANIZATION></code>	<ORGANIZATION> – название организации, задаётся строкой от 1 до 64 символов.
7	Указать подразделение организации (необязательно).	<code>rtt(config-trustpoint-subject-name) # organization-unit <ORGANIZATION-UNIT></code>	<ORGANIZATION-UNIT> – название подразделения организации, задаётся строкой от 1 до 64 символов.
8	Указать общее имя сертификата.	<code>rtt(config-trustpoint-subject-name) # common-name <COMMON-NAME></code>	<COMMON-NAME> – общее имя, задаётся строкой от 1 до 64 символов. Чаще всего в качестве общего имени клиентского сертификата используется полное доменное имя хоста, который использует данный клиентский сертификат или ФИО пользователя, использующего клиентский сертификат.
9	Перейти в режим настройки альтернативных имен сертификата (необязательно).	<code>rtt(config-trustpoint) # subject-alt-name</code>	
10	Указать IP-адрес в качестве альтернативного имени сертификата (необязательно).	<code>rtt(config-trustpoint-san) # ipv4 <ADDR></code>	<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		<code>rtt(config-trustpoint-san) # ipv6 <IPV6-ADDR></code>	<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].
11	Указать полное доменное имя хоста в качестве альтернативного имени сертификата (необязательно).	<code>rtt(config-trustpoint-san) # dns <NAME></code>	<NAME> – полное доменное имя хоста (FQDN), задается строкой от 1 до 235 символов. Пример записи доменного имени – router.example.loc.
12	Указать адрес электронной почты в качестве альтернативного имени сертификата (необязательно).	<code>rtt(config-trustpoint-san) # email <EMAIL></code>	<EMAIL> – адрес электронной почты, задается строкой от 6 до 254 символов. Пример записи электронной почты – router@example.loc.

Шаг	Описание	Команда	Ключи
13	Указать URL для подключения к PKI-серверу.	<code>rtt(config-trustpoint) # url <URL></code>	<p><URL> – URL для подключения к PKI-серверу, задается в виде</p> <p>"http:// <ADDR>[:<PORT>]/", где:</p> <ul style="list-style-type: none"> • <ADDR> – IP-адрес или доменное имя PKI-сервера; • <PORT> – порт, на котором запущен PKI-сервер, в случае если используется 80 порт по умолчанию, то настройку можно пропустить. <p>В случае если PKI-сервер расположен на том же маршрутизаторе, на котором настраивается PKI-клиент необходимо указать в URL любой из IP-интерфейсов, которые слушает PKI-сервер.</p>
14	Указать цифровой отпечаток сертификата PKI-сервера, к которому выполняется подключение.	<code>rtt(config-trustpoint) # fingerprint <FINGERPRINT></code>	<p><FINGERPRINT> – значение цифровой отпечаток сертификата, полученного при помощи алгоритма MD5. Имеет размер 16 байт и задается в виде HEX-строки длиной 32 символа без разделителей (YYYY...) или 47 символов с двоеточием в роли разделителя (YY:YY:YY...).</p>
15	Указать challenge-password, который будет использован для аутентификации на PKI-сервере (необязательно).	<code>rtt(config-trustpoint) # challenge-password { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<p><CLEAR-TEXT> – пароль, задаётся строкой от 8 до 32 символов;</p> <p><ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 32 байт (от 16 до 64 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).</p>
16	Включить PKI-клиент.	<code>rtt(config-trustpoint) # enable</code>	

15.1.4.2. Пример настройки PKI-клиента для подключения к корневому удостоверяющему центру

Задача:

Настроить на маршрутизаторе PKI-клиент так, чтобы он успешно подключался к удостоверяющему центру. В качестве удостоверяющего центра рассмотрим полностью настроенный PKI-сервер из предыдущего пункта документации. Помимо настройки отличительного имени сертификата настроим в качестве альтернативного имени IP-адрес маршрутизатора и его полное доменное имя.



В качестве начальной конфигурации на маршрутизаторе уже настроен сетевой интерфейс в сторону Интернет-провайдера, прописан шлюз по умолчанию и настроена зона безопасности. Шлюз Интернет-провайдера также может служить источником синхронизации времени по протоколу NTP.

```

hostname RTT.CA

security zone WAN
exit

interface gigabitethernet 1/0/1.103
    security-zone WAN
    ip address 10.0.103.2/30
exit

security zone-pair WAN self
    rule 10
        description "Allow access to PKI-server from WAN"
        action permit
        match protocol tcp
        match destination-address address-range 10.0.103.2
        match destination-port port-range 80
        enable
    exit
exit

ip route 0.0.0.0/0 10.0.103.1

ntp enable
ntp server 10.0.103.1
exit

crypto pki server
    challenge-password StR0nnGPass
    subject-name
        common-name ca.company.loc
        organization Company
        locality Moscow
        state Moscow
        country RU
    exit
    lifetime 14
    source-interface gigabitethernet 1/0/1.103
    enable
exit
hostname RTT.R1

security zone WAN
exit

interface gigabitethernet 1/0/1.113

```

```
security-zone WAN
ip address 10.0.113.2/30
exit

ip route 0.0.0.0/0 10.0.113.1
```

Решение:

Настроим NTP-клиент на получение точного времени от шлюза Интернет-провайдера:



Настройка NTP-клиента для работы PKI-клиента не является обязательной, в отличие от конфигурирования PKI-сервера. Но в связи с чувствительностью инфраструктуры открытых ключей к актуальности времени на узлах, использующих выписанные сертификаты настройка NTP рекомендуется и на клиентской стороне.

```
RTT.R1(config)# ntp enable
RTT.R1(config)# ntp server 10.0.113.1
RTT.R1(config-ntp-server)# exit
RTT.R1(config)#
```

Перейдем к настройке PKI-клиента:

```
RTT.R1(config)# crypto pki trustpoint TP_R1
RTT.R1(config-trustpoint)#
```

Перейдем в раздел настройки отличительного имени клиентского сертификата. Настроим те атрибуты, которые позволяют однозначно идентифицировать клиента:

```
RTT.R1(config-trustpoint)# subject-name
RTT.R1(config-trustpoint-subject-name)# country RU
RTT.R1(config-trustpoint-subject-name)# state Moscow
RTT.R1(config-trustpoint-subject-name)# locality Moscow
RTT.R1(config-trustpoint-subject-name)# organization Company
RTT.R1(config-trustpoint-subject-name)# common-name r1.company.loc
RTT.R1(config-trustpoint-subject-name)# exit
RTT.R1(config-trustpoint)#
```

Перейдем в раздел настройки альтернативных имен сертификата. Укажем в качестве альтернативных имен IP-адрес и полное доменное имя:

```
RTT.R1(config-trustpoint)# subject-alt-name
RTT.R1(config-trustpoint-san)# ipv4 10.0.113.2
RTT.R1(config-trustpoint-san)# dns r1.company.loc
RTT.R1(config-trustpoint-san)# exit
RTT.R1(config-trustpoint)#
```

Укажем URL подключения к удостоверяющему центру:

```
RTT.R1(config-trustpoint)# url http://10.0.103.2/
RTT.R1(config-trustpoint)#
```

Укажем цифровой отпечаток сертификата удостоверяющего центра, при несовпадении настроенного в конфигурации PKI-клиента отпечатка с отпечатком сертификата, которым представится

сам удостоверяющий центр процесс выпуска сертификата будет прерван. На сервисных маршрутизаторах RTT получить цифровой отпечаток сертификата удостоверяющего центра можно из вывода команды **show crypto pki server**:

```
RTT.R1(config-trustpoint)# fingerprint
79:D2:B6:7E:DF:77:2D:C5:27:68:99:10:BA:EC:D2:47
RTT.R1(config-trustpoint)#
```

Зададим challenge-password, для корректного обращения к удостоверяющему центру:

```
RTT.R1(config-trustpoint)# challenge-password StR0nnGP+ss
RTT.R1(config-trustpoint)#
```

Включим PKI-клиент:

```
RTT.R1(config-trustpoint)# enable
RTT.R1(config-trustpoint)# exit
RTT.R1(config)#
```

Поскольку трафик PKI-клиента исходящий, дополнительных правил фильтрации в Zone-Based Firewall добавлять не требуется.

Применим конфигурацию на маршрутизаторе:

```
RTT.R1(config)# end
Warning: you have uncommitted configuration changes.
RTT.R1# commit
Configuration has been successfully applied and saved to flash. Commit timer
started, changes will be reverted in 600 seconds.
RTT.R1# confirm
Configuration has been confirmed. Commit timer canceled.
RTT.R1#
```

В результате получим настроенный PKI-клиент, который сразу после запуска обратится к удостоверяющему центру за выпуском сертификата. Отследить состояние настроенных PKI-клиентов можно командой **show crypto pki trustpoints**:

```
RTT.R1# show crypto pki trustpoints
```

Name	Enrollment	Subject name	Status	Next action
TP_R1	SCEP	/CN=r1.company.loc/O=Company/L =Moscow/ST=Moscow/C=RU	Ready	2025-11-02 11:35:39

```
RTT.R1#
```

Более подробную информацию о PKI-клиенте можно посмотреть, если выполнить команду **show crypto pki trustpoint** с указанием имени PKI-клиента:

```
RTT.R1# show crypto pki trustpoint TP_R1
```

```
Name: TP_R1
Enrollment: SCEP
Subject name: /CN=r1.company.loc/O=Company/L=Moscow/ST=Moscow/C=RU
Challenge password: Active
Status: Ready
Renew date: 2025-11-02 11:35:39
```

RTT.R1#

15.1.5. Процесс автоматического перевыпуска сертификата PKI-клиента

При успешном получении сертификата от удостоверяющего центра PKI-клиент вычитывает период действия сертификата и за 20% времени до его истечения планирует процедуру перевыпуска нового сертификата. Дата запланированного перевыпуска сертификата соответствующего PKI-клиента присутствует в графе "Renew date" вывода команды **show crypto pki trustpoint** с указанием имени конкретного PKI-клиента:

```
RTT.R1# show crypto pki trustpoint TP_R1
Name: TP_R1
Enrollment: SCEP
Subject name: /CN=r1.company.loc/O=Company/L=Moscow/ST=Moscow/C=RU
Challenge password: Active
Status: Ready
Renew date: 2025-11-02 11:35:39
RTT.R1#
```

При наступлении этого времени PKI-клиент запустит процедуру перевыпуска сертификата и в случае успеха заменит текущий сертификат на новый. На это укажет как факт смены запланированной даты перевыпуска сертификата, так и новый серийный номер сертификата:

```
RTT.R1# show crypto pki trustpoint TP_R1 cert | include "Serial"
Serial:
04:7E:C0:EF:F7:D0:46:53:AF:9D:C8:EE:17:A6:14:CC
RTT.R1# show crypto pki trustpoint TP_R1
Name: TP_R1
Enrollment: SCEP
Subject name: /CN=r1.company.loc/O=Company/L=Moscow/ST=Moscow/C=RU
Challenge password: Active
Status: Ready
Renew date: 2025-11-02 11:35:39
RTT.R1#
RTT.R1# show date
"2025-11-02 11:35:46"
RTT.R1# show crypto pki trustpoint TP_R1
Name: TP_R1
Enrollment: SCEP
Subject name: /CN=r1.company.loc/O=Company/L=Moscow/ST=Moscow/C=RU
Challenge password: Active
Status: Ready
Renew date: 2025-11-13 18:21:11
RTT.R1# show crypto pki trustpoint TP_R1 cert | include "Serial"
Serial:
10:34:38:55:CE:D1:4A:98:A5:0E:3F:9E:32:77:E7:22
RTT.R1#
```

В базе сертификатов удостоверяющего центра, в свою очередь, выписанный ранее клиентский сертификат будет отозван:


```
RTT.CA# show crypto pki server database
```

Serial	State	Issue date	Expiration date	Revocation date	Subject name
04:7E:C0:EF:F7:D0:46:53:AF:9D:C8:EE:17:A6:14:CC	Revoked	2025-10-22 06:47:39	2025-11-05 06:47:39	2025-11-02 11:36:01	/CN=r1.company.loc/O=Company/L =Moscow/ST=Moscow/C=RU
10:34:38:55:CE:D1:4A:98:A5:0E:3F:9E:32:77:E7:22	Valid	2025-11-02 11:36:01	2025-11-16 08:02:34	--	/CN=r1.company.loc/O=Company/L =Moscow/ST=Moscow/C=RU

```
RTT.CA#
```

15.1.6. Процесс автоматического перевыпуска сертификата PKI-сервера



В текущей версии ПО маршрутизаторов RTT автоматический перевыпуск сертификата PKI-сервера не реализован. Самоподписанный сертификат удостоверяющего центра генерируется на этапе настройки PKI-сервера сроком на 10 лет с момента генерации. Поддержка перевыпуска сертификата удостоверяющего центра будет поддержана в одном из следующих релизов.

15.2. Ручная генерация и распространение ключей и сертификатов X.509

Процесс выпуска сертификатов и ключей для маршрутизаторов RTT может быть и ручным. Небольшой набор команд позволяет администратору формировать приватный ключ, сгенерировать по нему запрос на сертификацию и отправить его на удаленный хост, откуда его уже можно будет доставить в УЦ для выпуска сертификата. Пользовательский сертификат, а также сопутствующие файлы могут быть загружены обратно на RTT для дальнейшего использования в сервисах маршрутизатора.

15.2.1. Алгоритм генерации ключей и запросов на сертификацию

Шаг	Описание	Команда	Ключи
1	Сгенерировать приватный ключ RSA.	<code>rtt# crypto generate private-key rsa <KEY-SIZE> filename <NAME></code>	<p><KEY-SIZE> – размер ключа в битах. Значение может находиться в диапазоне от 1024 до 4096;</p> <p><NAME> – имя файла приватного ключа, задаётся строкой до 31 символа.</p>

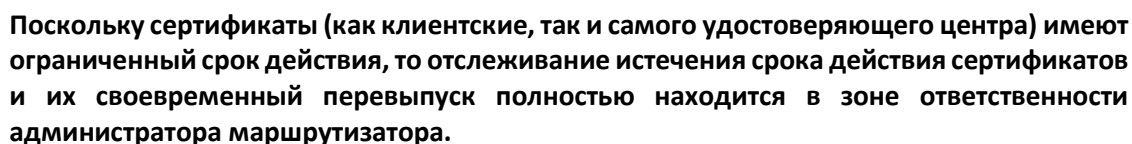
Шаг	Описание	Команда	Ключи
2	Сформировать запрос на сертификацию.	<pre> rtt# crypto generate csr private-key <PRIVATE-KEY> [[alternative-name <ALT-NAME>] [common-name <COMMON-NAME>] [country <COUNTRY>] [email-address <E-MAIL>] [locality <LOCATION>] [organization <ORGANIZATION>] [organizational-unit <ORGANIZATION-UNIT>] [state <STATE>]] filename <NAME> </pre>	<p><PRIVATE-KEY> – имя файла приватного ключа, задаётся строкой до 31 символа;</p> <p><ALT-NAME> – альтернативное имя сертификата, задаётся строкой от 5 до 255 символов в формате <TYPE>:<VALUE>, где:</p> <ul style="list-style-type: none"> • <TYPE> – спецификатор типа альтернативного имени, может принимать значения "IP" и "DNS". • <VALUE> – значение альтернативного имени сертификата, может принимать на сход корректно написанный IPv4-адрес или доменное имя. <p><COMMON-NAME> – общее имя, задаётся строкой от 1 до 64 символов. Чаще всего в качестве общего имени клиентского сертификата используется полное доменное имя хоста, который использует данный клиентский сертификат или ФИО пользователя, использующего клиентский сертификат;</p> <p><COUNTRY> – код страны, задаётся строкой длиной 2 символа. Рекомендуется использовать двухбуквенные обозначения стран "alpha-2" из стандарта ISO 3166-1;</p> <p><E-MAIL> – адрес электронной почты, задаётся строкой от 3 до 64 символов;</p> <p><LOCATION> – название населенного пункта или его территориальной единицы, задаётся строкой от 1 до 128 символов;</p> <p><ORGANIZATION> – название организации, задаётся строкой от 1 до 64 символов;</p> <p><ORGANIZATION-UNIT> – название подразделения организации, задаётся строкой от 1 до 64 символов;</p> <p><STATE> – название штата, области или провинции, задаётся строкой от 1 до 128 символов;</p> <p><NAME> – имя файла запроса на сертификацию, задаётся строкой до 31 символа.</p>

Шаг	Описание	Команда	Ключи
3	Выгрузить запрос на сертификацию с устройства.	<code>rtt# copy crypto:csr/<CERT-REQ> <DESTINATION></code>	<p><CERT-REQ> – имя файла запроса на сертификацию, задаётся строкой до 31 символа;</p> <p><DESTINATION> – локальное или удаленное файловое хранилище, куда будет выгружен запрос на сертификацию. Полный список возможных вариантов и синтаксис их описания подробно указан в описании команды copy в справочнике команд CLI.</p>
4	Загрузить на устройство клиентский сертификат, выписанный удостоверяющим центром.	<code>rtt# copy <SOURCE> crypto:cert/<CERT></code>	<p><SOURCE> – локальное или удаленное файловое хранилище, откуда будет загружен файл клиентского сертификата. Полный список возможных вариантов и синтаксис их описания подробно указан в описании команды copy в справочнике команд CLI;</p> <p><CERT> – имя файла клиентского сертификата, задаётся строкой до 31 символа.</p>
5	Загрузить на устройство сертификат удостоверяющего центра.	<code>rtt# copy <SOURCE> crypto:cert/<CA-CERT></code>	<p><SOURCE> – локальное или удаленное файловое хранилище, откуда будет загружен файл сертификата удостоверяющего центра. Полный список возможных вариантов и синтаксис их описания подробно указан в описании команды copy в справочнике команд CLI;</p> <p><CA-CERT> – имя файла сертификата удостоверяющего центра, задаётся строкой до 31 символа.</p>
6	Загрузить на устройство список отозванных сертификатов удостоверяющего центра.	<code>rtt# copy <SOURCE> crypto:crl/<CRL></code>	<p><SOURCE> – локальное или удаленное файловое хранилище, откуда будет загружен файл со списком отозванных сертификатов. Полный список возможных вариантов и синтаксис их описания подробно указан в описании команды copy в справочнике команд CLI;</p> <p><CRL> – имя файла со списком отозванных сертификатов, задаётся строкой до 31 символа.</p>

15.2.2. Пример ручного выпуска сертификата через внешний удостоверяющий

Сгенерировать на RTT приватный RSA ключ и сформировать запрос на сертификацию. Помимо настройки отличительного имени сертификата настроим в качестве альтернативного имени его полное доменное имя.

Для удобства дальнейшей идентификации ключей и сертификатов рекомендуется использовать в качестве имен файлов доменное имя хоста, для которого формируется ключевая пара и префикс сервиса.



При генерации ключей RSA рекомендуется использовать ключи длиной 2048 бит и более.

[illegible]

Сформируем запрос на сертификацию:

```
RTT.R1# crypto generate csr private-key r1.company.loc.key country RU state
Moscow locality Moscow organization Company common-name r1.company.loc email-
address netmaster@company.loc alternative-name DNS:r1.company.loc filename
r1.company.loc.csr
RTT.R1#
```

Выгрузим сформированный запрос на сертификацию на внешний TFTP-сервер для дальнейшей доставки в удостоверяющий центр:

```
RTT.R1# copy crypto:csr/r1.company.loc.csr tftp://10.0.113.1:/r1.company.loc.csr
|*****| 100% (1094B) Success!
RTT.R1#
```

Процесс доставки запроса на сертификацию в удостоверяющий центр и непосредственно сам выпуск сертификата индивидуален для каждого удостоверяющего центра и не рассматривается в данном разделе руководства.

После успешного выпуска сертификата удостоверяющим центром загрузим клиентский сертификат, сертификат удостоверяющего центра и список отозванных сертификатов на маршрутизатор:

```
RTT.R1# copy tftp://10.0.113.1:/ca.crt crypto:cert/ca.crt
|*****| 100% (2264B) Crypto file loaded
successfully!
RTT.R1# copy tftp://10.0.113.1:/ca.crl crypto:crl/ca.crl
|*****| 100% (1064B) Crypto file loaded
successfully!
RTT.R1# copy tftp://10.0.113.1:/r1.company.loc.crt crypto:cert/r1.company.loc.crt
|*****| 100% (1931B) Crypto file loaded
successfully!
RTT.R1#
```

Посмотреть итоговое содержимое сертификата можно командой **show crypto certificates**:

```
RTT.R1# show crypto certificates cert r1.company.loc.crt
Version: 3
Serial: 4096
Subject name:
  C(countryName): RU
  ST(stateOrProvinceName): Moscow
  O(organizationName): Company
  CN(commonName): r1.company.loc
  emailAddress(emailAddress): netmaster@company.loc
Issuer name:
  C(countryName): RU
  ST(stateOrProvinceName): Moscow
  L(localityName): Moscow
  O(organizationName): Company
  CN(commonName): Company Root Certificate Authority
Validity period:
  Valid after: 2025-10-28 03:26:41
  Invalid after: 2025-11-27 03:26:41
Signature:
  Algorithm: sha256WithRSAEncryption
```

Value:

```
37:6D:30:DE:C3:EF:D8:06:D6:4B:AA:AC:6A:78:65:C2:7C:7B:
EA:E9:F6:C0:A7:0F:9B:01:D2:C6:05:95:43:A1:C6:7B:F7:43:
F7:BE:78:7F:BA:65:73:88:31:91:C5:4F:FA:BF:41:99:D6:28:
A5:29:72:85:20:52:2E:0C:1D:3E:37:78:10:B5:CC:AE:D5:A9:
A6:79:FE:07:F4:93:E2:E9:F4:48:17:E5:A6:EB:36:D9:3E:41:
2E:8D:E9:7A:D4:75:49:A5:98:8F:76:73:8A:A7:E6:1D:89:CA:
46:B2:FC:A6:E7:96:F3:79:EB:5A:B1:B0:63:E4:AC:7C:D9:29:
AE:2D:04:4E:45:B0:08:38:7F:C6:62:72:04:C6:A2:7C:BC:77:
AF:CE:92:2F:66:75:33:8B:81:AB:98:40:61:74:9C:6B:10:15:
78:A7:58:02:DA:D1:69:C2:C2:8B:DA:66:18:BD:13:FB:4F:7D:
35:35:C3:21:6F:0A:A0:53:ED:56:F8:B9:E9:0E:6F:6D:DD:E1:
A0:AD:4A:07:97:AE:79:3C:2F:7C:E6:76:DD:9F:37:50:EB:AD:
56:3D:BA:51:D0:C1:15:25:54:F6:E2:1D:12:39:46:5F:E3:33:
1E:49:26:04:E0:23:FB:C5:FA:A8:0D:B7:16:23:C5:C9:3E:0C:
85:E2:CE:72:B5:97:0D:3D:15:D6:5D:F7:12:78:9C:84:D2:21:
C8:EC:BE:45:90:A5:CB:38:87:AB:8C:04:4B:BA:42:2B:40:95:
94:BE:F9:82:80:44:76:79:ED:42:5A:ED:28:07:E4:16:6D:C8:
80:D4:33:87:97:39:20:E9:CC:EB:F3:74:CE:F2:3C:6E:4E:C8:
37:51:21:F8:CA:AD:C6:09:3B:19:07:B2:34:3C:17:31:B8:22:
CC:BA:73:5E:9F:CD:F4:8B:38:71:BB:2F:7A:A5:F5:43:A8:8E:
07:47:36:BA:8D:BA:DB:BB:8F:9C:EB:49:A4:6C:2E:30:30:C1:
AF:06:F1:0D:E6:C7:DA:7B:FD:94:68:FD:F0:B3:3F:30:45:8C:
ED:77:FE:09:64:0E:4D:02:03:82:3A:30:61:24:08:4A:AF:BD:
C2:32:6B:70:78:E6:C1:F2:6E:2A:3E:30:1A:7A:A2:BE:70:7F:
86:8A:9B:12:D0:92:7D:14:99:72:FA:30:29:BE:44:8F:3C:D8:
75:16:AE:BD:23:97:E0:04:B5:8A:B9:71:F0:F7:15:0A:A8:95:
CC:51:23:21:6E:3F:9B:64:B1:73:A7:2F:03:22:46:6F:DD:A2:
90:A1:E4:7F:94:92:7F:E7:C2:C5:B9:F9:9D:D3:19:CF:34:3D:
D0:C0:E0:30:F8:77:1A:E8
```

Public key info:

Algorithm:	RSA
Key size:	2048
Exponent:	65537
Modulus:	

```
00:9D:41:BB:13:A8:99:9C:3E:E7:2C:0E:A5:B6:A8:CA:22:64:
```

```

BB:B9:77:E5:CE:DE:5E:71:83:9A:90:22:D1:32:E1:66:45:FC:
6C:53:DA:65:D5:FF:C7:35:2C:24:F6:BA:AD:72:DD:27:A5:09:
30:CC:AA:E3:F8:33:B5:10:1C:23:D9:EA:DA:30:6F:E4:2C:C4:
EC:08:E9:12:72:05:0C:C1:CF:6B:72:8F:B5:E8:5B:90:67:B1:
4C:59:D3:4D:CA:0C:73:94:47:F7:DB:BC:83:38:24:E2:AC:19:
DF:7D:8F:99:E0:B2:72:E3:A3:5B:7E:B8:EC:7B:6C:17:8C:48:
5A:F2:F5:A5:14:D3:07:E3:7E:5A:CD:70:6A:9E:38:2D:80:4F:
29:B3:60:F8:AC:7B:C5:09:09:B9:4B:92:D4:E0:44:5D:9E:1A:
AF:0E:25:FA:E5:73:C3:51:8F:DE:BB:F5:71:0C:2F:F3:AC:F5:
7D:79:8A:E6:87:0A:05:6A:D8:C8:6D:FE:BE:90:7D:B5:A2:3D:
3B:75:96:CF:25:98:5C:0B:F2:E3:C1:E7:B5:30:58:27:13:64:
DD:DB:77:A8:10:9C:A5:25:AC:85:DA:30:21:87:71:A8:D7:D8:
BC:60:40:C7:53:54:01:03:0E:60:5D:2B:43:99:97:F2:26:6E:
8F:F7:47:CF:9F
X509v3 Basic Constraints:
    CA: No
    Critical: Yes
X509v3 Subject key identifier:
    ID:
    40:6D:58:0E:0A:4C:CD:89:71:CA:DB:D2:BC:AD:FA:27:C9:1E:
    4D:D8
    Critical: No
X509v3 Authority key identifier:
    ID:
    7C:E6:3C:E3:FB:76:C5:18:B3:21:52:9D:8F:71:29:28:55:CA:
    96:63
    Critical: No
X509v3 Key Usage:
    Usage: Digital Signature
    Non Repudiation
    Critical: Yes
X509v3 Subject Alternative Name:
    Names: DNS:r1.company.loc
    Critical: No
RTT.R1#

```

Как можно заметить, выписанный сертификат действует 30 дней и в нем было сохранено заданное альтернативное имя. Теперь данный сертификат можно использовать в сервисах маршрутизатора.

Для корректной работы сервисов, использующих сертификаты, рекомендуется настроить на маршрутизаторе синхронизацию времени по протоколу NTP.

16. УПРАВЛЕНИЕ РЕЗЕРВИРОВАНИЕМ

16.1. Настройка VRRP

VRRP (англ. Virtual Router Redundancy Protocol) — сетевой протокол, предназначенный для увеличения доступности маршрутизаторов, выполняющих роль шлюза по умолчанию. Это достигается путём объединения группы маршрутизаторов в один виртуальный маршрутизатор и назначения им общего IP-адреса, который и будет использоваться как шлюз по умолчанию для хостов в сети.

16.1.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Перейти в режим конфигурирования интерфейса/сетевого моста, для которого необходимо настроить протокол VRRP.	<code>rtt(config)# interface <IF- TYPE><IF-NUM></code>	<IF-TYPE> – тип интерфейса; <IF-NUM> – F/S/P – F-фрейм (1), S – слот (0), P – порт.
		<code>rtt(config)# tunnel <TUN-TYPE><TUN-NUM></code>	<TUN-TYPE> – тип туннеля; <TUN-NUM> – номер туннеля.
		<code>rtt(config)# bridge <BR-NUM></code>	<BR-NUM> – номер сетевого моста.
2	Настроить необходимые параметры на интерфейсе/сетевом мосту, включая IP-адрес.		
3	Объявить номер экземпляра VRRP-процесса.	<code>rtt(config-if-gi)# vrrp <VRRP-NUM></code> <code>rtt(config-if-gi)# ipv6 vrrp <VRRP- NUM></code>	<VRRP-NUM> – номер экземпляра VRRP-процесса, принимает значения [1..255].
4	Установить виртуальный IP-адрес VRRP-маршрутизатора.	<code>rtt(config-vrrp)# ip address <ADDR/LEN> [secondary]</code>	<ADDR/LEN> – виртуальный IP-адрес и длина маски, 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 на интерфейс. secondary – ключ для установки дополнительного IP-адреса.
		<code>rtt(config-vrrp)# ipv6 ip address <IPV6-ADDR></code>	<IPV6-ADDR> – виртуальный IPv6-адрес, задаётся в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Можно указать до 8 IPv6-адресов перечислением через запятую.

Шаг	Описание	Команда	Ключи
5	Включить VRRP-экземпляр на IP-интерфейсе.	<code>rtt (config-vrrp) # enable</code>	
6	Установить приоритет VRRP-маршрутизатора (не обязательно).	<code>rtt (config-vrrp) # priority <PR></code>	<PR> – приоритет VRRP-маршрутизатора, принимает значения [1..254]. Значение по умолчанию: 100.
		<code>rtt (config-vrrp) # ipv6 priority <PR></code>	
7	Установить 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 произойдет смена ролей (не обязательно).	<code>rtt (config-vrrp) # group <GRID></code>	<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].
		<code>rtt (config-vrrp) # ipv6 group <GRID></code>	
8	Установить наследование состояний VRRP. Статус интерфейса наследника будет следовать состояниям VRRP родителя, идентификатор которого был задан (не обязательно).	<code>rtt (config-vrrp) # inherit-vrrp-id <VRID></code>	<VRID> – идентификатора VRRP-маршрутизатора, принимает значения [1..255].
		<code>rtt (config-vrrp) # inherit-vrrp-id <VRID></code>	
9	Установить IP-адрес, который будет использоваться в качестве IP-адреса отправителя для VRRP-сообщений (не обязательно).	<code>rtt (config-vrrp) # source-ip <IP></code>	<IP> – IP-адрес отправителя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		<code>rtt (config-vrrp) # ipv6 source-ip <IPV6></code>	<IPV6> – IPv6-адрес отправителя, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].
10	Установить интервал между отправкой VRRP-сообщений (не обязательно).	<code>rtt (config-vrrp) # timers advertise <TIME></code>	<TIME> – время в секундах, принимает значения [1..40].
		<code>rtt (config-vrrp) # ipv6 timers advertise <TIME></code>	Значение по умолчанию: 1 секунда.
11	Установить интервал, по истечении которого происходит отправка GratuitousARP-сообщения(ий) при переходе маршрутизатора в состояние Master (не обязательно).	<code>rtt (config-vrrp) # timers garp delay <TIME></code>	<TIME> – время в секундах, принимает значения [1..60]. Значение по умолчанию: 5 секунд.
12	Установить количество GratuitousARP-сообщений, которые будут отправлены при переходе маршрутизатора в состояние Master (не обязательно).	<code>rtt (config-vrrp) # timers garp repeat <COUNT></code>	<COUNT> – количество сообщений, принимает значения [1..60]. Значение по умолчанию: 5.

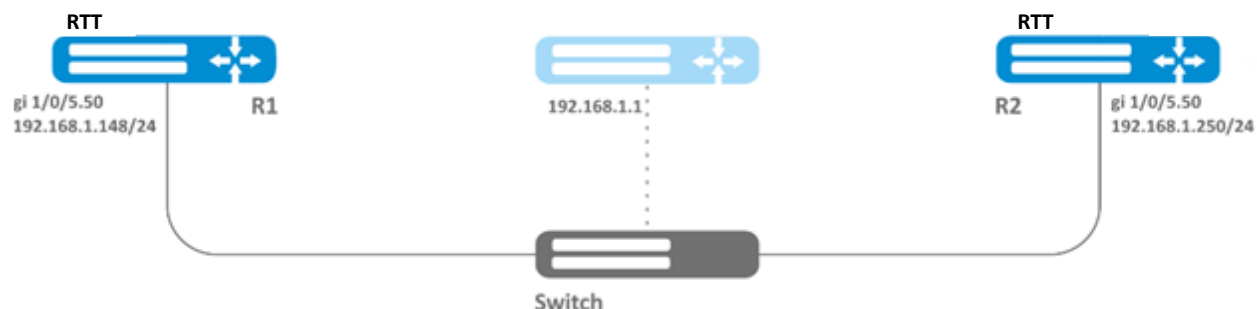
Шаг	Описание	Команда	Ключи
13	Установить интервал, по истечении которого будет происходить периодическая отправка GratuitousARP-сообщения(ий), пока маршрутизатор находится в состоянии Master (не обязательно).	<code>rtt (config-vrrp) # timers garp refresh <TIME></code>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: периодическая отправка отключена.
14	Установить количество GratuitousARP-сообщений, которые будут отправляться с периодом garpprefresh, пока маршрутизатор находится в состоянии Master (не обязательно).	<code>rtt (config-vrrp) # timers garp refresh-repeat <COUNT></code>	<COUNT> – количество сообщений, принимает значения [1..60]. Значение по умолчанию: 1.
15	Определить, будет ли Backup-маршрутизатор с более высоким приоритетом пытаться перехватить на себя роль Master у текущего Master-маршрутизатора с более низким приоритетом (не обязательно).	<code>rtt (config-vrrp) # preempt disable</code> <code>rtt (config-vrrp) # ipv6 preempt disable</code>	
16	Установить 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 (не обязательно).	<code>rtt (config-vrrp) # preempt delay <TIME></code> <code>rtt (config-vrrp) # ipv6 preempt delay <TIME></code>	<TIME> – время ожидания, определяется в секундах [1..1000]. Значение по умолчанию: 0.
17	Установить пароль для аутентификации с соседом (не обязательно).	<code>rtt (config-vrrp) # authentication key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<CLEAR-TEXT> – ключ, задаётся строкой от 1 до 8 символов; <ENCRYPTED-TEXT> – зашифрованный ключ размером от 1 до 8 байт (от 2 до 16 символов). Задаётся в шестнадцатеричном формате (0xYYYY...) или (YYYY...).
18	Определить алгоритм аутентификации (не обязательно).	<code>rtt (config-vrrp) # authentication algorithm <ALGORITHM></code>	<ALGORITHM> – алгоритм аутентификации: <ul style="list-style-type: none">• cleartext – пароль, передается открытым текстом;• md 5 – пароль хешируется по алгоритму md5.
19	Задать версию VRRP-протокола (не обязательно).	<code>rtt (config-vrrp) # version <VERSION></code>	<VERSION> – версия VRRP-протокола: 2, 3.

Шаг	Описание	Команда	Ключи
20	Установить режим, когда vrrp IP-адрес остается в состоянии UP вне зависимости от состояния самого интерфейса (не обязательно).	<code>rtr(config-vrrp)# force-up</code>	
21	Определить задержку между установлением ipv6 vrrp состояния MASTER и началом рассылки ND-сообщений (не обязательно).	<code>rtr(config-vrrp)# ipv6 timers nd delay <TIME></code>	<TIME> – время в секундах, принимает значения [1..60]. Значение по умолчанию: 5.
22	Определить период обновления информации протокола ND для ipv6 vrrp в состоянии MASTER (не обязательно).	<code>rtr(config-vrrp)# ipv6 timers nd refresh <TIME></code>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 5.
23	Определить количество ND сообщений, отправляемых за период обновления для ipv6 vrrp в состоянии MASTER (не обязательно).	<code>rtr(config-vrrp)# ipv6 timers nd refresh-repeat <NUM></code>	<NUM> – количество, принимает значения [1..60]. Значение по умолчанию: 0.
24	Определить количество отправок ND-пакетов после установки ipv6 vrrp в состоянии MASTER (не обязательно).	<code>rtr(config-vrrp)# ipv6 timers nd repeat <NUM></code>	<NUM> – количество, принимает значения [1..60]. Значение по умолчанию: 1.

16.1.2. Пример настройки 1

Задача:

Организовать виртуальный шлюз для локальной сети в VLAN 50, используя протокол VRRP. В качестве локального виртуального шлюза используется IP-адрес 192.168.1.1.



Решение:

Предварительно нужно выполнить следующие действия:

- создать соответствующий саб-интерфейс;
- настроить зону для саб-интерфейса;
- указать IP-адрес для саб-интерфейса.

Основной этап конфигурирования:

Настроим маршрутизатор R1.

В созданном саб-интерфейсе настроим VRRP. Укажем уникальный идентификатор VRRP:

```
R1(config)#interface gi 1/0/5.50
R1(config-if-sub)# vrrp 10
```

Укажем IP-адрес виртуального шлюза 192.168.1.1/24:

```
R1(config-vrrp)# ip address 192.168.1.1/24
```

Включим VRRP:

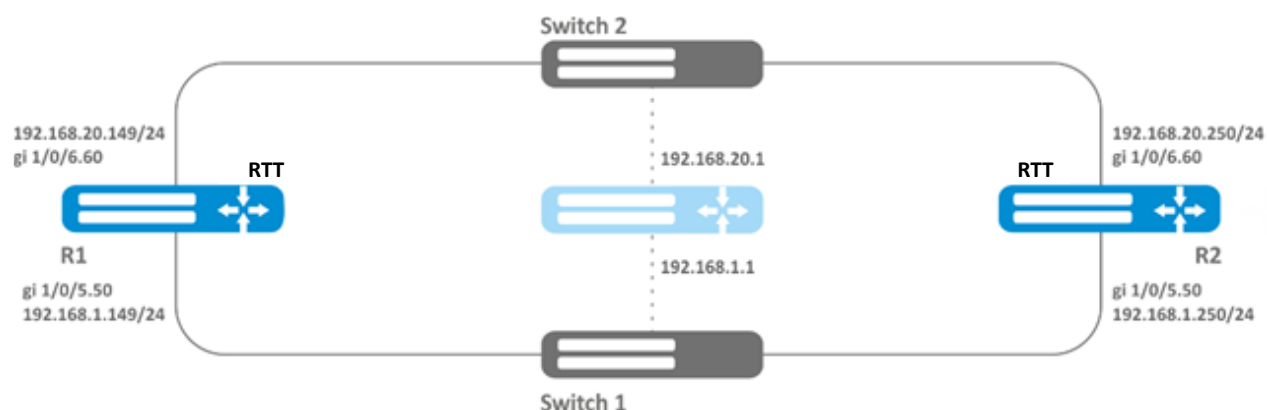
```
R1(config-vrrp)# enable
R1(config-vrrp)# exit
```

После чего необходимо произвести аналогичные настройки на R2.

16.1.3. Пример настройки 2

Задача:

Организовать виртуальные шлюзы для подсети 192.168.1.0/24 в VLAN 50 и подсети 192.168.20.0/24 в VLAN 60, используя протокол VRRP с функцией синхронизации мастера. Для этого используем объединение VRRP-процессов в группу. В качестве виртуальных шлюзов используются IP-адреса 192.168.1.1 и 192.168.20.1.



Решение:

Предварительно нужно выполнить следующие действия:

- создать соответствующие саб-интерфейсы;
- настроить зону для саб-интерфейсов;
- указать IP-адреса для саб-интерфейсов.

Основной этап конфигурирования:

Настроим маршрутизатор R1.

Настроим VRRP для подсети 192.168.1.0/24 в созданном саб-интерфейсе.

Укажем уникальный идентификатор VRRP:

```
R1(config-sub)#interface gi 1/0/5.50
R1(config-if-sub)# vrrp 10
```

Укажем IP-адрес виртуального шлюза 192.168.1.1/24:

```
R1(config-vrrp)# ip address 192.168.1.1/24
```

Укажем идентификатор VRRP-группы:

```
R1(config-vrrp)# group 5
```

Включим VRRP:

```
R1(config-vrrp)# enable
R1(config-vrrp)# exit
```

Настроим VRRP для подсети 192.168.20.0/24 в созданном саб-интерфейсе.

Укажем уникальный идентификатор VRRP:

```
R1(config-sub)#interface gi 1/0/6.60
R1(config-if-sub)# vrrp 20
```

Укажем IP-адрес виртуального шлюза 192.168.20.1/24:

```
R1(config-vrrp)# ip address 192.168.20.1/24
```

Укажем идентификатор VRRP-группы:

```
R1(config-vrrp)# group 5
```

Включим VRRP:

```
R1(config-vrrp)# enable
R1(config-vrrp)# exit
```

Произвести аналогичные настройки на R2.



Помимо создания туннеля необходимо в firewall разрешить протокол VRRP (112).



При использовании IPsec с VRRP рекомендуется настраивать DPD для ускорения перестроения IPsec-туннеля.

16.2. Настройка tracking

Tracking — механизм, позволяющий активировать сущности в зависимости от состояния VRRP/IP-SLA/туннеля/интерфейса.

16.2.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Настроить VRRP согласно разделу Настройка VRRP , настроить SLA по инструкции Настройка SLA, или сконфигурировать туннель/интерфейс.		
2	Добавить в систему Tracking-объект и перейти в режим настройки параметров Tracking-объекта.	<code>rtt(config)# track <ID></code>	<ID> – номер Tracking-объекта, принимает значения [1..100].
3	Задать правило слежения, на основании которых Tracking-объект будет переходить в активное состояние.	<code>rtt(config-track)# track vrrp id <VRID> state [not] { master backup fault } [vrf <VRF>]</code>	<VRID> – идентификатор отслеживаемого VRRP-маршрутизатора, принимает значения [1..255]; <VRF> – имя экземпляра VRF, задается строкой до 31 символа.
		<code>rtt(config-track)# track sla test <NUM> [mode <MODE>]</code>	<NUM> – номер SLA-теста, задается в диапазоне [1..10000]; <MODE> – режим слежения за sla-тестом, может принимать значения: <ul style="list-style-type: none"> • state success – отслеживается успешное состояние sla-теста; • state fail – отслеживается провальное состояние sla-теста; • reachability – отслеживаются состояние канала связи, по которому осуществляется sla-тест. При указании команды без аргумента mode, по-умолчанию устанавливается значение mode state success.

Шаг	Описание	Команда	Ключи
		<pre>rtt(config-track) # track { interface <IF> tunnel <TUN> } [state <STATE>]</pre>	<p><IF> – имя IP-интерфейса, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;</p> <p><TUN> – имя туннеля, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;</p> <p><STATE> – режим слежения за sla-тестом, может принимать значения:</p> <ul style="list-style-type: none"> • up – административное состояние "Up"; • down – административное состояние "Down". <p>При указании команды без аргумента state по умолчанию устанавливается значение mode state up.</p>
4	Включить Tracking-объект.	<pre>rtt(config-track) # enable</pre>	
5	Установить задержку смены состояния отслеживаемого объекта (не обязательно).	<pre>rtt(config-track) # delay { down up } <TIME></pre>	<p><TIME> – время задержки в секундах, задается в диапазоне [1..300].</p>
6	Задать режим работы track (не обязательно).	<pre>rtt(config-track) # mode <MODE></pre>	<p><MODE> – условие нахождения объекта отслеживания в активном состоянии, принимает значения:</p> <ul style="list-style-type: none"> • and – объект будет находиться в активном состоянии, если выполняются все отслеживаемые условия; • or – объект будет находиться в активном состоянии, если выполняется хотя бы одно из отслеживаемых условий. <p>По умолчанию используется устанавливается значение mode and.</p>
7	Создать сущность на RTT, которая будет меняться в зависимости от состояния Tracking-объекта.		

7.1	Добавить возможность управления статическим IP-маршрутом к указанной подсети (не обязательно).	<pre> rtt(config)# ip route [vrf <VRF>] <SUBNET> { <NEXTHOP> [resolve] interface <IF> tunnel <TUN> wan load- balance rule <RULE> blackhole unreachable prohibit } [<METRIC>] [track <TRACK-ID>] </pre>	<p><VRF> – имя экземпляра VRF, задается строкой до 31 символа;</p> <p><SUBNET> – адрес назначения, может быть задан в следующих видах:</p> <p>AAA.BBB.CCC.DDD – IP-адрес хоста, где каждая часть принимает значения [0..255];</p> <p>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].</p> <p><NEXTHOP> – IP-адрес шлюза задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <ul style="list-style-type: none"> • 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; <p><IF> – имя IP-интерфейса, задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;</p> <p><TUN> – имя туннеля, задается в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;</p> <p><RULE> – номер правила wan, задается в диапазоне [1..50];</p> <ul style="list-style-type: none"> • 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);
-----	--	--	---

Шаг	Описание	Команда	Ключи
			<ul style="list-style-type: none"> prohibit – при указании команды, пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Communication administratively prohibited, code 13); <p>[METRIC] – метрика маршрута, принимает значения [0..255];</p> <p><TRACK-ID> – идентификатор Tracking-объекта. Если маршрут привязан к Tracking-объекту, то он появится в системе только при выполнении всех условий, заданных в объекте.</p>
7.2	Добавить возможность управления логическим состоянием интерфейса (не обязательно).	<code>rtt(config-if-gi) # shutdown track <ID></code>	<ID> – номер Tracking-объекта, принимает значения [1..100].
7.3	Добавить возможность управления приоритетом VRRP-процесса (не обязательно).	<code>rtt(config-if-gi) # vrrp priority track <ID> { <PRIO> increment <INC> decrement <DEC> }</code>	<p><ID> – номер Tracking-объекта, принимает значения в диапазоне [1..100];</p> <p><PRIO> – приоритет VRRP-процесса, который выставится, если Tracking-объект будет в активном состоянии, принимает значения в диапазоне [1..254];</p> <p><INC> – значение, на которое увеличится приоритет VRRP-процесса, если Tracking-объект будет в активном состоянии, принимает значения в диапазоне [1..254];</p> <p><DEC> – значение, на которое уменьшится приоритет VRRP-процесса, если Tracking-объект будет в активном состоянии, принимает значения в диапазоне [1..254].</p>
7.4	Добавить возможность управления Next-Хоп для пакетов, которые попадают под критерии в указанном списке доступа (ACL) (не обязательно).	<code>rtt(config-route-map-rule) # action set ip next-hop verify-availability <NEXTHOP> <METRIC> track <ID></code>	<p><NEXTHOP> – IP-адрес шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><METRIC> – метрика маршрута, принимает значения [0..255];</p> <p><ID> – номер Tracking-объекта, принимает значения [1..100].</p>

Шаг	Описание	Команда	Ключи
7.5	Добавить возможность управления атрибутом BGP AS-Path, которое будет добавляться в начало списка AS-Path в маршруте (не обязательно).	<pre> rtt(config-route-map-rule)# action set as-path prepend <AS-PATH> track <ID> [default <AS-PATH>] </pre>	<p><AS-PATH> – список номеров автономных систем, который будет добавлен к текущему значению в маршруте. Задаётся в виде AS,AS,AS, принимает значения [1..4294967295];</p> <p><ID> – номер Tracking-объекта, принимает значения [1..100].</p>
7.6	Добавить возможность управления атрибутом BGP MED в маршруте, для которого должно срабатывать правило (не обязательно).	<pre> rtt(config-route-map-rule)# action set metric bgp { <METRIC> increment <INC> decrement <DEC> } track <ID> [default { <METRIC> increment <INC> decrement <DEC> }] </pre>	<p><ID> – номер Tracking-объекта, принимает значения [1..100];</p> <p><METRIC> – значение атрибута BGP MED, принимает значения [0..4294967295];</p> <p><INC> – значение, на которое увеличится атрибут BGP MED, если Tracking-объект будет в активном состоянии. Принимает значения [0..4294967295];</p> <p><DEC> – значение, на которое уменьшится атрибут BGP MED, если Tracking-объект будет в активном состоянии. Принимает значения [0..4294967295];</p>

Шаг	Описание	Команда	Ключи
7.7	Добавить возможность управления атрибутом BGP Community в маршруте, для которого должно срабатывать правило (не обязательно).	<pre> rtt(config-route-map- rule)# action { set add remove } community { no- advertise no-export <COMMUNITY-LIST> } track <TRACK-ID> [default <COMMUNITY- LIST>] </pre>	<p><COMMUNITY-LIST> – список community, задаётся в виде AS:N,AS:N,AS:N, где AS-часть принимает значения [0..65535], N-часть принимает значения [0..65535]. Можно указать до 64 community;</p> <p><TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100];</p> <p>no-advertise – при указании команды маршруты, которые передаются с данным значением атрибута community, не должны анонсироваться другим BGP-соседям;</p> <p>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.</p>
7.8	Добавить возможность управления атрибутом BGP ExtCommunity в маршруте, для которого должно срабатывать правило (не обязательно).	<pre> rtt(config-route-map- rule)# action { set add remove } extcommunity <EXTCOMMUNITY-LIST> track <TRACK-ID> [default <EXTCOMMUNITY- LIST>] </pre>	<p><TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100];</p> <p><EXTCOMMUNITY-LIST> – список community, задаётся в виде KIND:AS:N,KIND:AS:N,KIND:AS:N, где:</p> <ul style="list-style-type: none"> • 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].

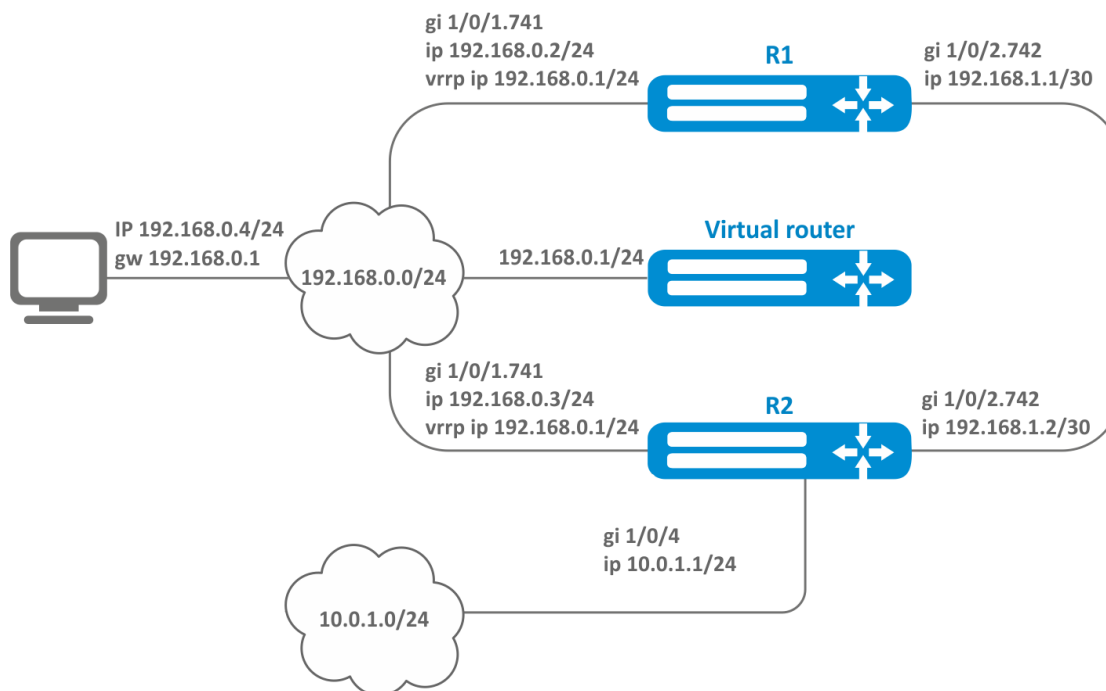
Шаг	Описание	Команда	Ключи
7.9	Добавить возможность управления атрибутом BGP Local Preference в маршруте, для которого должно срабатывать правило (не обязательно).	<pre> rtt(config-route-map- rule)# action set local-preference {<PREFERENCE> increment < VALUE > decrement < VALUE >} track <TRACK-ID> [default <PREFERENCE>] </pre>	<p><VALUE> – значение дельты изменения атрибута BGP Local Preference относительно исходного значения. Принимает значение [1..2147483647]. Если в результате применения операции increment/decrement значение метрики выйдет за допустимый диапазон, значение Local Preference принимается равным максимально или минимально допустимому значению соответственно;</p> <p><PREFERENCE> – значение атрибута BGP Local Preference, принимает значения [1..2147483647];</p> <p><TRACK-ID> – идентификатор tracking-объекта, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100].</p>
7.10	Добавить возможность управления атрибутом BGP Origin в маршруте, для которого должно срабатывать правило (не обязательно).	<pre> rtt(config-route-map- rule)# action set origin <ORIGIN> track <TRACK-ID> [default <ORIGIN>] </pre>	<p><TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100];</p> <p><ORIGIN> – значение атрибута BGP Origin, принимает следующие значения:</p> <ul style="list-style-type: none"> • egr – маршрут выучен по протоколу Exterior Gateway Protocol (EGP); • igp – маршрут получен внутри исходной автономной системы; • incomplete – маршрут выучен другим образом.
7.11	Добавить возможность управления атрибутом BGP Weight в маршруте, для которого должно срабатывать правило (не обязательно).	<pre> rtt(config-route-map- rule)# action set weight bgp {< WEIGHT > increment < VALUE > decrement < VALUE >} track <TRACK-ID> [default <WEIGHT>] </pre>	<p><WEIGHT> – значение атрибута BGP weight, принимает значения [0..65535];</p> <p><TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100].</p>
7.7	Добавить возможность управления активацией IPsec-туннеля.	<pre> rtt(config-ipsec-vpn)# enable track <ID> </pre>	<p><ID> – номер tracking-объекта, принимает значения [1..100];</p>

16.2.2. Пример настройки

Задача:

Для подсети 192.168.0.0/24 организован виртуальный шлюз 192.168.0.1/24 с использованием протокола VRRP на основе аппаратных маршрутизаторов R1 и R2. Также между маршрутизаторами R1 и R2 есть линк с вырожденной подсетью 192.168.1.0/30. Подсеть 10.0.1.0/24 терминируется только на маршрутизаторе R2. ПК имеет IP-адрес 192.168.0.4/24 и шлюз по умолчанию 192.168.0.1.

Когда маршрутизатор R1 находится в состоянии vrrp backup, трафик от ПК в подсеть 10.0.1.0/24 пойдет без дополнительных настроек. Когда маршрутизатор R1 находится в состоянии vrrp master, необходим дополнительный маршрут для подсети 10.0.1.0/24 через интерфейс 192.168.1.2.



Исходные конфигурации маршрутизаторов:

Маршрутизатор R1

```
hostname R1
interface gigabitethernet 1/0/1
  switchport forbidden default-vlan
exit
interface gigabitethernet 1/0/1.741
  ip firewall disable
  ip address 192.168.0.2/24
  vrrp 10
    ip address 192.168.0.1/24
    enable
  exit
interface gigabitethernet 1/0/2
  switchport forbidden default-vlan
```

```
exit
interface gigabitethernet 1/0/2.742
    ip firewall disable
    ip address 192.168.1.1/30
exit
```

Маршрутизатор R2

```
hostname R2
interface gigabitethernet 1/0/1
    switchport forbidden default-vlan
exit
interface gigabitethernet 1/0/1.741
    ip firewall disable
    ip address 192.168.0.3/24
    vrrp 10
        ip address 192.168.0.1/24
        enable
    exit
interface gigabitethernet 1/0/2
    switchport forbidden default-vlan
exit
interface gigabitethernet 1/0/2.742
    ip firewall disable
    ip address 192.168.1.2/30
exit
interface gigabitethernet 1/0/4
    ip firewall disable
    ip address 10.0.1.1/24
exit
```

Решение:

На маршрутизаторе R2 никаких изменений не требуется, так как подсеть 10.0.1.0/24 терминируется на нем, и в момент, когда R2 выступает в роли vrrp master, пакеты будут переданы в соответствующий интерфейс. На маршрутизаторе необходимо создать маршрут для пакетов с IP-адресом назначения из сети 10.0.1.0/24 в момент, когда R1 выступает в роли vrrp master.

Для этого создадим track-object с соответствующим условием:

```
R1(config)# track 1
R1(config-track)# track vrrp id 10 state master
R1(config-track)# enable
R1(config-track)# exit
```

Создадим статический маршрут в подсеть 10.0.1.0/24 через 192.168.1.2, который будет работать в случае удовлетворения условия из track 1:

```
R1(config)# ip route 10.0.1.0/24 192.168.1.2 track 1
```

16.3. Настройка Firewall/NAT failover

Firewall failover необходим для резервирования сессий firewall.

При включенном на устройстве firewall failover увеличивается потребление оперативной памяти на хранение firewall сессий.

16.3.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Перед настройкой сервисов резервирования необходимо настроить общие параметры failover.		
2	Переход в конфигурационное меню общих настроек failover-сервисов.	<code>rtt(config)# ip failover [vrf <VRF>]</code>	<VRF> – имя VRF, задается строкой до 31 символа.
3	Установка IP-адреса, на котором failover-сервисы принимают failover-сообщения при работе в режиме резервирования.	<code>rtt(config-failover)# local-address { <ADDR> object-group <NETWORK_OBJ_GROUP_NAME> }</code>	<ADDR> – IP-адрес, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]; <NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве local address.
4	Установка многоадресного IP-адреса, который будет использоваться для обмена информацией при работе резервирования failover-сервисов в multicast-режиме.	<code>rtt(config-failover)# multicast-address <ADDR></code>	<ADDR> – IP-адрес, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];
5	Установка идентификатора multicast-группы для обмена информацией при работе резервирования failover-сервисов в multicast-режиме.	<code>rtt(config-failover)# multicast-group <GROUP></code>	<GROUP> – multicast-группа, указывается в диапазоне [1000..9999].
6	Установка IP-адреса, на который failover-сервисы отправляют failover-сообщения при работе в режиме резервирования.	<code>rtt(config-failover)# remote-address { <ADDR> object-group <NETWORK_OBJ_GROUP_NAME> }</code>	<ADDR> – IP-адрес, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]; <NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве local address.
7	Выбор VRRP-группы, по состоянию которой будет определяться мастерство при работе failover-сервисов в режиме Active-Standby.	<code>rtt(config-failover)# vrrp-group <GRID></code>	<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].
8	Переход в конфигурационное меню настроек Firewall failover.	<code>rtt(config)# ip firewall failover [vrf <VRF>]</code>	<VRF> – имя VRF, задается строкой до 31 символа.

Шаг	Описание	Команда	Ключи
9	Выбор режима обмена информацией между маршрутизаторами.	<code>rtt(config-firewall-failover)# sync-type <MODE></code>	<p><MODE> – режим обмена информацией:</p> <ul style="list-style-type: none"> • unicast – режим unicast; • multicast – режим multicast.
10	Настройка номера UDP-порта службы резервирования сессий Firewall, через который происходит обмен информацией при работе в unicast-режиме (не обязательно).	<code>rtt(config-firewall-failover)# port <PORT></code>	<PORT> – номер порта службы резервирования сессий Firewall, указывается в диапазоне [1..65535].
11	Включение резервирования сессий Firewall.	<code>rtt(config-firewall-failover)# enable</code>	

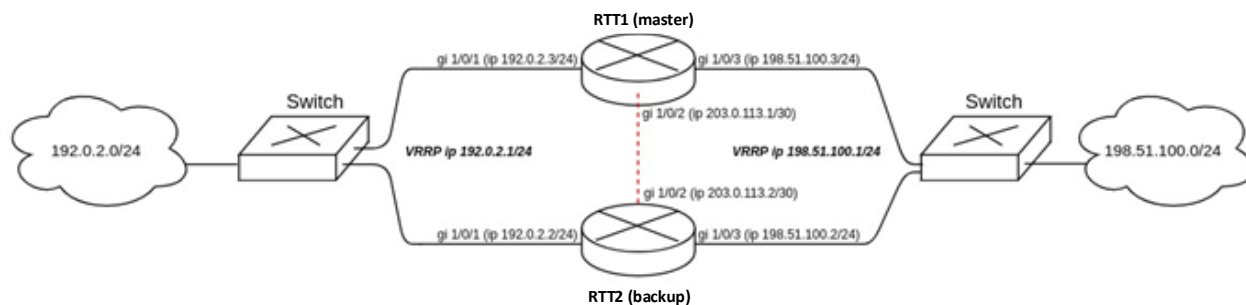


При настройке `firewall failover` также будут синхронизироваться NAT-сессии между устройствами.

16.3.2. Пример настройки

Задача:

Настроить резервирование сессий firewall для VRRP-группы в unicast-режиме. Необходимо организовать резервирование для двух подсетей с помощью протокола VRRP, синхронизировать vrrp-процессы на маршрутизаторах.



Основные этапы решения задачи:

1) Необходимо настроить vrrp-процессы на маршрутизаторах. Для master будем использовать vrrp priority 20, для backup будем использовать vrrp priority 10.

2) Необходимо настроить firewall failover в режиме unicast с номером udp-порта 3333 для VRRP-группы.

3) Необходимо настроить зону безопасности для протокола vrrp и протокола udp.

Решение:

Настроим маршрутизатор RTT-1 (master).

Предварительно на интерфейсах настроим IP-адрес и определим принадлежность к зоне безопасности.

```
master(config)# interface gigabitethernet 1/0/1
master(config-if-gi)# security-zone trusted
master(config-if-gi)# ip address 192.0.2.3/24
master(config-if-gi)# exit
master(config)# interface gigabitethernet 1/0/2
master(config-if-gi)# security-zone trusted
master(config-if-gi)# ip address 203.0.113.1/30
master(config-if-gi)# exit
master(config)# interface gigabitethernet 1/0/3
master(config-if-gi)# security-zone trusted
master(config-if-gi)# ip address 198.51.100.3/24
master(config-if-gi)# exit
```

Настроим vrrp-процессы на интерфейсах. Необходимо настроить следующие параметры на интерфейсах маршрутизатора: идентификатор VRRP, IP-адрес VRRP, приоритет VRRP, принадлежность VRRP-маршрутизатора к группе.

Также дополнительно на backup необходимо настроить vrrp preempt delay, в результате чего появится время на установление синхронизации firewall перед тем, как backup-маршрутизатор попытается перехватить мастерство.

После чего необходимо включить vrrp-процесс с помощью команды vrrp.



Вместо настройки vrrp preempt delay есть возможность выбора режима работы vrrp preempt disable, в результате которого маршрутизатор с более высоким vrrp-приоритетом не будет забирать мастерство у маршрутизатора с более низким vrrp-приоритетом после возвращения в работу.



На маршрутизаторе необходимо установить принадлежность vrrp-процессов к одной группе для синхронизации состояния vrrp-процессов (master, backup), а также для синхронизации сессий vrrp-процессов с помощью firewall failover.

```
master(config)# interface gigabitethernet 1/0/1
master(config-if-gi)# vrrp 1
master(config-vrrp)# ip address 192.0.2.1/24
master(config-vrrp)# priority 20
master(config-vrrp)# group 1
master(config-vrrp)# preempt delay 60
master(config-vrrp)# enable
```

```
master(config-vrrp)# exit
master(config)# interface gigabitethernet 1/0/3
master(config-if-gi)# vrrp 3
master(config-vrrp)# ip address 198.51.100.1/24
master(config-vrrp)# priority 20
master(config-vrrp)# group 1
master(config-vrrp)# preempt delay 60
master(config-vrrp)# enable
master(config-vrrp)# exit
```

Настроим общие параметры failover:

```
master(config)# ip failover
master(config-failover)# local-address 203.0.113.1
master(config-failover)# remote-address 203.0.113.2
master(config-failover)# vrrp-group 1
master(config-failover)# exit
```

Настроим firewall failover.

Выберем режим резервирования сессий unicast:

```
master(config)# ip firewall failover
master(config-firewall-failover)# sync-type unicast
```

Настроим номер UDP-порта службы резервирования сессий Firewall:

```
master(config-firewall-failover)# port 3333
```

Включим резервирования сессий Firewall:

```
master(config-firewall-failover)# enable
```

Для настройки правил зон безопасности потребуется создать профиль для порта firewall failover:

```
master(config)# object-group service failover
master(config-object-group-service)# port-range 3333
master(config-object-group-service)# exit
```

Дополнительно в security zone-pair trusted self необходимо разрешить следующие протоколы:

```
master(config)# security zone-pair trusted self
master(config-zone-pair)# rule 66
master(config-zone-pair-rule)# action permit
master(config-zone-pair-rule)# match protocol vrrp
master(config-zone-pair-rule)# enable
master(config-zone-pair-rule)# exit
master(config-zone-pair)# rule 67
master(config-zone-pair-rule)# action permit
master(config-zone-pair-rule)# match protocol udp
master(config-zone-pair-rule)# match destination-port object-group failover
master(config-zone-pair-rule)# enable
master(config-zone-pair-rule)# exit
master(config-zone-pair)# exit
```

Посмотреть статус vrrp-процессов есть возможность с помощью следующей команды:

```
master# show vrrp
Virtual router    Virtual IP        Priority    Preemption    State
-----
1                192.0.2.1/24     20         Enabled       Master
3                198.51.100.1/24  20         Enabled       Master
```

Посмотреть состояние резервирования сессий Firewall есть возможность с помощью следующей команды:

```
master# show ip firewall failover
Communication interface:    gigabitethernet 1/0/2
Status:                     Running
Bytes sent:                 2496
Bytes received:             640
Packets sent:              271
Packets received:          40
Send errors:               0
Receive errors:            0
```

Посмотреть состояние систем резервирования устройства есть возможность с помощью следующей команды:

```
master# show high-availability state
AP Tunnels:
  State:                    Disabled
  Last state change:       --
DHCP server:
  State:                    Disabled
  Last state change:       --
Firewall sessions:
  State:                    successful synchronization
  Last synchronization:    09:38:00 05.08.2021
```

Настроим маршрутизатор RTT-2 (backup).

Настройка интерфейсов:

```
backup(config)# interface gigabitethernet 1/0/1
backup(config-if-gi)# security-zone trusted
backup(config-if-gi)# ip address 192.0.2.2/24
backup(config-if-gi)# vrrp 1
backup(config-vrrp)# ip address 192.0.2.1/24
backup(config-vrrp)# priority 10
backup(config-vrrp)# group 1
backup(config-vrrp)# enable
backup(config-vrrp)# exit
backup(config)# interface gigabitethernet 1/0/2
backup(config-if-gi)# security-zone trusted
backup(config-if-gi)# ip address 203.0.113.2/30
backup(config-if-gi)# exit
backup(config)# interface gigabitethernet 1/0/3
backup(config-if-gi)# security-zone trusted
backup(config-if-gi)# ip address 198.51.100.2/24
```

```

backup(config-if-gi)# vrrp 3
backup(config-vrrp)# ip address 198.51.100.1/24
backup(config-vrrp)# priority 10
backup(config-vrrp)# group 1
backup(config-vrrp)# enable
backup(config-vrrp)# exit

```

Настройка firewall failover:

```

backup(config)# ip failover
backup(config-failover)# local-address 203.0.113.2
backup(config-failover)# remote-address 203.0.113.1
backup(config-failover)# vrrp-group 1
backup(config-failover)# exit
backup(config)# ip firewall failover
backup(config-firewall-failover)# sync-type unicast
backup(config-firewall-failover)# port 3333
backup(config-firewall-failover)# enable

```

Настройка зоны безопасности аналогична настройке на маршрутизаторе RTT-1 (master).

16.4. Настройка DHCP failover

DHCP failover используется для резервирования базы IP-адресов, которые были динамически выданы в процессе работы DHCP-server.

16.4.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Перед настройкой сервисов резервирования необходимо настроить общие параметры failover.		
2	Переход в конфигурационное меню общих настроек failover-сервисов.	<code>rtt(config)# ip failover [vrf <VRF>]</code>	<VRF> – имя VRF, задается строкой до 31 символа;
3	Установка IP-адреса, на котором failover-сервисы принимают failover-сообщения при работе в режиме резервирования.	<code>rtt(config-failover)# local-address { <ADDR> object-group <NETWORK_OBJ_GROUP_NAME> }</code>	<ADDR> – IP-адрес, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]; <NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве local address.

Шаг	Описание	Команда	Ключи
4	Установка IP-адреса, на который failover сервисы отправляют failover-сообщения при работе в режиме резервирования.	<pre>rtt(config-failover)# remote-address { <ADDR> object-group <NETWORK_OBJ_GROUP_NAME> }</pre>	<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]; <NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве local address.
5	Выбор VRRP-группы, по состоянию которой будет определяться мастерство при работе failover сервисов в режиме Active-Standby.	<pre>rtt(config-failover)# vrrp-group <GRID></pre>	<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].
6	Переход в конфигурационное меню DHCP failover для его настройки.	<pre>rtt(config)# ip dhcp- server failover [vrf <VRF>]</pre>	<VRF> – имя VRF, задается строкой до 31 символа;
7	Выбор режима работы DHCP failover.	<pre>rtt(config-dhcp-server- failover)# mode { active- active active-standby }</pre>	active-active – режим работы с двумя активными маршрутизаторами; active-standby – режим работы с одним активным маршрутизатором и одним резервным.
8	Настройка роли DHCP failover, при работе резервирования в режиме Active-Active.	<pre>rtt(config-dhcp-server- failover)# role <ROLE></pre>	<ROLE> – роль DHCP-сервера при работе в режиме резервирования: <ul style="list-style-type: none"> primary – режим активного DHCP-сервера; secondary – режим резервного DHCP-сервера.
9	Включение резервирования DHCP failover.	<pre>rtt(config-dhcp-server- failover)# enable</pre>	

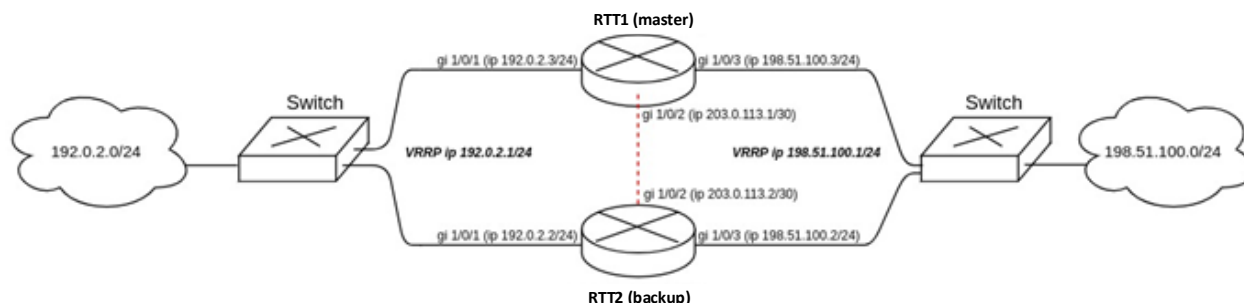


Режим active-standby не поддерживается в VRF.

16.4.2. Пример настройки

Задача:

Настроить резервирование DHCP-сервера в режиме Active-Standby. Необходимо организовать резервирование для двух подсетей с помощью протокола VRRP, синхронизировать vrrp-процессы на маршрутизаторах.



Основные этапы решения задачи:

1. Необходимо настроить vrrp-процессы на маршрутизаторах. Для master будем использовать vrrp priority 20, для backup будем использовать vrrp priority 10.
2. Необходимо настроить DHCP failover в режиме Active-Standby.
3. Необходимо настроить зону безопасности для протоколов vrrp, udp и tcp.

Решение:

1. Настройка маршрутизатора RTT-1 (master).

Предварительно на интерфейсах настроим IP-адрес и определим принадлежность к зоне безопасности.

```
master(config)# interface gigabitethernet 1/0/1
master(config-if-gi)# security-zone trusted
master(config-if-gi)# ip address 192.0.2.3/24
master(config-if-gi)# exit
master(config)# interface gigabitethernet 1/0/2
master(config-if-gi)# security-zone trusted
master(config-if-gi)# ip address 203.0.113.1/30
master(config-if-gi)# exit
master(config)# interface gigabitethernet 1/0/3
master(config-if-gi)# security-zone trusted
master(config-if-gi)# ip address 198.51.100.3/24
master(config-if-gi)# exit
```

Настроим vrrp-процессы на интерфейсах. Необходимо настроить следующие параметры на интерфейсах маршрутизатора: идентификатор VRRP, IP-адрес VRRP, приоритет VRRP, принадлежность VRRP-маршрутизатора к группе.

После чего необходимо включить vrrp-процесс с помощью команды "vrrp".



Вместо настройки vrrp preempt delay есть возможность выбора режима работы vrrp preempt disable, в результате которого маршрутизатор с более высоким vrrp-

приоритетом не будет забирать мастерство у маршрутизатора с более низким vrrp-приоритетом после возвращения в работу.



На маршрутизаторе необходимо установить принадлежность vrrp-процессов к одной группе для синхронизации состояния vrrp-процессов (master, backup).

```
master(config)# interface gigabitethernet 1/0/1
master(config-if-gi)# vrrp 1
master(config-vrrp)# ip address 192.0.2.1/24
master(config-vrrp)# priority 20
master(config-vrrp)# group 1
master(config-vrrp)# enable
master(config-vrrp)# exit
master(config)# interface gigabitethernet 1/0/3
master(config-if-gi)# vrrp 3
master(config-vrrp)# ip address 198.51.100.1/24
master(config-vrrp)# priority 20
master(config-vrrp)# group 1
master(config-vrrp)# enable
master(config-vrrp)# exit
```

Настроим DHCP failover. Для DHCP failover необходимо настроить следующие параметры: mode, local-address, remote-address, принадлежность VRRP-маршрутизатора к группе.

```
master(config)# ip dhcp-server pool LAN
master(config-dhcp-server)# network 192.0.2.0/24
master(config-dhcp-server)# address-range 192.0.2.10-192.0.2.20
master(config-dhcp-server)# exit
master(config)# ip dhcp-server
master(config)# ip failover
master(config-failover)# local-address 203.0.113.1
master(config-failover)# remote-address 203.0.113.2
master(config-failover)# vrrp-group 1
master(config-failover)# exit
master(config)# ip dhcp-server failover
master(config-dhcp-server-failover)# mode active-standby
master(config-dhcp-server-failover)# enable
master(config-dhcp-server-failover)# exit
```



Для запуска DHCP failover необходимо предварительно настроить и включить DHCP-server, который будет резервироваться.

Для настройки правил зон безопасности потребуется создать профиль для порта DHCP failover:

```
master(config)# object-group service dhcp_failover
master(config-object-group-service)# port-range 873
master(config-object-group-service)# exit
```



DHCP failover для синхронизации использует TCP-порт 873, его необходимо разрешить при настройке firewall.

Дополнительно в security zone-pair trusted self необходимо разрешить следующие протоколы:

Маршрутизаторы серии RTT. Руководство по эксплуатации

```

master(config)# security zone-pair trusted self
master(config-zone-pair)# rule 66
master(config-zone-pair-rule)# action permit
master(config-zone-pair-rule)# match protocol vrrp
master(config-zone-pair-rule)# enable
master(config-zone-pair-rule)# exit
master(config-zone-pair)# rule 67
master(config-zone-pair-rule)# action permit
master(config-zone-pair-rule)# match protocol tcp
master(config-zone-pair-rule)# match destination-port object-group dhcp_failover
master(config-zone-pair-rule)# enable
master(config-zone-pair-rule)# exit
master(config-zone-pair)# rule 68
master(config-zone-pair-rule)# action permit
master(config-zone-pair-rule)# match protocol udp
master(config-zone-pair-rule)# enable
master(config-zone-pair-rule)# exit

```

Посмотреть статус vrrp-процессов есть возможность с помощью следующей команды:

```

master# show vrrp

```

Virtual router	Virtual IP	Priority	Preemption	State
1	192.0.2.1/24	20	Enabled	Master
3	198.51.100.1/24	20	Enabled	Master

Посмотреть состояние резервирования сессий Firewall есть возможность с помощью следующей команды:

```

master# show ip dhcp server failover
VRF:      --
State: Successful

```

Посмотреть состояние систем резервирования устройства есть возможность с помощью следующей команды:

```

master# show high-availability state
AP Tunnels:
  State: Disabled
  Last state change: --
DHCP option 82 table:
  State: Disabled
  Last state change: --
DHCP server:
VRF:
  State: Successful synchronization
  State: Disabled
  Last synchronization: --

```



Для успешной синхронизации сервиса DHCP failover на устройствах должно быть выставлено идентичное время.

2. Настройка маршрутизатора RTT-2 (backup).

Настройка интерфейсов:

```
backup(config)# interface gigabitethernet 1/0/1
backup(config-if-gi)# security-zone trusted
backup(config-if-gi)# ip address 192.0.2.2/24
backup(config-if-gi)# vrrp 1
backup(config-vrrp)# ip address 192.0.2.1/24
backup(config-vrrp)# priority 20
backup(config-vrrp)# group 1
backup(config-vrrp)# enable
backup(config-vrrp)# exit
backup(config)# interface gigabitethernet 1/0/2
backup(config-if-gi)# security-zone trusted
backup(config-if-gi)# ip address 203.0.113.2/30
backup(config-if-gi)# exit
backup(config)# interface gigabitethernet 1/0/3
backup(config-if-gi)# security-zone trusted
backup(config-if-gi)# ip address 198.51.100.2/24
backup(config-if-gi)# vrrp 3
backup(config-vrrp)# ip address 198.51.100.1/24
backup(config-vrrp)# priority 10
backup(config-vrrp)# group 1
backup(config-vrrp)# enable
backup(config-vrrp)# exit
```

Настройка DHCP failover:

```
backup(config)# ip dhcp-server pool LAN
backup(config-dhcp-server)# network 192.0.2.0/24
backup(config-dhcp-server)# address-range 192.0.2.10-192.0.2.20
backup(config-dhcp-server)# exit
backup(config)# ip dhcp-server
backup(config)# ip failover
backup(config-failover)# local-address 203.0.113.2
backup(config-failover)# remote-address 203.0.113.1
backup(config-failover)# vrrp-group 1
backup(config-failover)# exit
backup(config)# ip dhcp-server failover
backup(config-dhcp-server-failover)# mode active-standby
backup(config-dhcp-server-failover)# enable
backup(config-dhcp-server-failover)# exit
```

Настройка зоны безопасности аналогична настройке на маршрутизаторе RTT-1 (master).

17.УПРАВЛЕНИЕ КЛАСТЕРИЗАЦИЕЙ

17.1. Настройка кластера

Кластер используется для резервирования устройств в сети. Резервирование обеспечивается за счет синхронизации работы различных сервисов между устройствами, а также за счет организации единой конфигурации и единой точки управления устройствами.

17.1.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Сменить юнит у устройства, при необходимости. (смена юнита устройства вступает в силу после перезагрузки)	<code>rtt# set unit id <ID></code>	<ID> – номер юнита, принимает значения [1..4].
2	Создать VLAN, который будет использоваться в кластерном интерфейсе. Можно также использовать vlan 1, созданный по умолчанию.	<code>rtt(config)# vlan <VID></code>	<VID> – идентификатор VLAN, задаётся в диапазоне [2..4094].
3	Перейти в режим конфигурирования физических интерфейсов, которые будут использованы для работы кластерного интерфейса. Необходимо настроить интерфейсы всех юнитов, которые будут участвовать в кластере.	<code>rtt(config)# interface gigabitethernet</code> <code>rtt(config)# interface tengigabitethernet</code> <code>rtt(config)# interface fortygigabitethernet</code> <code>rtt(config)# interface twentyfivegigabitethernet</code>	
4	Установить режим работы физических интерфейсов.	<code>rtt(config-if-gi)# mode switchport</code>	Допустимо для всех моделей.
		<code>rtt(config-if-gi)# mode hybrid</code>	Допустимо только для R800
5	Задать режим работы L2-интерфейсов.	<code>rtt(config-if-gi)# switchport mode access</code>	Только для R100/200. Данный режим является режимом по умолчанию и не отображается в конфигурации.
		<code>rtt(config-if-gi)# switchport mode trunk</code>	Только для R100/200.

Шаг	Описание	Команда	Ключи
		<code>rtt(config-if-gi)# switchport mode general</code>	Только для R800. Данный режим является режимом по умолчанию и не отображается в конфигурации.
6	Настроить заранее созданный VLAN на интерфейсах.	<code>rtt(config-if-gi)# switchport access vlan <VID></code>	Только для R100/200. <VID> – идентификационный номер VLAN, задаётся в диапазоне [1...4094].
		<code>rtt(config-if-gi)# switchport trunk allowed vlan <ACT> <VID></code>	Для R100/200. <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 «,».

Шаг	Описание	Команда	Ключи
		<pre> rtt(config-if-gi)# switchport general allowed vlan <ACT> <VID> [<TYPE>] </pre>	<p>Для R800.</p> <p><ACT> – назначаемое действие:</p> <p>add – включение интерфейса во VLAN;</p> <p>remove – исключение интерфейса из VLAN.</p> <p><VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094]. Можно задать диапазоном через «-» или перечислением через «,»;</p> <p><TYPE> – тип пакета:</p> <p>tagged – интерфейс будет передавать и принимать пакеты в указанных VLAN тегированными;</p> <p>untagged – интерфейс будет передавать пакеты в указанных VLAN нетегированными.</p> <p>VLAN, в который будут направлены входящие нетегированные пакеты, настраивается командой switchport general pvid.</p>
7	Перейти в режим конфигурирования сетевого моста, который будет использован в качестве кластерного интерфейса.	<pre> rtt(config)# bridge <BR- NUM> </pre>	<BR-NUM> – номер сетевого моста.
8	Настроить заранее созданный VLAN на кластерном интерфейсе.	<pre> rtt(config-bridge)# vlan <VID> </pre>	<VID> – идентификационный номер VLAN, задаётся в диапазоне [1...4094].
9	Указать IPv4-адрес и маску подсети для кластерного интерфейса. Необходимо установить адреса для всех юнитов кластера. (для работы кластерного интерфейса поддерживается только IPv4-адресация)	<pre> rtt(config-bridge)# ip address <ADDR/LEN> [unit <ID>] </pre>	<p><ADDR/LEN> – IP-адрес и длина маски подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].</p> <p><ID> – номер юнита, принимает значения [1..4].</p> <p>Дополнительные функции IPv4-адресации см. в документе «Справочник команд CLI».</p>

Шаг	Описание	Команда	Ключи
10	Установить номер экземпляра VRRP-процесса.	<code>rtt(config-bridge) # vrrp <VRRP-NUM></code>	<VRRP-NUM> – номер экземпляра VRRP-процесса, принимает значения [1..255]. Дополнительные функции VRRP-процесса см. в разделе Управление резервированием .
11	Установить виртуальный IP-адрес VRRP-маршрутизатора (адрес должен быть из той же подсети, что и ip address юнитов).	<code>rtt(config-vrrp) # ip address <ADDR/LEN> [secondary]</code>	<ADDR/LEN> – виртуальный IP-адрес и длина маски, 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 на интерфейс. secondary – ключ для установки дополнительного IP-адреса.
12	Установить 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 произойдёт смена ролей.	<code>rtt(config-vrrp) # group <GRID></code>	<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32]
13	Включить VRRP-процесс на IP-интерфейсе.	<code>rtt(config-vrrp) # enable</code>	
14	Активировать сетевой мост.	<code>rtt(config-bridge) # enable</code>	
15	Перейти в режим конфигурирования кластера.	<code>rtt(config) # cluster</code>	
16	Установить интерфейс, через который будет происходить обмен служебными сообщениями между юнитами в кластере.	<code>rtt(config-cluster) # cluster-interface bridge [<BRIDGE-ID>]</code>	<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора .

Шаг	Описание	Команда	Ключи
17	Отключить синхронизацию конфигураций в кластере между юнитами (не обязательно).	<code>rtt(config-cluster)# sync config disable</code>	
18	Перейти в режим конфигурирования юнита в кластере.	<code>rtt(config-cluster)# unit <ID></code>	<ID> – номер юнита, принимает значения [1..4].
19	Настроить для юнита соответствующий системный MAC-адрес устройства.	<code>rtt(config-cluster- unit)# mac-address <ADDR></code>	<ADDR> – MAC-адрес сетевого моста, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].
20	Включить работу кластера.	<code>rtt(config-cluster)# enable</code>	



Данные между юнитами кластера через канал синхронизации передаются в открытом виде. Также все вводимые команды конфигурирования, содержащие чувствительную информацию не в encrypted-виде, будут переданы в том же виде, в котором введены, после чего будут преобразованы в encrypted-вид.

17.1.2. Пример настройки кластера

В настоящем руководстве приведено описание настройки кластера для администратора сервисного маршрутизатора RTT (далее — маршрутизатор).

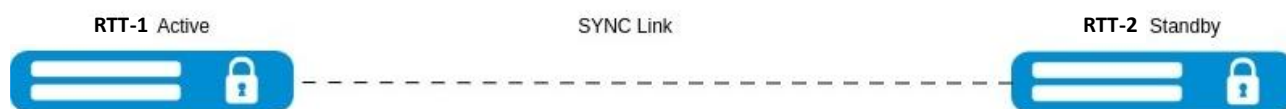


Схема реализации HA Cluster из 2 юнитов

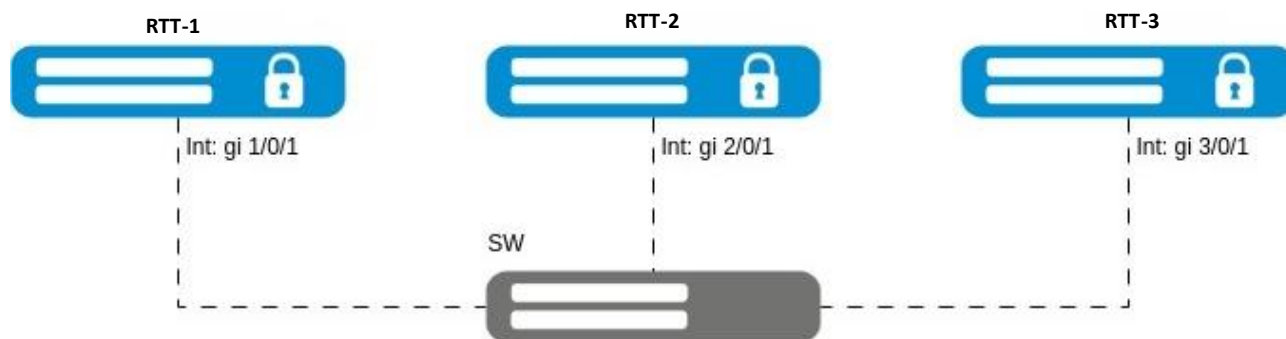


Схема реализации HA Cluster из 3 юнитов

В примере настройки кластера будет рассмотрен HA Cluster из 2 юнитов.

Для настройки более чем 2 юнитов в кластере необходимо дополнить конфигурацию юнитизированными командами по аналогии с указанным примером.

17.1.2.1. Первичная настройка кластера

Для начала работы необходимо полностью настроить одно устройство из кластера.

После включения устройства примените конфигурацию по умолчанию на устройствах, предназначенных для объединения в кластер:

RTT-1

```
rtt# copy system:default-config system:candidate-config
Entire candidate configuration will be reset to default, all settings will be
lost upon commit.
Do you really want to continue? (y/N): y
|*****| 100% (59B) Default configuration
loaded successfully.
```

Для более удобного и ясного восприятия рекомендуется переименовать устройства. В кластерной версии прошивки предусмотрена возможность указать имя устройства с привязкой к юниту. Устройство будет использовать только тот hostname, юнитом которого он является:

RTT-1

```
rtt# configure
rtt(config)# hostname RTT-1 unit 1
rtt(config)# hostname RTT-2 unit 2
```

В конфигурации может одновременно находиться hostname с unit и hostname без unit.

Более приоритетным является **hostname**, указанный с привязкой к **unit**.

Чтобы изменить юнит устройства, выполните следующие команды:

RTT-1

```
RTT-1# set unit id 1
Unit ID will be 1 after reboot
RTT-1# reload system
Do you really want to reload system now? (y/N): y
```

Смена юнита устройства вступает в силу после перезагрузки.

При изменении номера юнита маршрутизатора не происходит автоматической конвертации конфигурации. В случае если до маршрутизатора настроен удаленный доступ и у него меняется номер юнита, необходимо до перезагрузки настроить ip-интерфейсы для нового юнита аналогично текущим.

В заводской конфигурации присутствуют настройки интерфейсов только для юнита по умолчанию (unit = 1).

При копировании и применении заводской конфигурации настройка номера юнита не изменяется на значение по умолчанию.

Установить номер юнита по умолчанию возможно следующими способами:

1. используя консольное подключение;
2. зажав функциональную кнопку "F" на 15 секунд.

Убедитесь в том, что настройка юнита применилась успешно:

RTT-1

```
RTT-1# show unit id
Unit ID is 1
Unit ID will be 1 after reboot
```

Объединение устройств в кластер невозможно, если они относятся к одному и тому же юниту. Исключение — процесс ZTP, так как в процессе ZTP нужный unit у устройства выставится автоматически.

Для объединения в кластер vRTT предварительно необходимо сделать разные системные MAC-адреса на устройствах путем смены серийного номера.

17.1.2.2. Настройка кластерного интерфейса

Для полноценной работы кластера требуется сконфигурировать кластерный интерфейс, который будет использоваться для передачи control plane трафика. В качестве кластерного интерфейса назначен bridge. В качестве механизма, отвечающего за определение ролей устройств, участвующих в резервировании, назначен протокол VRRP. Настройки cluster-интерфейса должны быть идентичны для всех участников кластера.

Так как кластер выполняет синхронизацию состояний между устройствами, необходимо создать зону безопасности SYNC (synchronization):

RTT-1

```
RTT-1(config)# security zone SYNC
RTT-1(config-security-zone)# exit
```

Далее перейдите к настройкам кластерного интерфейса:

RTT-1


```
RTT-1(config)# bridge 1
```

В текущей версии ПО в качестве cluster-интерфейса поддержан только bridge.

Укажите, к какому VLAN относится bridge, и зону безопасности:

RTT-1

```
RTT-1(config-bridge)# vlan 1
RTT-1(config-bridge)# security-zone SYNC
```

Далее укажите IP-адреса:

RTT-1

```
RTT-1(config-bridge)# ip address 198.51.100.254/24 unit 1
RTT-1(config-bridge)# ip address 198.51.100.253/24 unit 2
```

Для работы кластерного интерфейса поддерживается только IPv4-адресация.

На cluster-интерфейсе необходима настройка адресов с привязкой к unit. Количество настраиваемых адресов зависит от количества настраиваемых участников кластера.

Настройте идентификатор VRRP, принадлежность VRRP-маршрутизатора к группе, IP-адрес VRRP:

RTT-1

```
RTT-1(config-bridge)# vrrp 1
RTT-1(config-vrrp)# group 1
RTT-1(config-vrrp)# ip address 198.51.100.1/24
```

Адрес VRRP должен быть из той же подсети, что и адреса на интерфейсе.

Также на VRRP-интерфейсе можно назначить разные приоритеты для разных юнитов.

RTT-1

```
RTT-1(config-vrrp)# priority 254 unit 1
RTT-1(config-vrrp)# priority 253 unit 2
```

Включите протокол VRRP и bridge:

RTT-1

```
RTT-1(config-vrrp)# enable
RTT-1(config-vrrp)# exit
RTT-1(config-bridge)# enable
RTT-1(config-bridge)# exit
```

Настройте физические порты для выделенного линка синхронизации маршрутизаторов RTT-1 и RTT-2:

RTT-1

```
RTT-1(config)# interface gigabitethernet 1/0/1
RTT-1(config-if-gi)# description "Network: SYNC"
RTT-1(config-if-gi)# mode switchport
RTT-1(config-if-gi)# exit
RTT-1(config)# interface gigabitethernet 2/0/1
RTT-1(config-if-gi)# description "Network: SYNC"
RTT-1(config-if-gi)# mode switchport
RTT-1(config-if-gi)# exit
```

Для проверки работы протокола VRRP выполните следующую команду:

RTT-1

```
rtt# show vrrp
Virtual router      Virtual IP          Priority  Preemption  State   Inherit  Sync
group ID
-----
1                  198.51.100.1/24      100      Disabled    Backup  --       1
```

Можно увидеть, что устройство приняло состояние Backup. Через 10 секунд устройство примет состояние Master.

17.1.2.3. Настройка кластера

Для запуска кластера необходимо указать заранее настроенный кластерный интерфейс и юниты, которые будут выполнять роли Active и Standby.

Перейдите в режим настройки кластера:

RTT-1

```
RTT-1(config)# cluster
```

Настройте юниты:

RTT-1

```
RTT-1(config-cluster)# unit 1
RTT-1(config-cluster-unit)# mac-address E4:5A:D4:A0:BE:35
RTT-1(config-cluster-unit)# exit
RTT-1(config-cluster)# unit 2
RTT-1(config-cluster-unit)# mac-address A8:F9:4B:AF:35:84
RTT-1(config-cluster-unit)# exit
```

В качестве mac-address указывается системный MAC-адрес устройства, его можно узнать с помощью команды **show system | include MAC**.

Укажите кластерный интерфейс, созданный ранее, и активируйте кластер:

RTT-1

```
RTT-1(config-cluster)# cluster-interface bridge 1
RTT-1(config-cluster)# enable
RTT-1(config-cluster)# exit
```

Первое устройство полностью настроено и готово к работе.

Аналогичные настройки необходимо произвести на втором устройстве, предварительно сменив у него юнит на требуемый. Также возможна настройка второго устройства средствами ZTP.

Для активации процесса ZTP необходимо на втором устройстве запустить dhcp-client на bridge-интерфейсе, логический или физический интерфейс которого будет включен в кластерный интерфейс первого устройства.

В качестве примера такой конфигурации подойдет factory-конфигурация (в factory-конфигурации для vRTT нет настроенного dhcp-client).

В процессе ZTP устройство автоматически выставит себе:

- 1) Конфигурацию;
- 2) Юнит;
- 3) Версию ПО, на котором работает Active RTT;
- 4) Лицензию, если она предварительно загружена на Active RTT.

После выполнения этих шагов кластер будет успешно запущен. Текущее состояние кластера можно узнать, выполнив команду:

RTT-1

```
RTT-1# show cluster status
```

Unit	Hostname	Role	MAC address	State	IP address
1*	RTT-1	Active	e4:5a:d4:a0:be:35	Joined	198.51.100.254
2	RTT-2	Standby	a8:f9:4b:af:35:84	Joined	198.51.100.253

После включения кластера и установления юнитов в состояние Joined дальнейшее конфигурирование устройств осуществляется настройкой Active-устройства.

Синхронизируются команды конфигурации, а также команды: **commit**, **confirm**, **rollback**, **restore**, **save**, **copy <source> system:candidate-config**.

В случае, если конфигурирование осуществляется на Standby, то внесенные изменения в конфигурацию засинхронизированы не будут. Все внесённые изменения в конфигурацию Standby будут потеряны при выполнении **commit** на Active-устройстве.

Есть возможность отключения синхронизации командой **sync config disable**.

Текущее состояние синхронизации подсистем кластера можно узнать, выполнив команду:

RTT-1

```
RTT-1# show cluster sync status
System part          Synced
-----
candidate-config     Yes
running-config       Yes
SW version           Yes
licence              Yes
licence (After reboot) Yes
date                 Yes
```

В текущей версии ПО не поддерживается синхронное шифрование паролей, вводимых не в encryption-виде.

Такие пароли будут зашифрованы каждым из участников кластера самостоятельно.

Через минуту после включения кластера синхронизируется время, на Standby установится время Active-юнита.

Синхронизация времени проверяется раз в минуту, в случае расхождения время синхронизируется.

17.2. Подключение сервисов

После успешной настройки кластера можно приступать к конфигурации различных сервисов.

17.2.1. Настройка System prompt

System prompt позволяет отобразить оперативное состояние кластера непосредственно в строке приглашения CLI устройства, что упрощает получение актуальной информации.

Варианты настройки system prompt, включая доступные параметры и синтаксис команды, приведены в документе «Справочник команд CLI».

17.2.1.1. Пример настройки

Задача:

Настроить system prompt в кластере маршрутизаторов RTT-1 и RTT-2 со следующими параметрами:

- необходимо получать информацию о статусе полной синхронизации кластера;
- необходимо получать информацию о номере юнита администрируемого устройства;
- необходимо получать информацию о роли устройства в кластере;
- необходимо получать информацию о статусе кластерного VRRP;

- необходимо получать информацию о hostname устройства.

Исходная конфигурация кластера:

RTT-1

```
cluster
  cluster-interface bridge 1
  unit 1
    mac-address cc:9d:a2:71:83:78
  exit
  unit 2
    mac-address cc:9d:a2:71:82:38
  exit
  enable
exit

hostname RTT-1 unit 1
hostname RTT-2 unit 2

security zone SYNC
exit

bridge 1
  vlan 1
  security-zone SYNC
  ip address 198.51.100.254/24 unit 1
  ip address 198.51.100.253/24 unit 2
  vrrp 1
    ip address 198.51.100.1/24
    priority 254 unit 1
    priority 253 unit 2
    group 1
    enable
  exit
  enable
exit

interface gigabitethernet 1/0/1
  mode switchport
  spanning-tree disable
exit
interface gigabitethernet 2/0/1
  mode switchport
  spanning-tree disable
exit

security zone-pair SYNC self
  rule 1
    action permit
    match protocol icmp
    enable
  exit
exit
```

Решение:

Перейдем в режим конфигурирования устройства:

RTT-1

```
RTT-1# configure
RTT-1(config)#
```

Добавим в system prompt информацию о статусе полной синхронизации кластера:

RTT-1

```
RTT-1(config)# system prompt '(Cluster: %s%)'
```

Добавим в system prompt информацию о номере юнита администрируемого устройства:

RTT-1

```
RTT-1(config)# system prompt '(Cluster: %s% | Unit: %u%)'
```

Добавим в system prompt информацию о роли устройства в кластере:

RTT-1

```
RTT-1(config)# system prompt '(Cluster: %s% | Unit: %u% | State: %r%)'
```

Добавим в system prompt информацию о статусе кластерного VRRP:

RTT-1

```
RTT-1(config)# system prompt '(Cluster: %s% | Unit: %u% | State: %r% | VRRP id
1: %v1%)'
```

Добавим в system prompt информацию о hostname устройства:

RTT-1

```
RTT-1(config)# system prompt '(Cluster: %s% | Unit: %u% | State: %r% | VRRP id
1: %v1%)| %h%'
```

Применим конфигурацию и обновим пользовательскую сессию CLI:

RTT-1

```
RTT-1# commit
Configuration has been successfully applied and saved to flash. Commit timer
started, changes will be.
RTT-1# confirm
Configuration has been confirmed. Commit timer canceled.
RTT-1# exit

RTT-1 login: admin
```

Password:

```
*****
*                               *
*           Welcome to RTT      *
*                               *
*****
```

(Cluster: Yes | Unit: 1 | State: Active | VRRP id 1: Master)|RTT-1#

Обновим пользовательскую сессию CLI на втором устройстве:

RTT-2

```
RTT-2# 2024-12-27T15:25:04+00:00 %CLUSTER-I-SYNC_CONFIG_INFO: unit 1 'RTT-1'
starts a synchronous operation 'commit'
2024-12-27T15:25:09+00:00 %CLUSTER-I-SYNC_CONFIG_INFO: 'commit' successful
performed
RTT-2# exit
```

RTT-2 login: admin
Password:

```
*****
*                               *
*           Welcome to RTT      *
*                               *
*****
```

(Cluster: Yes | Unit: 2 | State: Standby | VRRP id 1: Backup)|RTT-2#

Чтобы system prompt корректно работал, необходимо обновить пользовательскую сессию.

17.2.2. Настройка Port-channel U/N

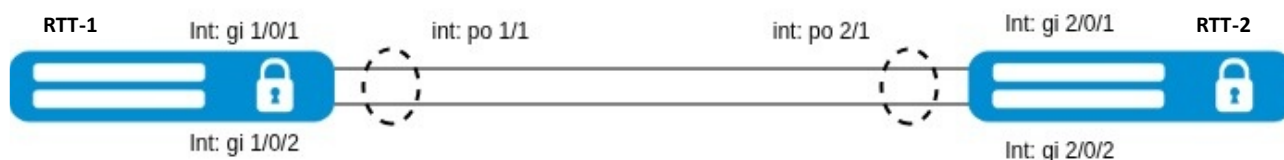
Port-channel U/N позволяет объединять каналы в группы агрегации для конкретного устройства в составе группы (unit), обеспечивая единообразие конфигурации кластера и возможность индивидуальной настройки агрегации на каждом юните.

Варианты настройки port-channel, включая доступные параметры и синтаксис команды, приведены в разделе **Типы и порядок именования интерфейсов маршрутизатора**.

17.2.2.1. Пример настройки

Задача:

Настроить port-channel U/N в кластере маршрутизаторов RTT-1 и RTT-2 для передачи Control Plane-трафика кластера через агрегированный интерфейс.



Исходная конфигурация кластера:

RTT-1

```
cluster
  cluster-interface bridge 1
  unit 1
    mac-address cc:9d:a2:71:83:78
  exit
  unit 2
    mac-address cc:9d:a2:71:82:38
  exit
  enable
exit

hostname RTT-1 unit 1
hostname RTT-2 unit 2

security zone SYNC
exit

bridge 1
  vlan 1
  security-zone SYNC
  ip address 198.51.100.254/24 unit 1
  ip address 198.51.100.253/24 unit 2
  vrrp 1
    ip address 198.51.100.1/24
    priority 254 unit 1
    priority 253 unit 2
    group 1
    enable
  exit
  enable
exit

interface gigabitethernet 1/0/1
  mode switchport
  spanning-tree disable
exit
interface gigabitethernet 2/0/1
  mode switchport
  spanning-tree disable
exit

security zone-pair SYNC self
  rule 1
    action permit
    match protocol icmp
    enable
  exit
exit
```

Создадим для каждого юнита собственный агрегированный интерфейс:

RTT-2

```
RTT-1(config)# interface port-channel 1/1
RTT-1(config-if-port-channel)# mode switchport
RTT-1(config-if-port-channel)# description Control-Plane
RTT-1(config-if-port-channel)# exit
RTT-1(config)# interface port-channel 2/1
RTT-1(config-if-port-channel)# mode switchport
RTT-1(config-if-port-channel)# description Control-Plane
RTT-1(config-if-port-channel)# exit
```

Добавим каналы в агрегированные интерфейсы, которые отвечает за Control Plane кластера и применим конфигурацию:

RTT-2

```
RTT-1(config)# interface gigabitethernet 1/0/1
RTT-1(config-if-gi)# channel-group 1 mode auto
RTT-1(config-if-gi)# exit
RTT-1(config)# interface gigabitethernet 2/0/1
RTT-1(config-if-gi)# channel-group 1 mode auto
RTT-1(config-if-gi)# exit
RTT-1(config)# interface gigabitethernet 1/0/2
RTT-1(config-if-gi)# mode switchport
RTT-1(config-if-gi)# channel-group 1 mode auto
RTT-1(config-if-gi)# spanning-tree disable
RTT-1(config-if-gi)# exit
RTT-1(config)# interface gigabitethernet 2/0/2
RTT-1(config-if-gi)# mode switchport
RTT-1(config-if-gi)# channel-group 1 mode auto
RTT-1(config-if-gi)# spanning-tree disable
RTT-1(config-if-gi)# exit
RTT-1# commit
Configuration has been successfully applied and saved to flash. Commit timer
started, changes will be.
RTT-1# confirm
Configuration has been confirmed. Commit timer canceled.
```

Проверить состояние работы port-channel можно с помощью команды:

RTT-2

```
RTT-1# show interfaces status port-channel
```

```
Unit 1* 'RTT-1'
```

Interface	Admin	Link	MTU	MAC address	Last change	Mode
State	State			(d,h:m:s)		
po1/1	Up	Up	1500	a8:f9:4b:ad:07:f9	00,00:26:29	switchport

```
Unit 2 'RTT-2'
```

Interface	Admin	Link	MTU	MAC address	Last change	Mode
State	State			(d,h:m:s)		

Юнитизированный агрегированный интерфейс для Control Plane-трафика показан как пример. Можно использовать и для передачи Data Plane-трафика.

17.2.3. Настройка MultiWAN

Технология MultiWAN позволяет организовать отказоустойчивое соединение с резервированием линков от нескольких провайдеров.

Алгоритм настройки MultiWAN описан в разделе **Настройка MultiWAN**.

17.2.3.1. Пример настройки

Задача:

Настроить MultiWAN в кластере маршрутизаторов RTT-1 и RTT-2 со следующими параметрами:

- обеспечить резервирование линков от нескольких провайдеров;
- обеспечить балансировку трафика в соотношении 70/30.

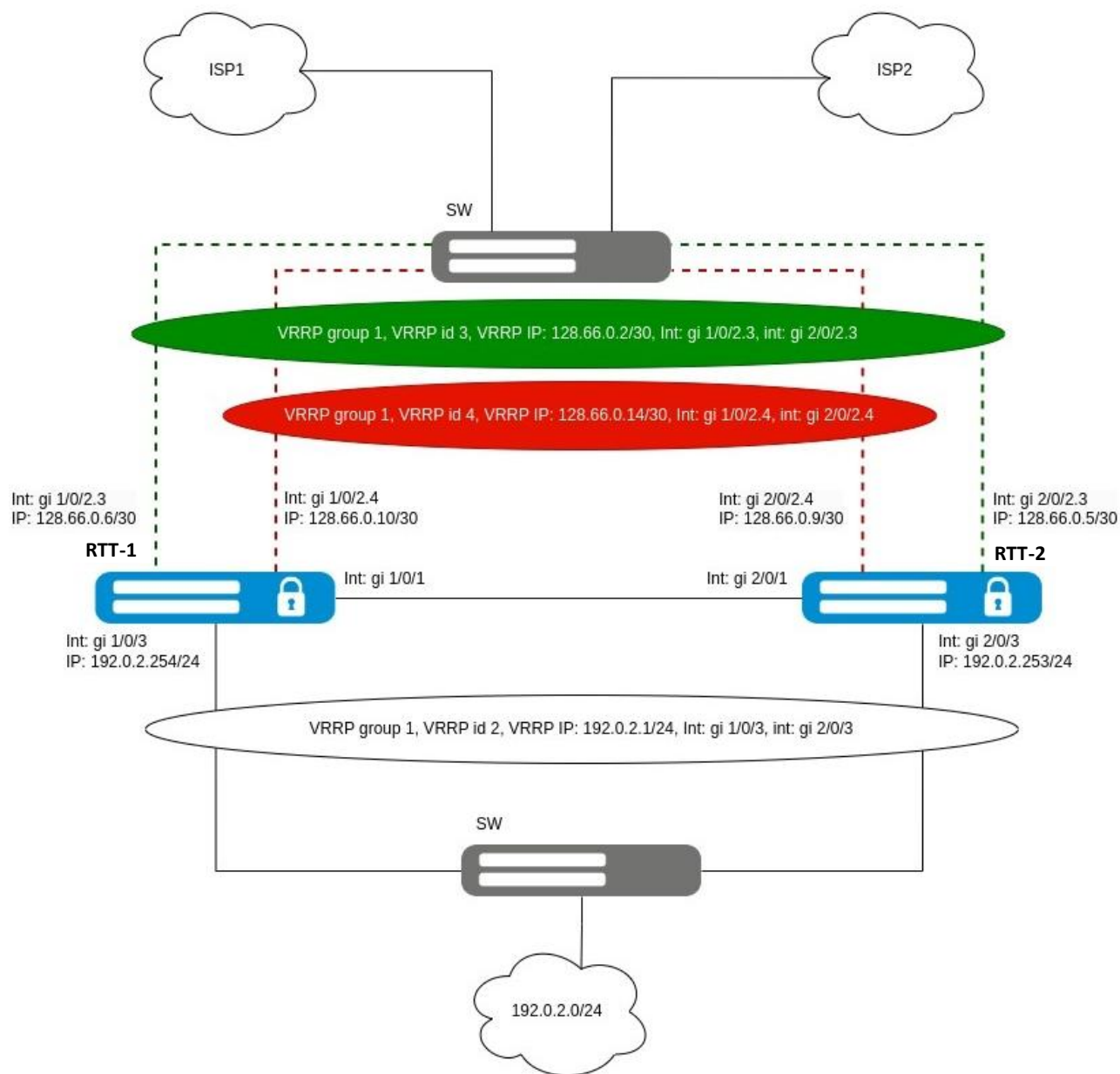


Схема реализации MultiWAN

Исходная конфигурация кластера:

RTT-1

```
cluster
cluster-interface bridge 1
unit 1
mac-address cc:9d:a2:71:83:78
exit
unit 2
mac-address cc:9d:a2:71:82:38
exit
```

```
enable
exit

hostname RTT-1 unit 1
hostname RTT-2 unit 2

security zone SYNC
exit
security zone LAN
exit
security zone WAN
exit

bridge 1
  vlan 1
  security-zone SYNC
  ip address 198.51.100.254/24 unit 1
  ip address 198.51.100.253/24 unit 2
  vrrp 1
    ip address 198.51.100.1/24
    priority 254 unit 1
    priority 253 unit 2
    group 1
    enable
  exit
  enable
exit

interface gigabitethernet 1/0/1
  mode switchport
  spanning-tree disable
exit
interface gigabitethernet 1/0/2.3
  security-zone WAN
  ip address 128.66.0.6/30
  vrrp 3
    ip address 128.66.0.2/30
    group 1
    enable
  exit
exit
interface gigabitethernet 1/0/2.4
  security-zone WAN
  ip address 128.66.0.10/30
  vrrp 4
    ip address 128.66.0.14/30
    group 1
    enable
  exit
exit
interface gigabitethernet 1/0/3
  security-zone LAN
  ip address 192.0.2.254/24
  vrrp 2
    ip address 192.0.2.1/24
    group 1
    enable
  exit
exit
```

```
interface gigabitethernet 2/0/1
 mode switchport
 spanning-tree disable
exit
interface gigabitethernet 2/0/2.3
 security-zone WAN
 ip address 128.66.0.5/30
 vrrp 3
   ip address 128.66.0.2/30
   group 1
   enable
exit
exit
interface gigabitethernet 2/0/2.4
 security-zone WAN
 ip address 128.66.0.9/30
 vrrp 4
   ip address 128.66.0.14/30
   group 1
   enable
exit
exit
interface gigabitethernet 2/0/3
 security-zone LAN
 ip address 192.0.2.253/24
 vrrp 2
   ip address 192.0.2.1/24
   group 1
   enable
exit
exit

security zone-pair SYNC self
 rule 1
   action permit
   match protocol icmp
   enable
exit
exit
security zone-pair WAN self
 rule 1
   action permit
   match protocol vrrp
   enable
exit
exit
security zone-pair LAN self
 rule 1
   action permit
   match protocol vrrp
   enable
exit
exit
```

Создадим список IP-адресов для проверки целостности соединения:

RTT-1

```
RTT-1(config)# wan load-balance target-list WAN
RTT-1(config-wan-target-list)# target 1
RTT-1(config-wan-target)# ip address 128.66.0.17
RTT-1(config-wan-target)# enable
RTT-1(config-wan-target)# exit
RTT-1(config-wan-target-list)# exit
```

Настроим WAN на интерфейсе в сторону провайдера ISP1:

RTT-1

```
RTT-1(config)# interface gigabitethernet 1/0/2.3
RTT-1(config-if-sub)# wan load-balance nexthop 128.66.0.1
RTT-1(config-if-sub)# wan load-balance target-list WAN
RTT-1(config-if-sub)# wan load-balance enable
RTT-1(config-if-sub)# exit
RTT-1(config)# interface gigabitethernet 2/0/2.3
RTT-1(config-if-sub)# wan load-balance nexthop 128.66.0.1
RTT-1(config-if-sub)# wan load-balance target-list WAN
RTT-1(config-if-sub)# wan load-balance enable
RTT-1(config-if-sub)# exit
```

Настроим WAN на интерфейсе в сторону провайдера ISP2:

RTT-1

```
RTT-1(config)# interface gigabitethernet 1/0/2.4
RTT-1(config-if-sub)# wan load-balance nexthop 128.66.0.13
RTT-1(config-if-sub)# wan load-balance target-list WAN
RTT-1(config-if-sub)# wan load-balance enable
RTT-1(config-if-sub)# exit
RTT-1(config)# interface gigabitethernet 2/0/2.4
RTT-1(config-if-sub)# wan load-balance nexthop 128.66.0.13
RTT-1(config-if-sub)# wan load-balance target-list WAN
RTT-1(config-if-sub)# wan load-balance enable
RTT-1(config-if-sub)# exit
```

Укажем статический маршрут и создадим правило для балансировки трафика:

RTT-1

```
RTT-1(config)# ip route 0.0.0.0/0 wan load-balance rule 1 10
RTT-1(config)# wan load-balance rule 1
RTT-1(config-wan-rule)# outbound interface gigabitethernet 1/0/2.3 70
RTT-1(config-wan-rule)# outbound interface gigabitethernet 1/0/2.4 30
RTT-1(config-wan-rule)# outbound interface gigabitethernet 2/0/2.3 70
RTT-1(config-wan-rule)# outbound interface gigabitethernet 2/0/2.4 30
RTT-1(config-wan-rule)# enable
RTT-1(config-wan-rule)# exit
```

Проверить состояние работы MultiWAN можно с помощью команды:

RTT-1

```
RTT-1# show wan rules
Rule 1 detailed information:
  VRF:          default
  Failover:     Disabled
  Network: 0.0.0.0/0 Metric: 10
    gil/0/2.3 Weight: 70 Nexthop: 128.66.0.1 [Active]
    gil/0/2.4 Weight: 30 Nexthop: 128.66.0.13 [Active]
```

Также состояние работы MultiWAN можно проверить с помощью команды:

RTT-1

```
RTT-1# show wan interfaces status
```

Interface	Nexthop	Status	Uptime/Downtime (d,h:m:s)
-----	-----	-----	-----
gil/0/2.3	128.66.0.1	Active	00,00:00:44
gil/0/2.4	128.66.0.13	Active	00,00:00:45

17.2.4. Настройка IPsec VPN

IPsec — это набор протоколов, обеспечивающих защиту данных, передаваемых по протоколу IP. Данный набор протоколов позволяет осуществлять подтверждение подлинности (аутентификацию), проверку целостности и шифрование IP-пакетов, а также включает в себя протоколы для защищённого обмена ключами в сети Интернет.

17.2.4.1. Пример настройки Route-based IPsec VPN

Алгоритм настройки Route-based IPsec VPN описан в разделе **Настройка IPsec VPN**.

Задача:

- Настроить IPsec туннель. Туннель необходимо поднять между адресами: кластер – 203.0.113.2 (VIP адрес), ответная сторона – 203.0.113.6;
- IKE:
 - группа Диффи-Хэллмана: 2;
 - алгоритм шифрования: AES 128 bit;
 - алгоритм аутентификации: MD5.
- IP sec:
 - алгоритм шифрования: AES 128 bit;
 - алгоритм аутентификации: MD5.

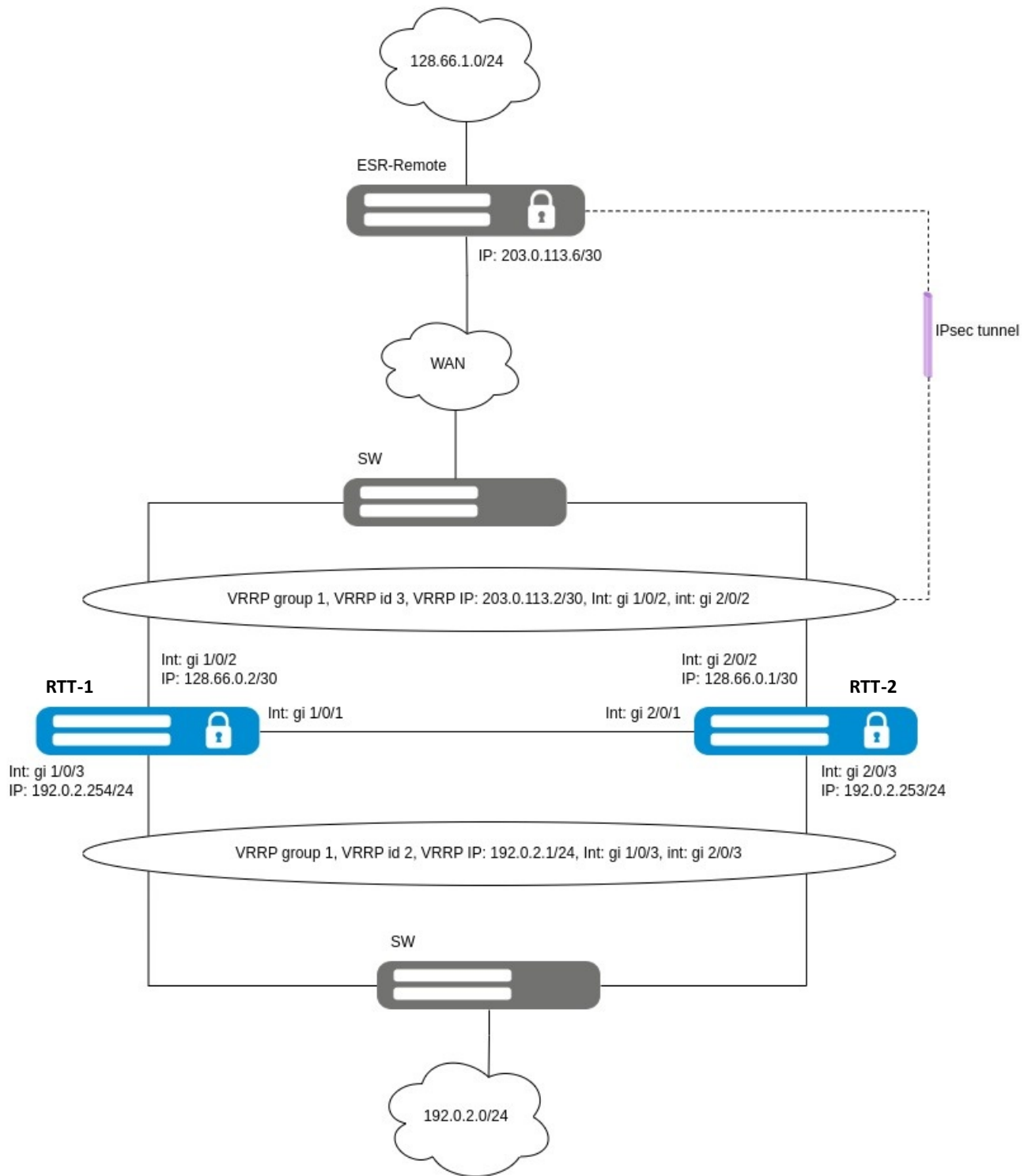


Схема реализации Route-based IPsec VPN

Исходная конфигурация кластера:

```
cluster
  cluster-interface bridge 1
  unit 1
```



```
        mac-address cc:9d:a2:71:83:78
    exit
    unit 2
        mac-address cc:9d:a2:71:82:38
    exit
    enable
exit

hostname RTT-1 unit 1
hostname RTT-2 unit 2

security zone SYNC
exit
security zone WAN
exit
security zone LAN
exit

bridge 1
    vlan 1
        security-zone SYNC
        ip address 198.51.100.254/24 unit 1
        ip address 198.51.100.253/24 unit 2
    vrrp 1
        ip address 198.51.100.1/24
        priority 254 unit 1
        priority 253 unit 2
        group 1
        enable
    exit
    enable
exit

interface gigabitethernet 1/0/1
    mode switchport
    spanning-tree disable
exit
interface gigabitethernet 1/0/2
    security-zone WAN
    ip address 128.66.0.2/30
    vrrp 3
        ip address 203.0.113.2/30
        group 1
        enable
    exit
exit
interface gigabitethernet 1/0/3
    security-zone LAN
    ip address 192.0.2.254/24
    vrrp 2
        ip address 192.0.2.1/24
        group 1
        enable
    exit
exit
interface gigabitethernet 2/0/1
    mode switchport
    spanning-tree disable
```

```
exit
interface gigabitethernet 2/0/2
  security-zone WAN
  ip address 128.66.0.1/30
  vrrp 3
    ip address 203.0.113.2/30
    group 1
    enable
  exit
exit
interface gigabitethernet 2/0/3
  security-zone LAN
  ip address 192.0.2.253/24
  vrrp 2
    ip address 192.0.2.1/24
    group 1
    enable
  exit
exit

security zone-pair SYNC self
  rule 1
    action permit
    match protocol icmp
    enable
  exit
exit
security zone-pair WAN self
  rule 1
    action permit
    match protocol vrrp
    enable
  exit
exit
security zone-pair LAN self
  rule 1
    action permit
    match protocol vrrp
    enable
  exit
exit
```

Решение:

Создадим профиль ISAKMP-портов, необходимых для работы протокола IPsec, включающий разрешение UDP-пакетов на порту 500 (а также на порту 4500 для поддержки NAT-T при необходимости):

RTT-1

```
RTT-1(config)# object-group service ISAKMP
RTT-1(config-object-group-service)# port-range 500
RTT-1(config-object-group-service)# port-range 4500
RTT-1(config-object-group-service)# exit
```

Добавим правила, разрешающее прохождение пакетов протоколов VRRP и ESP, а также UDP-пакетов с портами 500 и 4500, через IPsec-туннель:

RTT-1

```
RTT-1(config)# security zone-pair WAN self
RTT-1(config-security-zone-pair)# rule 2
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# match protocol udp
RTT-1(config-security-zone-pair-rule)# match destination-port object-group
network ISAKMP
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# rule 3
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# match protocol esp
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
```

Создадим зону безопасности IPsec и туннель VTI, через который будет перенаправляться трафик в IPsec-туннель. В качестве локального шлюза назначим VIP IP-адрес, настроенный на интерфейсах в сторону зоны WAN, а в качестве удалённого шлюза – IP-адрес соответствующего интерфейса:

RTT-1

```
RTT-1(config)# security zone IPSEC
RTT-1(config-security-zone)# exit
RTT-1(config)# tunnel vti 1
RTT-1(config-vti)# security-zone IPSEC
RTT-1(config-vti)# local address 203.0.113.2
RTT-1(config-vti)# remote address 203.0.113.6
RTT-1(config-vti)# ip address 128.66.0.6/30
RTT-1(config-vti)# enable
RTT-1(config-vti)# exit
```

Добавим правило, разрешающее прохождение трафика между зонами LAN и IPSEC:

RTT-1

```
RTT-1(config)# security zone-pair LAN IPSEC
RTT-1(config-security-zone-pair)# rule 1
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
RTT-1(config)# security zone-pair IPSEC LAN
RTT-1(config-security-zone-pair)# rule 1
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
```

Создадим профиль протокола IKE, в котором зададим следующие параметры безопасности: группу Диффи-Хэллмана 2, алгоритм шифрования AES 128 bit и алгоритм аутентификации MD5. Данные настройки обеспечивают надежную защиту IKE-соединения:

RTT-1

```
RTT-1(config)# security ike proposal ike_prop
RTT-1(config-ike-proposal)# dh-group 2
RTT-1(config-ike-proposal)# authentication algorithm md5
RTT-1(config-ike-proposal)# encryption algorithm aes128
RTT-1(config-ike-proposal)# exit
```

Создадим политику протокола IKE. В политике указывается список профилей протокола IKE, по которым могут согласовываться узлы и ключ аутентификации:

RTT-1

```
RTT-1(config)# security ike policy ike_pol
RTT-1(config-ike-policy)# pre-shared-key ascii-text password
RTT-1(config-ike-policy)# proposal ike_prop
RTT-1(config-ike-policy)# exit
```

Создадим шлюз протокола IKE с указанием VTI-туннеля, применимой политики, версии протокола и режима перенаправления трафика в туннель, а также отключим mobike:

RTT-1

```
RTT-1(config)# security ike gateway ike_gw
RTT-1(config-ike-gw)# version v2-only
RTT-1(config-ike-gw)# ike-policy ike_pol
RTT-1(config-ike-gw)# mode route-based
RTT-1(config-ike-gw)# mobike disable
RTT-1(config-ike-gw)# bind-interface vti 1
RTT-1(config-ike-gw)# exit
```

Создадим профиль параметров безопасности для IPsec-туннеля, в котором укажем алгоритм шифрования AES 128 bit и алгоритм аутентификации MD5, обеспечивая надежную защиту передаваемых данных:

RTT-1

```
RTT-1(config)# security ipsec proposal ipsec_prop
RTT-1(config-ipsec-proposal)# authentication algorithm md5
RTT-1(config-ipsec-proposal)# encryption algorithm aes128
RTT-1(config-ipsec-proposal)# exit
```

Создадим политику для IPsec-туннеля, в которой укажем перечень профилей IPsec-туннеля, используемых для согласования параметров безопасности между узлами:

RTT-1

```
RTT-1(config)# security ipsec policy ipsec_pol
```

```
RTT-1(config-ipsec-policy)# proposal ipsec_prop
RTT-1(config-ipsec-policy)# exit
```

Создадим IPsec VPN, в котором задаются следующие параметры: шлюз IKE-протокола, политика IPsec-туннеля, режим обмена ключами и способ установления соединения:

RTT-1

```
RTT-1(config)# security ipsec vpn ipsec
RTT-1(config-ipsec-vpn)# ike establish-tunnel route
RTT-1(config-ipsec-vpn)# ike gateway ike_gw
RTT-1(config-ipsec-vpn)# ike ipsec-policy ipsec_pol
RTT-1(config-ipsec-vpn)# enable
RTT-1(config-ipsec-vpn)# exit
```

Добавим статический маршрут до встречной клиентской подсети через VTI-туннель:

RTT-1

```
RTT-1(config)# ip route 128.66.1.0/24 128.66.0.5
```



Аналогичную настройку требуется выполнить на встречном устройстве.

Просмотреть состояние VTI-туннеля можно с помощью команды:

RTT-1

```
RTT-1# show tunnels status
```

Tunnel	Admin state	Link state	MTU	Local IP	Remote IP	Last change (d,h:m:s)
vti 1	Up	Up	1500	203.0.113.2	203.0.113.6	00,00:05:59

Посмотреть состояние IPsec-туннеля можно с помощью команды:

RTT-1

```
RTT-1# show security ipsec vpn status
```



Аналогичную настройку требуется выполнить на встречном устройстве.

17.2.4.2. Пример настройки Policy-based IPsec VPN

Алгоритм настройки Policy-based IPsec VPN описан в разделе **Настройка IPsec VPN**.

Задача:

- Настроить IPsec туннель. Туннель необходимо поднять между адресами: кластер – 203.0.113.2 (VIP адрес), ответная сторона – 203.0.113.6. Туннель необходим для организации доступа между клиентскими подсетями 192.0.2.0/24 и 128.66.1.0/24;
- IKE:
 - группа Диффи-Хэллмана: 2;
 - алгоритм шифрования: AES 128 bit;
 - алгоритм аутентификации: MD5.
- IP sec:
 - алгоритм шифрования: AES 128 bit;
 - алгоритм аутентификации: MD5.

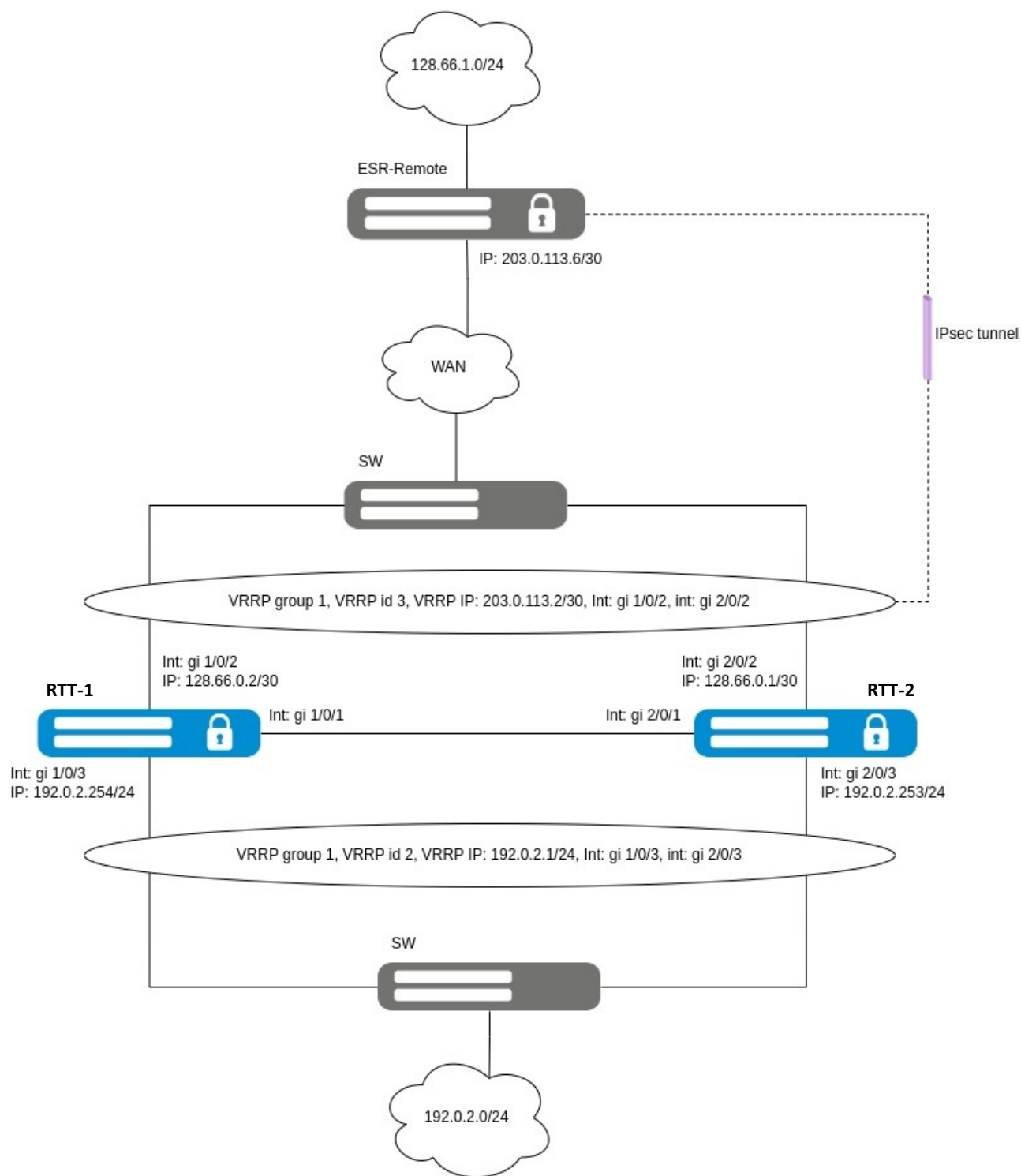


Схема реализации Policy-based IPsec VPN

Исходная конфигурация кластера:

```
cluster
cluster-interface bridge 1
```

```
unit 1
  mac-address cc:9d:a2:71:83:78
exit
unit 2
  mac-address cc:9d:a2:71:82:38
exit
enable
exit

hostname RTT-1 unit 1
hostname RTT-2 unit 2

security zone SYNC
exit
security zone WAN
exit
security zone LAN
exit

bridge 1
  vlan 1
  security-zone SYNC
  ip address 198.51.100.254/24 unit 1
  ip address 198.51.100.253/24 unit 2
  vrrp 1
    ip address 198.51.100.1/24
    priority 254 unit 1
    priority 253 unit 2
    group 1
    enable
  exit
  enable
exit

interface gigabitethernet 1/0/1
  mode switchport
  spanning-tree disable
exit
interface gigabitethernet 1/0/2
  security-zone WAN
  ip address 128.66.0.2/30
  vrrp 3
    ip address 203.0.113.2/30
    group 1
    enable
  exit
exit
interface gigabitethernet 1/0/3
  security-zone LAN
  ip address 192.0.2.254/24
  vrrp 2
    ip address 192.0.2.1/24
    group 1
    enable
  exit
exit
interface gigabitethernet 2/0/1
  mode switchport
  spanning-tree disable
```



```
exit
interface gigabitethernet 2/0/2
  security-zone WAN
  ip address 128.66.0.1/30
  vrrp 3
    ip address 203.0.113.2/30
    group 1
    enable
  exit
exit
interface gigabitethernet 2/0/3
  security-zone LAN
  ip address 192.0.2.253/24
  vrrp 2
    ip address 192.0.2.1/24
    group 1
    enable
  exit
exit

security zone-pair SYNC self
  rule 1
    action permit
    match protocol icmp
    enable
  exit
exit
security zone-pair WAN self
  rule 1
    action permit
    match protocol vrrp
    enable
  exit
exit
security zone-pair LAN self
  rule 1
    action permit
    match protocol vrrp
    enable
  exit
exit
```

Решение:

Создадим профиль ISAKMP-портов, необходимых для работы протокола IPsec, включающий разрешение UDP-пакетов на порту 500 (а также на порту 4500 для поддержки NAT-T при необходимости):

RTT-1

```
RTT-1(config)# object-group service ISAKMP
RTT-1(config-object-group-service)# port-range 500
RTT-1(config-object-group-service)# port-range 4500
RTT-1(config-object-group-service)# exit
```

Добавим правила, разрешающее прохождение пакетов протоколов VRRP и ESP, а также UDP-пакетов с портами 500 и 4500, через IPsec-туннель:

RTT-1

```
RTT-1(config)# security zone-pair WAN self
RTT-1(config-security-zone-pair)# rule 2
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# match protocol udp
RTT-1(config-security-zone-pair-rule)# match destination-port object-group
ISAKMP
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# rule 3
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# match protocol esp
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
```

Добавим правило, разрешающее прохождение трафика между зонами LAN и WAN для клиентских подсетей:

RTT-1

```
RTT-1(config)# object-group network LAN
RTT-1(config-object-group-network)# ip prefix 192.0.2.0/24
RTT-1(config-object-group-network)# exit
RTT-1(config)# object-group network IPSEC
RTT-1(config-object-group-network)# ip prefix 128.66.1.0/24
RTT-1(config-object-group-network)# exit
RTT-1(config)# security zone-pair LAN WAN
RTT-1(config-security-zone-pair)# rule 1
RTT-1(config-security-zone-pair-rule)# match source-address object-group network
LAN
RTT-1(config-security-zone-pair-rule)# match destination-address object-group
network IPSEC
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
RTT-1(config)# security zone-pair WAN LAN
RTT-1(config-security-zone-pair)# rule 1
RTT-1(config-security-zone-pair-rule)# match source-address object-group network
IPSEC
RTT-1(config-security-zone-pair-rule)# match destination-address object-group
network LAN
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
```

Создадим профиль протокола IKE, в котором зададим следующие параметры безопасности: группу Диффи-Хэллмана 2, алгоритм шифрования AES 128 bit и алгоритм аутентификации MD5. Данные настройки обеспечивают надежную защиту IKE-соединения:

RTT-1

```
RTT-1(config)# security ike proposal ike_prop
RTT-1(config-ike-proposal)# dh-group 2
RTT-1(config-ike-proposal)# authentication algorithm md5
RTT-1(config-ike-proposal)# encryption algorithm aes128
RTT-1(config-ike-proposal)# exit
```

Создадим политику протокола IKE. В политике указывается список профилей протокола IKE, по которым могут согласовываться узлы и ключ аутентификации:

RTT-1

```
RTT-1(config)# security ike policy ike_pol
RTT-1(config-ike-policy)# pre-shared-key ascii-text password
RTT-1(config-ike-policy)# proposal ike_prop
RTT-1(config-ike-policy)# exit
```

Создадим шлюз протокола IKE, определив применимую IKE-политику, локальные и удалённые параметры, а также режим перенаправления трафика в туннель. В качестве локального шлюза назначим VIP IP-адрес, настроенный на интерфейсах в сторону зоны WAN, с локальной подсетью 192.0.2.0/24, а удалённым – IP-адрес 203.0.113.1 с удаленной подсетью 128.66.1.0/24. Режим перенаправления трафика установлен как policy-based:

RTT-1

```
RTT-1(config)# security ike gateway ike_gw
RTT-1(config-ike-gw)# ike-policy ike_pol
RTT-1(config-ike-gw)# local address 203.0.113.2
RTT-1(config-ike-gw)# local network 192.0.2.0/24
RTT-1(config-ike-gw)# remote address 203.0.113.6
RTT-1(config-ike-gw)# remote network 128.66.1.0/24
RTT-1(config-ike-gw)# mode policy-based
RTT-1(config-ike-gw)# exit
```

Создадим профиль параметров безопасности для IPsec-туннеля, в котором укажем алгоритм шифрования AES 128 bit и алгоритм аутентификации MD5, обеспечивая надежную защиту передаваемых данных:

RTT-1

```
RTT-1(config)# security ipsec proposal ipsec_prop
RTT-1(config-ipsec-proposal)# authentication algorithm md5
RTT-1(config-ipsec-proposal)# encryption algorithm aes128
RTT-1(config-ipsec-proposal)# exit
```

Создадим политику для IPsec-туннеля, в которой укажем перечень профилей IPsec-туннеля, используемых для согласования параметров безопасности между узлами:

RTT-1

```
RTT-1(config)# security ipsec policy ipsec_pol
```

```
RTT-1(config-ipsec-policy)# proposal ipsec_prop
RTT-1(config-ipsec-policy)# exit
```

Создадим IPsec VPN, в котором задаются следующие параметры: шлюз IKE-протокола, политика IPsec-туннеля, режим обмена ключами и способ установления соединения:

RTT-1

```
RTT-1(config)# security ipsec vpn ipsec
RTT-1(config-ipsec-vpn)# ike establish-tunnel route
RTT-1(config-ipsec-vpn)# ike gateway ike_gw
RTT-1(config-ipsec-vpn)# ike ipsec-policy ipsec_pol
RTT-1(config-ipsec-vpn)# enable
RTT-1(config-ipsec-vpn)# exit
```

Добавим статический маршрут до встречной клиентской подсети через IPsec-туннель:

RTT-1

```
RTT-1(config)# ip route 128.66.1.0/24 203.0.113.1
```



Аналогичную настройку требуется выполнить на устройстве, находящемся на другой стороне туннеля.

Посмотреть состояние IPsec-туннеля можно с помощью команды:

RTT-1

```
RTT-1# show security ipsec vpn status
```

Name	Local host	Remote host	Initiator spi	Responder spi	State
ipsec	203.0.113.2	203.0.113.6	0x201602ebcafb809b	0x4556a21a7012d2c0	Established

17.2.5. Настройка firewall/NAT failover

Firewall failover необходим для резервирования сессий firewall.

Алгоритм настройки firewall/NAT failover описан в разделе **Настройка Firewall/NAT failover**.

17.2.5.1. Пример настройки firewall failover

Задача:

Настроить firewall failover в кластере маршрутизаторов RTT-1 и RTT-2 со следующими параметрами:

- режим резервирования сессий unicast;
- номер UDP-порта службы резервирования 9999;
- клиентская подсеть: 192.0.2.0/24.

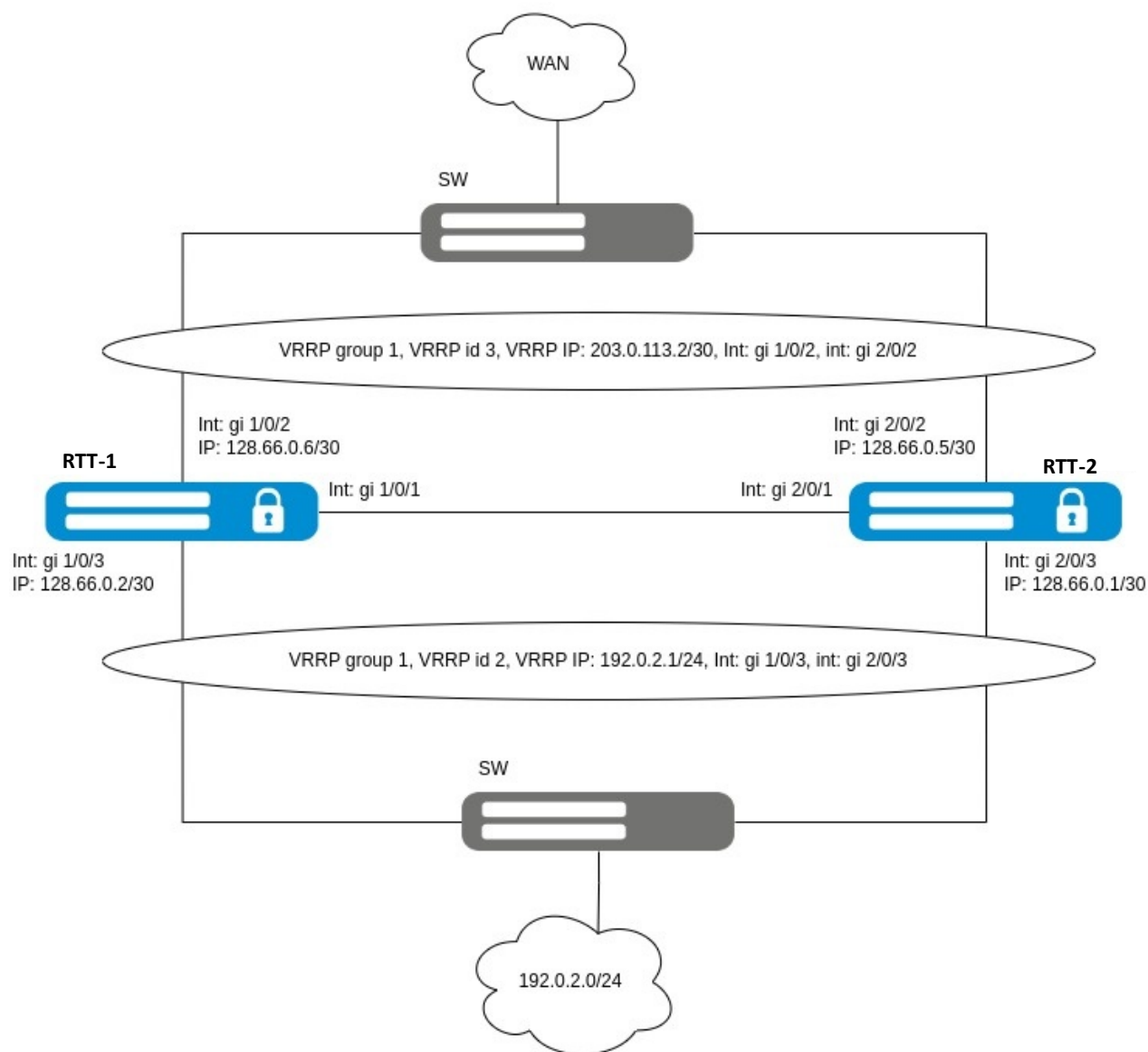


Схема реализации Firewall failover

Исходная конфигурация кластера:

RTT-1

```
cluster
cluster-interface bridge 1
unit 1
mac-address cc:9d:a2:71:83:78
exit
unit 2
mac-address cc:9d:a2:71:82:38
exit
enable
```

exit

hostname RTT-1 unit 1
hostname RTT-2 unit 2

security zone SYNC
exit
security zone WAN
exit
security zone LAN
exit

bridge 1
 vlan 1
 security-zone SYNC
 ip address 198.51.100.254/24 unit 1
 ip address 198.51.100.253/24 unit 2
 vrrp 1
 ip address 198.51.100.1/24
 priority 254 unit 1
 priority 253 unit 2
 group 1
 enable
 exit
 enable
exit

interface gigabitethernet 1/0/1
 mode switchport
 spanning-tree disable
exit

interface gigabitethernet 1/0/2
 security-zone WAN
 ip address 128.66.0.6/30
 vrrp 3
 ip address 203.0.113.2/30
 group 1
 enable
 exit
exit

interface gigabitethernet 1/0/3
 security-zone LAN
 ip address 128.66.0.2/30
 vrrp 2
 ip address 192.0.2.1/24
 group 1
 enable
 exit
exit

interface gigabitethernet 2/0/1
 mode switchport
 spanning-tree disable
exit

interface gigabitethernet 2/0/2
 security-zone WAN
 ip address 128.66.0.5/30
 vrrp 3
 ip address 203.0.113.2/30
 group 1

```
        enable
    exit
exit
interface gigabitethernet 2/0/3
    security-zone LAN
    ip address 128.66.0.1/30
    vrrp 2
        ip address 192.0.2.1/24
        group 1
        enable
    exit
exit
security zone-pair SYNC self
    rule 1
        action permit
        match protocol icmp
        enable
    exit
exit
security zone-pair LAN self
    rule 1
        action permit
        match protocol vrrp
        enable
    exit
exit
security zone-pair WAN self
    rule 1
        action permit
        match protocol vrrp
        enable
    exit
exit
security zone-pair LAN WAN
    rule 1
        action permit
        enable
    exit
exit
ip route 0.0.0.0/0 203.0.113.1
```

Решение:

Сконфигурируем object-group для настройки failover-сервисов:

RTT-1

```
RTT-1(config)# object-group network SYNC_SRC
RTT-1(config-object-group-network)# ip address-range 198.51.100.254 unit 1
RTT-1(config-object-group-network)# ip address-range 198.51.100.253 unit 2
RTT-1(config-object-group-network)# exit
RTT-1(config)# object-group network SYNC_DST
RTT-1(config-object-group-network)# ip address-range 198.51.100.253 unit 1
RTT-1(config-object-group-network)# ip address-range 198.51.100.254 unit 2
```

```
RTT-1(config-object-group-network)# exit
```

Перейдем к настройке общих параметров для failover-сервисов, а именно к выбору: IP-адреса, с которого будут отправляться сообщения для синхронизации, IP-адреса получателя сообщений для синхронизации и VRRP-группу, на основе которой определяется состояние (основной/резервный) маршрутизатора при работе failover-сервисов:

RTT-1

```
RTT-1(config)# ip failover
RTT-1(config-failover)# local-address object-group SYNC_SRC
RTT-1(config-failover)# remote-address object-group SYNC_DST
RTT-1(config-failover)# vrrp-group 1
RTT-1(config-failover)# exit
```



При включенном кластере использование object-group в настройке failover-сервисов для local-/remote-адресов обязательно.

RTT-1

```
RTT-1(config)# object-group service FAILOVER
RTT-1(config-object-group-service)# port-range 9999
RTT-1(config-object-group-service)# exit
```

Создадим разрешающее правило для зоны безопасности SYNC, разрешив прохождение необходимого трафика для работы firewall failover:

RTT-1

```
RTT-1(config)# security zone-pair SYNC self
RTT-1(config-security-zone-pair)# rule 4
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# match protocol udp
RTT-1(config-security-zone-pair-rule)# match destination-port object-group
FAILOVER
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
```

Выполним настройку firewall failover. Настроим режим резервирования сессий unicast, номер UDP-порта службы резервирования сессий firewall и включим firewall failover:

RTT-1

```
RTT-1(config)# ip firewall failover
RTT-1(config-firewall-failover)# sync-type unicast
RTT-1(config-firewall-failover)# port 9999
RTT-1(config-firewall-failover)# enable
RTT-1(config-firewall-failover)# exit
```

После успешного запуска firewall failover можно посмотреть информацию о сервисе с помощью команды:

RTT-1

```
RTT-1# show ip firewall failover
Communication interface:      bridge 1
Status:                      Running
Bytes sent:                  3420
Bytes received:              3320
Packets sent:                209
Packets received:            209
Send errors:                 0
Receive errors:              0
Resend queue:
  Active entries:            1
  Errors:
    No space left:          0
Hold queue:
  Active entries:            0
  Errors:
    No space left:          0
```

Также возможно узнать текущее состояние firewall failover сервиса, выполнив команду:

RTT-1

```
RTT-1# show high-availability state
AP Tunnels:
  State:                     Disabled
  Last state change:         --
DHCP option 82 table:
  State:                     Disabled
  Last state change:         --
DHCP server:
  State:                     Disabled
  Last state change:         --
crypto-sync:
  State:                     Disabled
Firewall sessions and NAT translations:
VRF:
  Tracking VRRP Group        1
  Tracking VRRP Group state: Master
  State:                     Successful synchronization
  Fault Reason:              --
  Last synchronization:      2025-02-12 07:05:47
```

Сгенерируем одну клиентскую сессию из LAN в WAN.

Посмотреть firewall-сессии, которые синхронизируются между устройствами, можно командами:

RTT-1

```
RTT-1# show ip firewall sessions failover internal
```

Codes: E - expected, U - unreplied,
A - assured, C - confirmed

Prot	Aging	Inside source	Inside destination	Outside source	Outside destination	Pkts	Bytes	Status
tcp	0	192.0.2.10:44812	128.66.1.1:22	128.66.1.1:22	192.0.2.10:44812	--	--	AC

RTT-2

RTT-1# show ip firewall sessions failover external

Codes: E - expected, U - unreplied,
A - assured, C - confirmed

Prot	Aging	Inside source	Inside destination	Outside source	Outside destination	Pkts	Bytes	Status
tcp	0	192.0.2.10:44812	128.66.1.1:22	128.66.1.1:22	192.0.2.10:44812	--	--	AC

Посмотреть счетчики для кэшей firewall failover можно командой:

RTT-1

RTT-1# show ip firewall failover cache

Internal sessions cache counters:

Active entries:	1
Added:	5
Deleted:	4
Updated:	4
Failed adding:	0
No memory left:	0
No space left:	0
Failed deleting:	0
No entry found:	0
Failed updating:	0
No entry found:	0

External sessions cache counters:

Active entries:	0
Added:	0
Deleted:	0
Updated:	0
Installed to Kernel:	0
Failed adding:	0
No memory left:	0
No space left:	0
Failed deleting:	0
No entry found:	0
Failed updating:	0
No entry found:	0
Failed installing to Kernel:	0

17.2.5.2. *Пример настройки нескольких экземпляров firewall failover, каждый – в своём VRF*

Задача:

Настроить несколько экземпляров firewall failover в кластере маршрутизаторов RTT-1 и RTT-2, каждый в своем VRF, со следующими параметрами:

- режим резервирования сессий unicast;
- номер UDP-порта службы резервирования 9999;
- настроить приоритеты у разных firewall failover так, чтобы один из юнитов кластера был Master в одном VRF, а в другом был Backup;
- клиентская подсеть через первый VRF: 192.0.2.0/24;
- клиентская подсеть через второй VRF: 128.66.0.0/24.

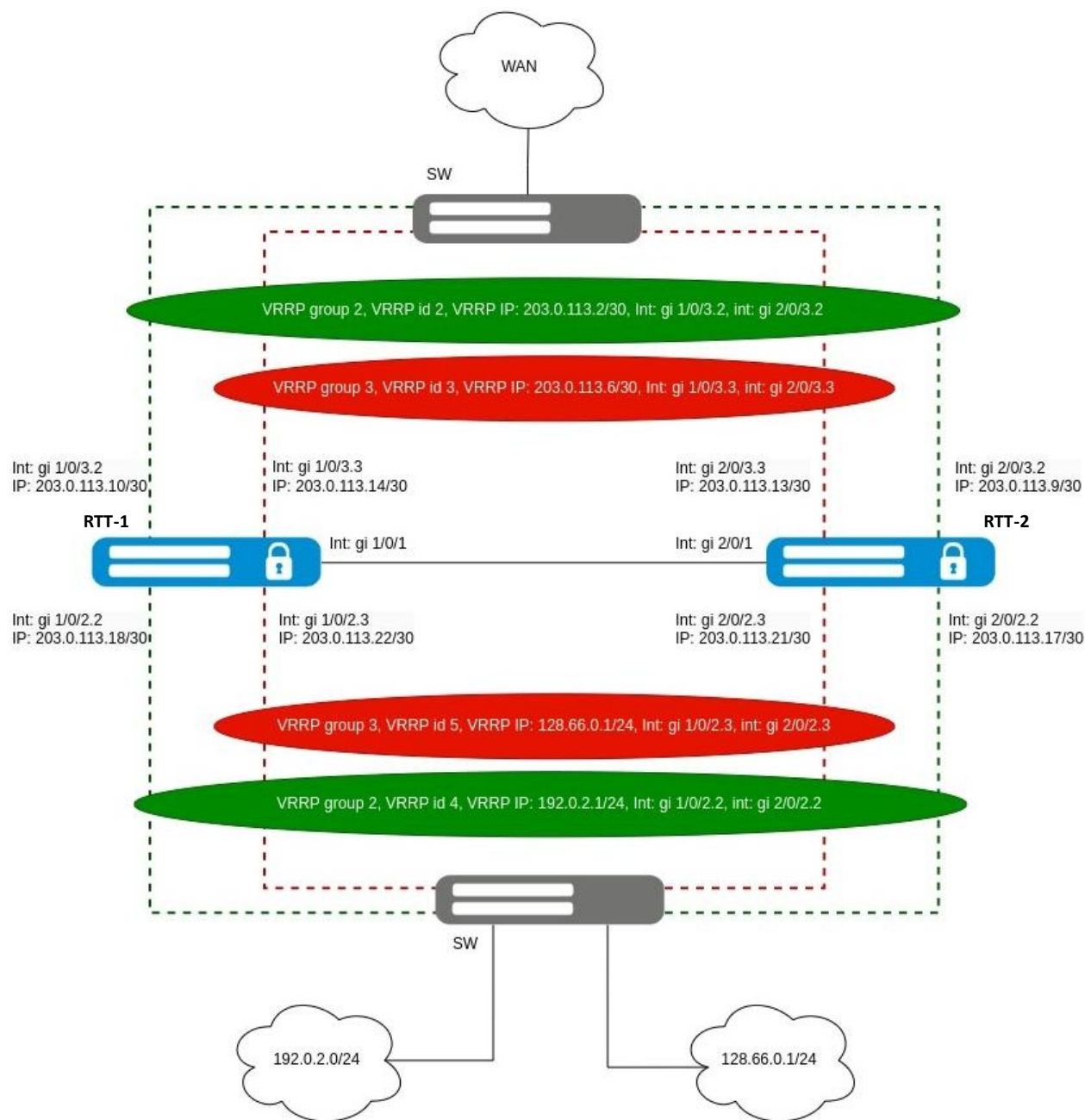


Схема реализации firewall failover в нескольких VRF

Исходная конфигурация кластера:

```
cluster
  cluster-interface bridge 1
  unit 1
    mac-address a2:00:00:10:c0:00
  exit
  unit 2
    mac-address a2:00:00:10:d0:00
  exit
  enable
exit

hostname RTT-1 unit 1
hostname RTT-2 unit 2

ip vrf PAIR_ONE
exit
ip vrf PAIR_TWO
exit

security zone SYNC
exit
security zone LAN_ONE
  ip vrf forwarding PAIR_ONE
exit
security zone LAN_TWO
  ip vrf forwarding PAIR_TWO
exit
security zone WAN_ONE
  ip vrf forwarding PAIR_ONE
exit
security zone WAN_TWO
  ip vrf forwarding PAIR_TWO
exit

bridge 1
  vlan 1
  security-zone SYNC
  ip address 198.51.100.254/24 unit 1
  ip address 198.51.100.253/24 unit 2
  vrrp 1
    ip address 198.51.100.1/24
    priority 254 unit 1
    priority 253 unit 2
    group 1
    enable
  exit
  enable
exit

interface gigabitethernet 1/0/1
  mode switchport
exit
interface gigabitethernet 1/0/2.2
  ip vrf forwarding PAIR_ONE
  security-zone LAN_ONE
  ip address 203.0.113.18/30
```

```
vrrp 4
  ip address 192.0.2.1/24
  priority 120
  group 2
  enable
exit
exit
interface gigabitethernet 1/0/2.3
  ip vrf forwarding PAIR_TWO
  security-zone LAN_TWO
  ip address 203.0.113.22/30
  vrrp 5
    ip address 128.66.0.1/24
    priority 110
    group 3
    enable
  exit
exit
interface gigabitethernet 1/0/3.2
  ip vrf forwarding PAIR_ONE
  security-zone WAN_ONE
  ip address 203.0.113.10/30
  vrrp 2
    ip address 203.0.113.2/30
    group 2
    enable
  exit
exit
interface gigabitethernet 1/0/3.3
  ip vrf forwarding PAIR_TWO
  security-zone WAN_TWO
  ip address 203.0.113.14/30
  vrrp 3
    ip address 203.0.113.6/30
    group 3
    enable
  exit
exit
interface gigabitethernet 2/0/1
  mode switchport
exit
interface gigabitethernet 2/0/2.2
  ip vrf forwarding PAIR_ONE
  security-zone LAN_ONE
  ip address 203.0.113.17/30
  vrrp 4
    ip address 192.0.2.1/24
    priority 110
    group 2
    enable
  exit
exit
interface gigabitethernet 2/0/2.3
  ip vrf forwarding PAIR_TWO
  security-zone LAN_TWO
  ip address 203.0.113.21/30
  vrrp 5
    ip address 128.66.0.1/24
```

```
    priority 120
    group 3
    enable
  exit
exit
interface gigabitethernet 2/0/3.2
  ip vrf forwarding PAIR_ONE
  security-zone WAN_ONE
  ip address 203.0.113.9/30
  vrrp 2
    ip address 203.0.113.2/30
    group 2
    enable
  exit
exit
interface gigabitethernet 2/0/3.3
  ip vrf forwarding PAIR_TWO
  security-zone WAN_TWO
  ip address 203.0.113.13/30
  vrrp 3
    ip address 203.0.113.6/30
    group 3
    enable
  exit
exit
security zone-pair SYNC self
  rule 1
    action permit
    match protocol icmp
    enable
  exit
exit
security zone-pair LAN_ONE self
  rule 1
    action permit
    match protocol vrrp
    enable
  exit
exit
security zone-pair LAN_TWO self
  rule 1
    action permit
    match protocol vrrp
    enable
  exit
exit
security zone-pair WAN_ONE self
  rule 1
    action permit
    match protocol vrrp
    enable
  exit
exit
security zone-pair WAN_TWO self
  rule 1
    action permit
    match protocol vrrp
    enable
```

```
exit
exit
security zone-pair LAN_ONE WAN_ONE
    rule 1
        action permit
        enable
    exit
exit
security zone-pair LAN_TWO WAN_TWO
    rule 1
        action permit
        enable
    exit
exit
```

Решение:

Сконфигурируем object-group для настройки failover-сервисов:

RTT-1

```
RTT-1(config)# object-group network DST_PAIR_ONE
RTT-1(config-object-group-network)# ip address-range 203.0.113.17 unit 1
RTT-1(config-object-group-network)# ip address-range 203.0.113.18 unit 2
RTT-1(config-object-group-network)# exit
RTT-1(config)# object-group network DST_PAIR_TWO
RTT-1(config-object-group-network)# ip address-range 203.0.113.21 unit 1
RTT-1(config-object-group-network)# ip address-range 203.0.113.22 unit 2
RTT-1(config-object-group-network)# exit
RTT-1(config)# object-group network SRC_PAIR_ONE
RTT-1(config-object-group-network)# ip address-range 203.0.113.18 unit 1
RTT-1(config-object-group-network)# ip address-range 203.0.113.17 unit 2
RTT-1(config-object-group-network)# exit
RTT-1(config)# object-group network SRC_PAIR_TWO
RTT-1(config-object-group-network)# ip address-range 203.0.113.22 unit 1
RTT-1(config-object-group-network)# ip address-range 203.0.113.21 unit 2
RTT-1(config-object-group-network)# exit
```

Перейдем к настройке ip failover для каждого VRF, настроим там local-address/remote-address и укажем привязки к соответствующим VRRP-group, на основе которых будет определяться, какой из маршрутизаторов будет синхронизировать сессии:

RTT-1

```
RTT-1(config)# ip failover vrf PAIR_ONE
RTT-1(config-failover)# local-address object-group SRC_PAIR_ONE
RTT-1(config-failover)# remote-address object-group DST_PAIR_ONE
RTT-1(config-failover)# vrrp-group 2
RTT-1(config-failover)# exit
RTT-1(config)# ip failover vrf PAIR_TWO
RTT-1(config-failover)# local-address object-group SRC_PAIR_TWO
RTT-1(config-failover)# remote-address object-group DST_PAIR_TWO
RTT-1(config-failover)# vrrp-group 3
RTT-1(config-failover)# exit
```

Перейдем к настройке firewall failover, каждый в своем VRF. Для каждого экземпляра необходимо указать режим синхронизирования unicast, настроить port 9999, а также включить его:

RTT-1

```
RTT-1(config)# ip firewall failover vrf PAIR_ONE
RTT-1(config-firewall-failover)# sync-type unicast
RTT-1(config-firewall-failover)# port 9999
RTT-1(config-firewall-failover)# enable
RTT-1(config-firewall-failover)# exit
RTT-1(config)# ip firewall failover vrf PAIR_TWO
RTT-1(config-firewall-failover)# sync-type unicast
RTT-1(config-firewall-failover)# port 9999
RTT-1(config-firewall-failover)# enable
RTT-1(config-firewall-failover)# exit
```

Разрешим в настройках firewall работу firewall failover в соответствующих зонах:

RTT-1

```
RTT-1(config)# object-group service FAILOVER
RTT-1(config-object-group-service)# port-range 9999
RTT-1(config-object-group-service)# exit
RTT-1(config)# security zone-pair LAN_ONE self
RTT-1(config-security-zone-pair)# rule 3
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# match protocol udp
RTT-1(config-security-zone-pair-rule)# match destination-port object-group
FAILOVER
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
RTT-1(config)# security zone-pair LAN_TWO self
RTT-1(config-security-zone-pair)# rule 3
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# match protocol udp
RTT-1(config-security-zone-pair-rule)# match destination-port object-group
FAILOVER
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
```

Просмотреть VRRP-статусы в разных VRF можно, используя команду **show vrrp**. Убедимся, что в одном VRF устройство находится в статусе Master, а в другом VRF – в статусе Backup:

RTT-1

```
RTT-1# show vrrp vrf PAIR_ONE
```

```
Unit 1* 'RTT-1'
-----
Virtual router   Virtual IP           Priority   Preemption   State   Inherit   Sync group
ID
-----
2                203.0.113.2/30       100       Enabled      Master  --        2
4                192.0.2.1/24         120       Enabled      Master  --        2
```


Unit 2 'RTT-2'

Virtual router ID	Virtual IP	Priority	Preemption	State	Inherit	Sync group
2	203.0.113.2/30	100	Enabled	Backup	--	2
4	192.0.2.1/24	110	Enabled	Backup	--	2

RTT-1# show vrrp vrf PAIR_TWO

Unit 1* 'RTT-1'

Virtual router ID	Virtual IP	Priority	Preemption	State	Inherit	Sync group
3	203.0.113.6/30	100	Enabled	Backup	--	3
5	128.66.0.1/24	110	Enabled	Backup	--	3

Unit 2 'RTT-2'

Virtual router ID	Virtual IP	Priority	Preemption	State	Inherit	Sync group
3	203.0.113.6/30	100	Enabled	Master	--	3
5	128.66.0.1/24	120	Enabled	Master	--	3

Посмотреть информацию о сервисе firewall failover в каждом VRF можно с помощью следующей команды:

RTT-1

RTT-1# show ip firewall failover vrf PAIR_ONE

```

Communication interface:      gigabitethernet 1/0/2.2
Status:                        Running
Bytes sent:                    7420
Bytes received:                7200
Packets sent:                  465
Packets received:              460
Send errors:                   0
Receive errors:                0
Resend queue:
    Active entries:            1
    Errors:
        No space left:         0
Hold queue:
    Active entries:            0
    Errors:
        No space left:         0

```

RTT-1# show ip firewall failover vrf PAIR_TWO

```

Communication interface:      gigabitethernet 1/0/2.3
Status:                        Running
Bytes sent:                    7320
Bytes received:                7380
Packets sent:                  468
Packets received:              464
Send errors:                   0

```

```

Receive errors:                                0
Resend queue:
  Active entries:                              1
  Errors:
    No space left:                             0
Hold queue:
  Active entries:                              0
  Errors:
    No space left:                             0

```

Также возможно узнать текущее состояние firewall failover сервисов во всех VRF, выполнив команду:

RTT-1

```

RTT-1# show high-availability state
DHCP server:
  State:                                Disabled
  Last state change:                    --
crypto-sync:
  State:                                Disabled
Firewall sessions and NAT translations:
VRF:                                    PAIR_ONE
  State:                                Successful synchronization
  Fault Reason:                         --
  Last synchronization:                 2025-02-18 08:51:34
VRF:                                    PAIR_TWO
  State:                                Successful synchronization
  Fault Reason:                         --
  Last synchronization:                 2025-02-18 08:51:34

```

Сгенерируем по одной клиентской сессии из каждого LAN-пула.

Посмотреть вывод текущих сессий на устройстве можно с помощью команды **show ip firewall sessions**. Убедимся, что в выводе есть сессия только для того VRF, в котором устройство является в статусе Master:

RTT-1

```

RTT-1# show ip firewall sessions vrf PAIR_ONE protocol tcp
Codes: E - expected, U - unreplied,
       A - assured, C - confirmed

```

Prot	Aging	Inside source	Inside destination	Outside source	Outside destination	Pkts	Bytes	Status
tcp	110	192.0.2.10:47406	203.0.113.1:22	192.0.2.10:47406	203.0.113.1:22	--	--	AC

RTT-2

```

RTT-2# show ip firewall sessions vrf PAIR_ONE protocol tcp
RTT-2# show ip firewall sessions vrf PAIR_TWO protocol tcp
Codes: E - expected, U - unreplied,
       A - assured, C - confirmed

```

Prot	Aging	Inside source	Inside destination	Outside source	Outside destination	Pkts	Bytes	Status
tcp	113	128.66.0.10:59108	203.0.113.5:22	128.66.0.10:59108	203.0.113.5:22	--	--	AC

Посмотреть вывод активных синхронизируемых сессий, используемых для работы firewall failover, на устройстве можно с помощью команды **show ip firewall session failover external/internal**. Убедимся, что для одного из VRF сессия находится в internal cash, а для второго VRF сессия находится в external cash:

RTT-1

```
RTT-1# show ip firewall sessions failover external vrf PAIR_ONE
```

```
RTT-1# show ip firewall sessions failover internal vrf PAIR_ONE
```

Codes: E - expected, U - unreplied,
A - assured, C - confirmed

Prot	Aging	Inside source	Inside destination	Outside source	Outside destination	Pkts	Bytes	Status
tcp	0	192.0.2.10:47406	203.0.113.1:22	203.0.113.1:22	192.0.2.10:47406	--	--	AC

```
RTT-1# show ip firewall sessions failover internal vrf PAIR_TWO
```

RTT-2

```
RTT-2# show ip firewall sessions failover external vrf PAIR_ONE
```

Codes: E - expected, U - unreplied,
A - assured, C - confirmed

Prot	Aging	Inside source	Inside destination	Outside source	Outside destination	Pkts	Bytes	Status
tcp	0	192.0.2.10:47406	203.0.113.1:22	203.0.113.1:22	192.0.2.10:47406	--	--	AC

```
RTT-2# show ip firewall sessions failover internal vrf PAIR_ONE
```

```
RTT-2# show ip firewall sessions failover external vrf PAIR_TWO
```

```
RTT-2# show ip firewall sessions failover internal vrf PAIR_TWO
```

Codes: E - expected, U - unreplied,
A - assured, C - confirmed

Prot	Aging	Inside source	Inside destination	Outside source	Outside destination	Pkts	Bytes	Status
tcp	0	128.66.0.10:59108	203.0.113.5:22	203.0.113.5:22	128.66.0.10:59108	--	--	AC

17.2.5.3. Пример настройки firewall failover для кластера из 3 юнитов

Задача:

Настроить firewall failover в кластере маршрутизаторов RTT-1, RTT-2 и RTT-3 со следующими параметрами:

- режим резервирования сессий multicast;
- номер UDP-порта службы резервирования 2000;
- клиентская подсеть: 192.0.2.0/24.

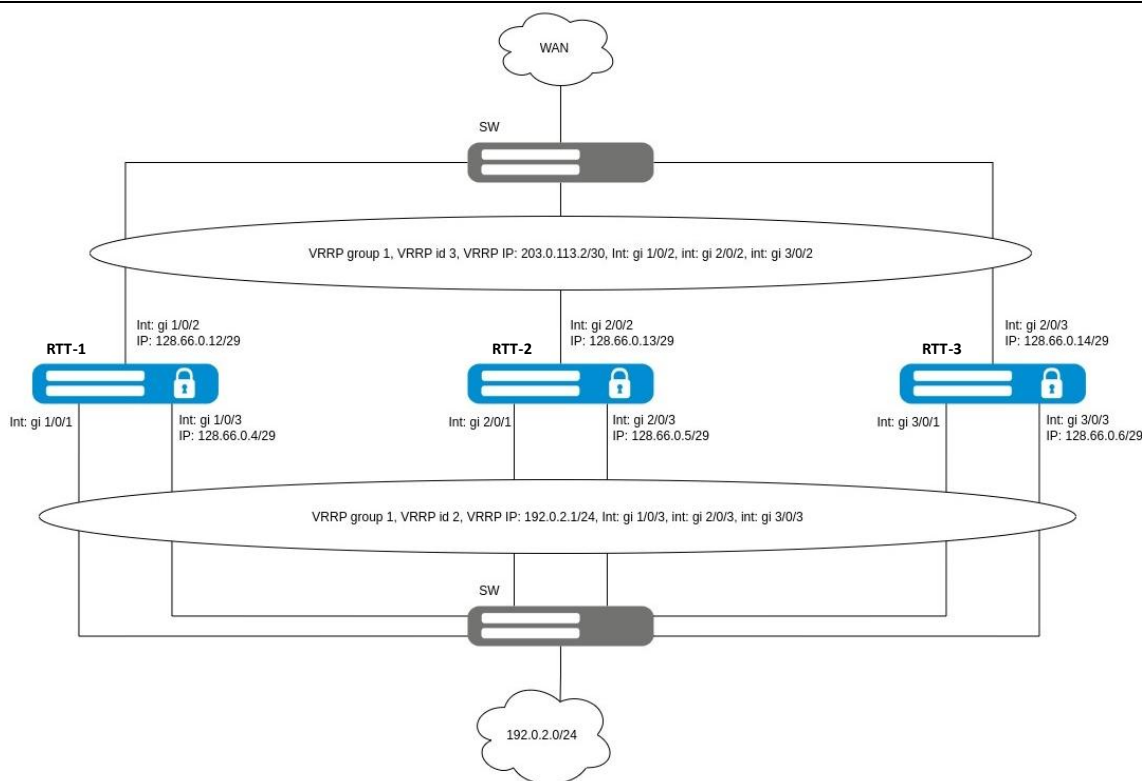


Схема реализации Firewall failover для кластера из 3-х юнитов

Исходная конфигурация кластера:

RTT-1

```
cluster
cluster-interface bridge 1
unit 1
    mac-address cc:9d:a2:71:83:78
exit
unit 2
    mac-address cc:9d:a2:71:82:38
exit
unit 3
    mac-address 68:13:e2:e2:05:28
exit
enable
exit

hostname RTT-1 unit 1
hostname RTT-2 unit 2
hostname RTT-3 unit 3

security zone SYNC
exit
security zone WAN
exit
security zone LAN
exit
```

```
bridge 1
  vlan 1
  security-zone SYNC
  ip address 198.51.100.254/24 unit 1
  ip address 198.51.100.253/24 unit 2
  ip address 198.51.100.252/24 unit 3
  vrrp 1
    ip address 198.51.100.1/24
    priority 254 unit 1
    priority 253 unit 2
    priority 252 unit 3
    group 1
    enable
  exit
enable
exit
```

```
interface gigabitethernet 1/0/1
  mode switchport
  spanning-tree disable
exit
```

```
interface gigabitethernet 1/0/2
  security-zone WAN
  ip address 128.66.0.12/29
  vrrp 3
    ip address 203.0.113.2/30
    group 1
    enable
  exit
exit
```

```
interface gigabitethernet 1/0/3
  security-zone LAN
  ip address 128.66.0.4/29
  vrrp 2
    ip address 192.0.2.1/24
    group 1
    enable
  exit
exit
```

```
interface gigabitethernet 2/0/1
  mode switchport
  spanning-tree disable
exit
```

```
interface gigabitethernet 2/0/2
  security-zone WAN
  ip address 128.66.0.13/29
  vrrp 3
    ip address 203.0.113.2/30
    group 1
    enable
  exit
exit
```

```
interface gigabitethernet 2/0/3
  security-zone LAN
  ip address 128.66.0.5/29
  vrrp 2
    ip address 192.0.2.1/24
```

```
        group 1
        enable
    exit
exit
interface gigabitethernet 3/0/1
    mode switchport
    spanning-tree disable
exit
interface gigabitethernet 3/0/2
    security-zone WAN
    ip address 128.66.0.14/29
    vrrp 3
        ip address 203.0.113.2/30
        group 1
        enable
    exit
exit
interface gigabitethernet 3/0/3
    security-zone LAN
    ip address 128.66.0.6/29
    vrrp 2
        ip address 192.0.2.1/24
        group 1
        enable
    exit
exit

security zone-pair SYNC self
    rule 1
        action permit
        match protocol icmp
        enable
    exit
exit
security zone-pair LAN self
    rule 1
        action permit
        match protocol vrrp
        enable
    exit
exit
security zone-pair WAN self
    rule 1
        action permit
        match protocol vrrp
        enable
    exit
exit
security zone-pair LAN WAN
    rule 1
        action permit
        enable
    exit
exit

ip route 0.0.0.0/0 203.0.113.1
```

Решение:

Сконфигурируем object-group для настройки failover-сервисов:

RTT-1

```
RTT-1(config)# object-group network SYNC_SRC
RTT-1(config-object-group-network)# ip address-range 198.51.100.254 unit 1
RTT-1(config-object-group-network)# ip address-range 198.51.100.253 unit 2
RTT-1(config-object-group-network)# ip address-range 198.51.100.252 unit 3
RTT-1(config-object-group-network)# exit
```

Перейдем к настройке общих параметров для failover-сервисов, а именно к выбору: IP-адреса, с которого будут отправляться сообщения для синхронизации, multicast-группы, multicast IP-адреса, на который будут отправляться сообщения для синхронизации, и VRRP-группы, на основе которой определяется состояние (основной/резервный) маршрутизатора при работе failover-сервисов:

RTT-1

```
RTT-1(config)# ip failover
RTT-1(config-failover)# local-address object-group SYNC_SRC
RTT-1(config-failover)# multicast-address 224.0.0.1
RTT-1(config-failover)# multicast-group 2000
RTT-1(config-failover)# vrrp-group 1
RTT-1(config-failover)# exit
```



При включенном кластере использование object-group в настройке failover-сервисов для local-/remote-адресов обязательно.

Для настройки правил зон безопасности создадим профиль для порта firewall failover:

RTT-1

```
RTT-1(config)# object-group service FAILOVER
RTT-1(config-object-group-service)# port-range 2000
RTT-1(config-object-group-service)# exit
```

Создадим разрешающее правило для зоны безопасности SYNC, разрешив прохождение необходимого трафика для работы firewall failover:

RTT-1

```
RTT-1(config)# security zone-pair SYNC self
RTT-1(config-security-zone-pair)# rule 4
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# match protocol udp
RTT-1(config-security-zone-pair-rule)# match destination-port object-group FAILOVER
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
```

Выполним настройку firewall failover. Настроим режим резервирования сессий multicast и включим firewall failover:

RTT-1

```
RTT-1(config)# ip firewall failover
RTT-1(config-firewall-failover)# sync-type multicast
RTT-1(config-firewall-failover)# enable
RTT-1(config-firewall-failover)# exit
```

После успешного запуска firewall failover можно посмотреть информацию о сервисе с помощью команды:

RTT-1

```
RTT-1# show ip firewall failover
Communication interface:          bridge 1
Status:                           Running
Bytes sent:                        8160
Bytes received:                    15520
Packets sent:                      1002
Packets received:                  1938
Send errors:                       0
Receive errors:                    0
```

Также возможно узнать текущее состояние firewall failover сервиса, выполнив команду:

RTT-1

```
RTT-1# show high-availability state
AP Tunnels:
  State:                            Disabled
  Last state change:                 --
DHCP option 82 table:
  State:                            Disabled
  Last state change:                 --
DHCP server:
  State:                            Disabled
  Last state change:                 --
crypto-sync:
  State:                            Disabled
Firewall sessions and NAT translations:
VRF:
  Tracking VRRP Group                1
  Tracking VRRP Group state:         Master
  State:                            Successful synchronization
  Fault Reason:                      --
  Last synchronization:              2025-10-02 16:53:30
```

Сгенерируем одну клиентскую сессию из LAN в WAN.

Посмотреть firewall-сессии, которые синхронизируются между устройствами, можно командами:

RTT-1

```
RTT-1# show ip firewall sessions failover internal
```

```
Codes: E - expected, U - unreplied,
A - assured, C - confirmed
```

Prot	Aging	Inside source	Inside destination	Outside source	Outside destination	Pkts	Bytes	Status
tcp	0	192.0.2.10:44812	128.66.1.1:22	128.66.1.1:22	192.0.2.10:44812	--	--	AC

RTT-2

```
RTT-2# show ip firewall sessions failover external
```

```
Codes: E - expected, U - unreplied,
A - assured, C - confirmed
```

Prot	Aging	Inside source	Inside destination	Outside source	Outside destination	Pkts	Bytes	Status
tcp	0	192.0.2.10:44812	128.66.1.1:22	128.66.1.1:22	192.0.2.10:44812	--	--	AC

RTT-3

```
RTT-3# show ip firewall sessions failover external
```

```
Codes: E - expected, U - unreplied,
A - assured, C - confirmed
```

Prot	Aging	Inside source	Inside destination	Outside source	Outside destination	Pkts	Bytes	Status
tcp	0	192.0.2.10:44812	128.66.1.1:22	128.66.1.1:22	192.0.2.10:44812	--	--	AC

Посмотреть счетчики для кэшей firewall failover можно командой:

RTT-1

```
RTT-1# show ip firewall failover cache
```

Internal sessions cache counters:

Active entries:	1
Added:	5
Deleted:	4
Updated:	4
Failed adding:	0
No memory left:	0
No space left:	0
Failed deleting:	0
No entry found:	0
Failed updating:	0
No entry found:	0

External sessions cache counters:

Active entries:	0
Added:	0
Deleted:	0
Updated:	0
Installed to Kernel:	0
Failed adding:	0
No memory left:	0
No space left:	0
Failed deleting:	0
No entry found:	0

Failed updating:	0
No entry found:	0
Failed installing to Kernel:	0

17.2.6. Настройка DHCP failover

DHCP-failover позволяет обеспечить высокую доступность службы DHCP.

Алгоритм настройки DHCP failover описан в разделе **Настройка DHCP failover**.

17.2.6.1. Пример настройки

Задача:

Настроить DHCP failover в кластере маршрутизаторов RTT-1 и RTT-2 со следующими параметрами:

- в качестве default-router используется IP-адрес VRRP;
- установить в качестве необходимого режима работы резервирования active-standby;
- клиентская подсеть: 192.0.2.0/24.

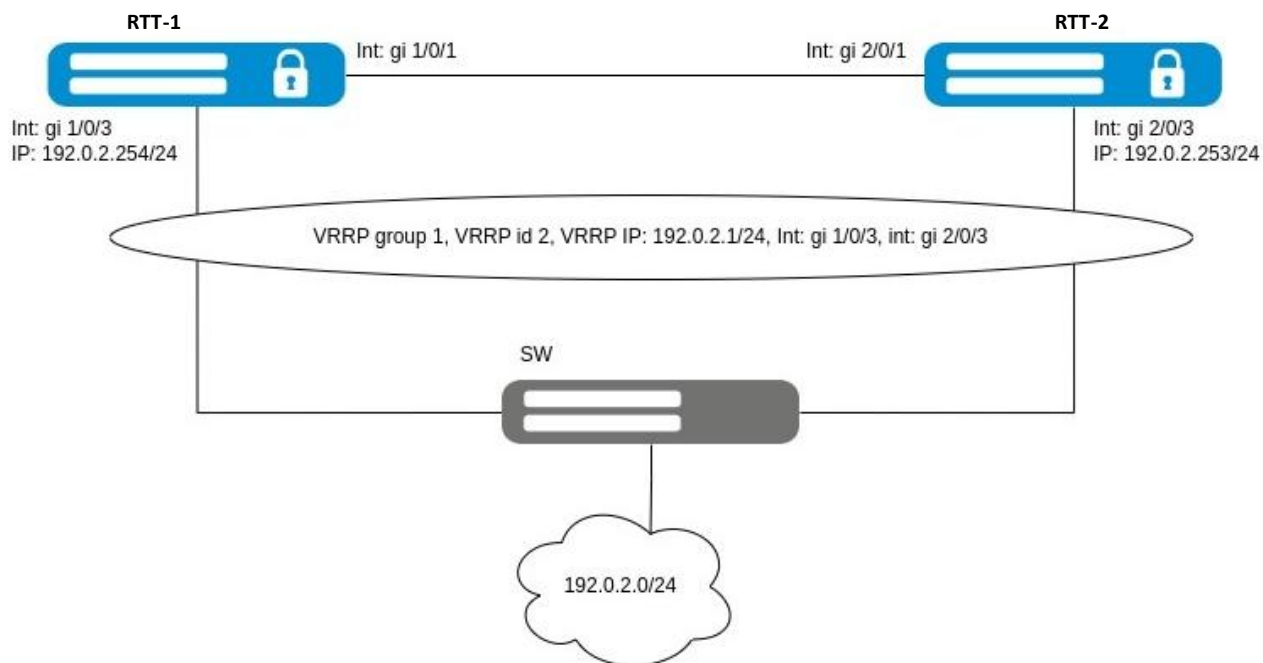


Схема реализации DHCP failover

Исходная конфигурация кластера:

RTT-1

```
cluster
  cluster-interface bridge 1
  unit 1
```

```
        mac-address a2:00:00:10:c0:00
    exit
    unit 2
        mac-address a2:00:00:10:d0:00
    exit
    enable
exit

hostname RTT-1 unit 1
hostname RTT-2 unit 2

object-group service DHCP_SERVER
    port-range 67
exit
object-group service DHCP_CLIENT
    port-range 68
exit

security zone SYNC
exit
security zone LAN
exit

bridge 1
    vlan 1
        security-zone SYNC
        ip address 198.51.100.254/24 unit 1
        ip address 198.51.100.253/24 unit 2
    vrrp 1
        ip address 198.51.100.1/24
        priority 254 unit 1
        priority 253 unit 2
        group 1
        enable
    exit
    enable
exit

interface gigabitethernet 1/0/1
    mode switchport
    spanning-tree disable
exit
interface gigabitethernet 1/0/3
    security-zone LAN
    ip address 192.0.2.254/24
    vrrp 2
        ip address 192.0.2.1/24
        group 1
        enable
    exit
exit
interface gigabitethernet 2/0/1
    mode switchport
    spanning-tree disable
exit
interface gigabitethernet 2/0/3
    security-zone LAN
    ip address 192.0.2.253/24
```

```
vrrp 2
  ip address 192.0.2.1/24
  group 1
  enable
exit
exit

security zone-pair SYNC self
  rule 1
    action permit
    match protocol icmp
    enable
  exit
exit
security zone-pair LAN self
  rule 1
    action permit
    match protocol vrrp
    enable
  exit
  rule 2
    action permit
    match protocol udp
    match source-port object-group DHCP_CLIENT
    match destination-port object-group DHCP_SERVER
    enable
  exit
exit

ip dhcp-server
ip dhcp-server pool TRUSTED
  network 192.0.2.0/24
  address-range 192.0.2.10-192.0.2.100
  default-router 192.0.2.1
exit
```

Решение:

Сконфигурируем object-group для настройки failover-сервисов:

RTT-1

```
RTT-1(config)# object-group network SYNC_SRC
RTT-1(config-object-group-network)# ip address-range 198.51.100.254 unit 1
RTT-1(config-object-group-network)# ip address-range 198.51.100.253 unit 2
RTT-1(config-object-group-network)# exit
RTT-1(config)# object-group network SYNC_DST
RTT-1(config-object-group-network)# ip address-range 198.51.100.253 unit 1
RTT-1(config-object-group-network)# ip address-range 198.51.100.254 unit 2
RTT-1(config-object-group-network)# exit
```

Перейдем к настройке общих параметров для failover-сервисов, а именно к выбору: IP-адреса, с которого будут отправляться сообщения для синхронизации, IP-адреса получателя сообщений для синхронизации и VRRP-группу, на основе которой определяется состояние (основной/резервный) маршрутизатора при работе failover-сервисов:

RTT-1

```
RTT-1(config)# ip failover
RTT-1(config-failover)# local-address object-group SYNC_SRC
RTT-1(config-failover)# remote-address object-group SYNC_DST
RTT-1(config-failover)# vrrp-group 1
RTT-1(config-failover)# exit
```

Перейдем к настройке резервирования DHCP-сервера, укажем режим работы резервирования и включим DHCP-failover:

RTT-1

```
RTT-1(config)# ip dhcp-server failover
RTT-1(config-dhcp-server-failover)# mode active-standby
RTT-1(config-dhcp-server-failover)# enable
RTT-1(config-dhcp-server-failover)# exit
```



Для работы в кластере необходимо использовать режим active-standby.

Создадим разрешающее правило для зоны безопасности SYNC, разрешив прохождение необходимого трафика для работы DHCP failover:

RTT-1

```
RTT-1(config)# object-group service SYNC
RTT-1(config-object-group-service)# port-range 873
RTT-1(config-object-group-service)# exit
RTT-1(config)# security zone-pair SYNC self
RTT-1(config-security-zone-pair)# rule 4
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# match protocol tcp
RTT-1(config-security-zone-pair-rule)# match destination-port object-group SYNC
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
```

Посмотреть состояние резервирования DHCP-сервера можно с помощью команды:

RTT-1

```
RTT-1# show ip dhcp server failover
VRF:
Mode: Active-Standby
Role: Master
State: Synchronized
Last synchronization: 2025-02-12 07:56:40
```

Посмотреть состояние резервирования сессий DHCP можно с помощью команды:

RTT-1

```
RTT-1# show high-availability state
AP Tunnels:
  State: Disabled
  Last state change: --
DHCP option 82 table:
  State: Disabled
  Last state change: --
DHCP server:
VRF:
  Mode: Active-Standby
  State: Successful synchronization
  Last synchronization: 2025-02-12 07:56:36
crypto-sync:
  State: Disabled
Firewall sessions and NAT translations:
  State: Disabled
```

Выданные адреса DHCP можно просмотреть с помощью команды:

RTT-1

```
RTT-1# show ip dhcp binding
IP address      MAC / Client ID      Binding type      Lease expires at
-----
192.0.2.10      e4:5a:d4:01:18:04    active           2025-02-13 07:56:09
```

17.2.6.2. *Пример настройки нескольких экземпляров DHCP failover, каждый в своем VRF*

Задача:

Настроить два экземпляра DHCP failover, каждый в своём VRF, в кластере маршрутизаторов RTT-1 и RTT-2 со следующими параметрами:

- в качестве default-router используется IP-адрес VRRP;
- установить в качестве необходимого режима работы резервирования active-standby;
- настроить приоритеты у разных DHCP failover так, чтобы один из юнитов кластера был Master в одном VRF, а в другом был Backup;
- клиентская подсеть в первом VRF: 192.0.2.0/24;
- клиентская подсеть в втором VRF: 128.66.0.0/24.

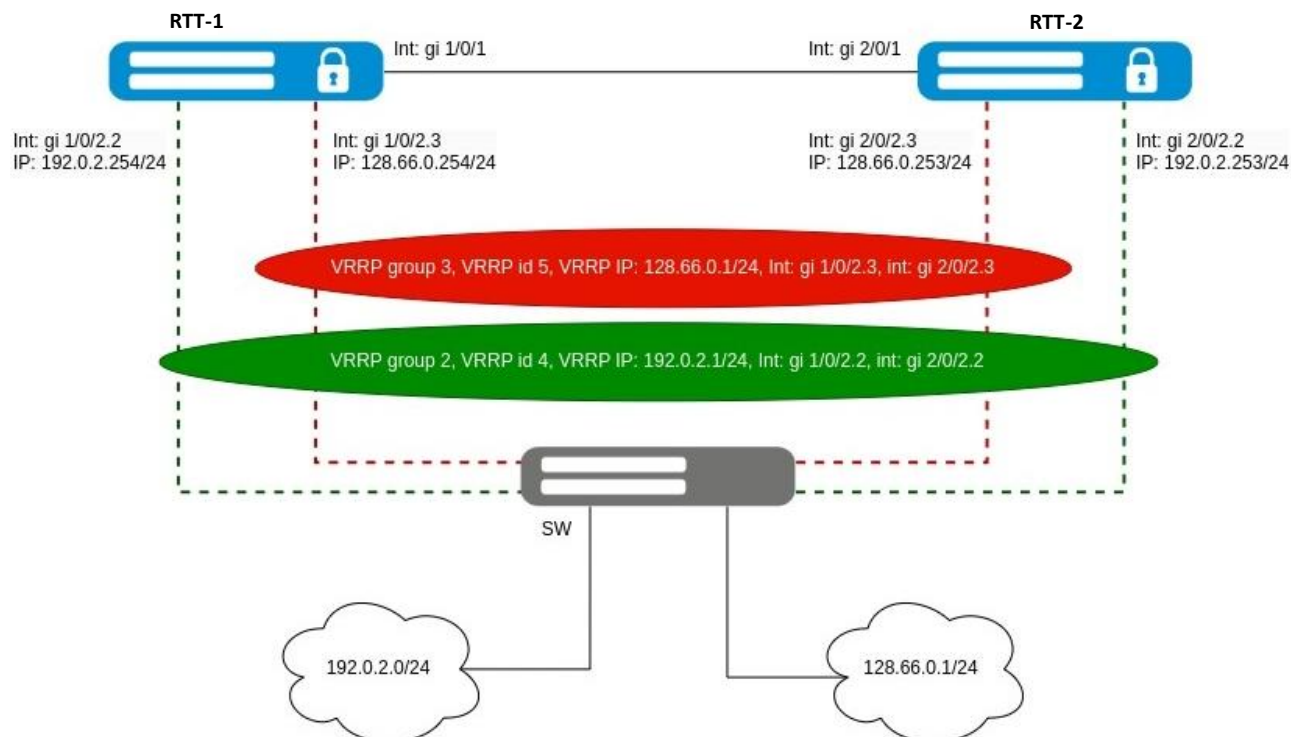


Схема реализации DHCP failover в нескольких VRF

Исходная конфигурация кластера:

RTT-1

```
cluster
  cluster-interface bridge 1
  unit 1
    mac-address a2:00:00:10:c0:00
  exit
  unit 2
    mac-address a2:00:00:10:d0:00
  exit
  enable
exit

hostname RTT-1 unit 1
hostname RTT-2 unit 2

object-group service DHCP_SERVER
  port-range 67
exit
object-group service DHCP_CLIENT
  port-range 68
exit

ip vrf LAN_ONE
exit
ip vrf LAN_TWO
```

exit

```
security zone SYNC
exit
security zone LAN_ONE
    ip vrf forwarding LAN_ONE
exit
security zone LAN_TWO
    ip vrf forwarding LAN_TWO
exit
```

```
bridge 1
    vlan 1
    security-zone SYNC
    ip address 198.51.100.254/24 unit 1
    ip address 198.51.100.253/24 unit 2
    vrrp 1
        ip address 198.51.100.1/24
        priority 254 unit 1
        priority 253 unit 2
        group 1
        enable
    exit
    enable
exit
```

```
interface gigabitethernet 1/0/1
    mode switchport
exit
interface gigabitethernet 1/0/2.2
    ip vrf forwarding LAN_ONE
    security-zone LAN_ONE
    ip address 192.0.2.254/24
    vrrp 4
        ip address 192.0.2.1/24
        priority 120
        group 2
        enable
    exit
exit
interface gigabitethernet 1/0/2.3
    ip vrf forwarding LAN_TWO
    security-zone LAN_TWO
    ip address 128.66.0.254/24
    vrrp 5
        ip address 128.66.0.1/24
        priority 110
        group 3
        enable
    exit
exit
interface gigabitethernet 2/0/1
    mode switchport
exit
interface gigabitethernet 2/0/2.2
    ip vrf forwarding LAN_ONE
    security-zone LAN_ONE
    ip address 192.0.2.253/24
    vrrp 4
```



```
        ip address 192.0.2.1/24
        priority 110
        group 2
        enable
    exit
exit
interface gigabitethernet 2/0/2.3
    ip vrf forwarding LAN_TWO
    security-zone LAN_TWO
    ip address 128.66.0.253/24
    vrrp 5
        ip address 128.66.0.1/24
        priority 120
        group 3
        enable
    exit
exit

security zone-pair SYNC self
    rule 1
        action permit
        match protocol icmp
        enable
    exit
exit
security zone-pair LAN_ONE self
    rule 1
        action permit
        match protocol vrrp
        enable
    exit
    rule 2
        action permit
        match protocol udp
        match source-port object-group DHCP_CLIENT
        match destination-port object-group DHCP_SERVER
        enable
    exit
exit
security zone-pair LAN_TWO self
    rule 1
        action permit
        match protocol vrrp
        enable
    exit
    rule 2
        action permit
        match protocol udp
        match source-port object-group DHCP_CLIENT
        match destination-port object-group DHCP_SERVER
        enable
    exit
exit

ip dhcp-server vrf LAN_ONE
ip dhcp-server pool LAN_ONE vrf LAN_ONE
    network 192.0.2.0/24
    address-range 192.0.2.10-192.0.2.253
```

```
default-router 192.0.2.1
exit
ip dhcp-server vrf LAN_TWO
ip dhcp-server pool LAN_TWO vrf LAN_TWO
network 128.66.0.0/24
address-range 128.66.0.10-128.66.0.253
default-router 128.66.0.1
exit
```

Решение:

Сконфигурируем object-group для настройки DHCP failover-сервисов:

RTT-1

```
RTT-1(config)# object-group network DST_LAN_ONE
RTT-1(config-object-group-network)# ip address-range 192.0.2.253 unit 1
RTT-1(config-object-group-network)# ip address-range 192.0.2.254 unit 2
RTT-1(config-object-group-network)# exit
RTT-1(config)# object-group network DST_LAN_TWO
RTT-1(config-object-group-network)# ip address-range 128.66.0.253 unit 1
RTT-1(config-object-group-network)# ip address-range 128.66.0.254 unit 2
RTT-1(config-object-group-network)# exit
RTT-1(config)# object-group network SRC_LAN_ONE
RTT-1(config-object-group-network)# ip address-range 192.0.2.254 unit 1
RTT-1(config-object-group-network)# ip address-range 192.0.2.253 unit 2
RTT-1(config-object-group-network)# exit
RTT-1(config)# object-group network SRC_LAN_TWO
RTT-1(config-object-group-network)# ip address-range 128.66.0.254 unit 1
RTT-1(config-object-group-network)# ip address-range 128.66.0.253 unit 2
RTT-1(config-object-group-network)# exit
```

Перейдем к настройке ip failover для каждого VRF, настроим там local-address/remote-address и укажем привязки к соответствующим VRRP-group, на основе которых будет определяться, кто из маршрутизаторов будет выдавать адреса:

RTT-1

```
RTT-1(config)# ip failover vrf LAN_ONE
RTT-1(config-failover)# local-address object-group SRC_LAN_ONE
RTT-1(config-failover)# remote-address object-group DST_LAN_ONE
RTT-1(config-failover)# vrrp-group 2
RTT-1(config-failover)# exit
RTT-1(config)# ip failover vrf LAN_TWO
RTT-1(config-failover)# local-address object-group SRC_LAN_TWO
RTT-1(config-failover)# remote-address object-group DST_LAN_TWO
RTT-1(config-failover)# vrrp-group 3
RTT-1(config-failover)# exit
```

Перейдем к настройке DHCP failover, каждый в своем VRF. Для каждого экземпляра необходимо указать режим работы Active-Standby, а также включить его:

RTT-1

```
RTT-1(config)# ip dhcp-server failover vrf LAN_ONE
RTT-1(config-dhcp-server-failover)# mode active-standby
RTT-1(config-dhcp-server-failover)# enable
RTT-1(config-dhcp-server-failover)# exit
RTT-1(config)# ip dhcp-server failover vrf LAN_TWO
RTT-1(config-dhcp-server-failover)# mode active-standby
RTT-1(config-dhcp-server-failover)# enable
RTT-1(config-dhcp-server-failover)# exit
```

Разрешим в настройках firewall работу dhcp-failover в соответствующих зонах:

RTT-1

```
RTT-1(config)# object-group service SYNC
RTT-1(config-object-group-service)# port-range 873
RTT-1(config-object-group-service)# exit
RTT-1(config)# security zone-pair LAN_ONE self
RTT-1(config-security-zone-pair)# rule 3
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# match protocol tcp
RTT-1(config-security-zone-pair-rule)# match destination-port object-group SYNC
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
RTT-1(config)# security zone-pair LAN_TWO self
RTT-1(config-security-zone-pair)# rule 3
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# match protocol tcp
RTT-1(config-security-zone-pair-rule)# match destination-port object-group SYNC
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
RTT-1(config)# exit
```

Посмотреть статус работы DHCP-failover можно с помощью команды, один из экземпляров должен быть в Role - Master, второй в Role - Backup:

RTT-1

```
RTT-1# show ip dhcp server failover vrf LAN_ONE
VRF:                               LAN_ONE
Mode:                               Active-Standby
Role:                               Master
State:                              Synchronized
Last synchronization:              2025-02-18 09:34:44
RTT-1# show ip dhcp server failover vrf LAN_TWO
VRF:                               LAN_TWO
Mode:                               Active-Standby
Role:                               Backup
State:                              Synchronized
Last synchronization:              2025-02-18 09:34:46
```

Также статусы работы DHCP-серверов можно посмотреть с помощью команды:

RTT-1

```
RTT-1# show high-availability state
DHCP server:
VRF:
    Mode: LAN_TWO
    State: Active-Standby
    Last synchronization: Successful synchronization
    2025-02-18 09:34:30
VRF:
    Mode: LAN_ONE
    State: Active-Standby
    Last synchronization: Successful synchronization
    2025-02-18 09:34:28
crypto-sync:
    State: Disabled
Firewall sessions and NAT translations:
    State: Disabled
```

Выданные адреса DHCP можно просмотреть с помощью команды:

RTT-1

```
RTT-1# show ip dhcp binding vrf LAN_ONE
IP address      MAC / Client ID      Binding type      Lease expires at
-----
192.0.2.10      50:52:e5:02:0c:00    active           2025-02-19 09:34:06
RTT-1# show ip dhcp binding vrf LAN_TWO
IP address      MAC / Client ID      Binding type      Lease expires at
-----
128.66.0.10     50:6d:ae:02:0e:00    active           2025-02-19 09:34:09
```

17.2.7. Настройка SNMP

Протокол SNMP (Simple Network Management Protocol) реализует модель «менеджер–агент» для централизованного управления сетевыми устройствами: агенты, установленные на устройствах, собирают данные, структурированные в MIB, а менеджер запрашивает информацию, мониторит состояние сети, контролирует производительность и вносит изменения в конфигурацию оборудования.

Подробный алгоритм настройки SNMP описан в документе «Мониторинг маршрутизаторов по SNMP».

17.2.7.1. Пример настройки

Схема реализации SNMP

Задача:

- обеспечить возможность мониторинга сети через management-интерфейс каждого устройства в кластере;
- обеспечить возможность мониторинга состояния сети и внесения изменений в конфигурацию устройства, выполняющего роль VRRP Master;
- устройство управления (MGMT) доступно по IP-адресу 192.0.2.10.

Исходная конфигурация кластера:

```
cluster
  cluster-interface bridge 1
  unit 1
    mac-address a2:00:00:10:c0:00
  exit
  unit 2
    mac-address a2:00:00:10:d0:00
  exit
  enable
exit

hostname RTT-1 unit 1
hostname RTT-2 unit 2

security zone SYNC
exit
security zone MGMT
exit

bridge 1
  vlan 1
  security-zone SYNC
  ip address 198.51.100.254/24 unit 1
  ip address 198.51.100.253/24 unit 2
  vrrp 1
    ip address 198.51.100.1/24
    priority 254 unit 1
    priority 253 unit 2
    group 1
    enable
  exit
  enable
exit

interface gigabitethernet 1/0/1
  mode switchport
  spanning-tree disable
exit
interface gigabitethernet 1/0/2
  security-zone MGMT
  ip address 192.0.2.254/24
  vrrp 2
    ip address 192.0.2.1/24
    group 1
    enable
  exit
exit
interface gigabitethernet 2/0/1
  mode switchport
  spanning-tree disable
exit
interface gigabitethernet 2/0/2
  security-zone MGMT
  ip address 192.0.2.253/24
  vrrp 2
```

```
ip address 192.0.2.1/24
group 1
enable
exit
exit

security zone-pair SYNC self
rule 1
action permit
match protocol icmp
enable
exit
exit

security zone-pair MGMT self
rule 1
action permit
match protocol vrrp
enable
exit
exit
```

Решение:

Создадим профиль SNMP-портов, предоставляющий доступ в MGMT зону безопасности:

RTT-1

```
RTT-1(config)# object-group service SNMP
RTT-1(config-object-group-service)# port-range 161
RTT-1(config-object-group-service)# port-range 162
RTT-1(config-object-group-service)# exit
```

Добавим правило, предусматривающее проверку, что порт назначения UDP-пакетов соответствует профилю SNMP-портов:

RTT-1

```
RTT-1(config)# security zone-pair MGMT self
RTT-1(config-security-zone-pair)# rule 2
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# match protocol udp
RTT-1(config-security-zone-pair-rule)# match destination-port object-group SNMP
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
```

Активируем SNMP-сервер, настроив параметр snmp-community для обеспечения аутентификации и корректного доступа к данным мониторинга:

RTT-1

```
RTT-1(config)# snmp-server
RTT-1(config)# snmp-server community cluster rw
```

Благодаря данной настройке обеспечивается возможность централизованного мониторинга и управления юнитами кластера как отдельными устройствами, так и устройством, выполняющим роль VRRP Master:

RTT-1

```
snmpset -v2c -c cluster 192.0.2.253 .1.3.6.1.2.1.1.5.0 s 'RTT-1'  
SNMPv2-MIB::sysName.0 = STRING: RTT-1  
snmpset -v2c -c cluster 192.0.2.254 .1.3.6.1.2.1.1.5.0 s 'RTT-2'  
SNMPv2-MIB::sysName.0 = STRING: RTT-2  
snmpset -v2c -c cluster 192.0.2.1 .1.3.6.1.2.1.1.5.0 s 'VRRP-Master'  
SNMPv2-MIB::sysName.0 = STRING: VRRP-Master
```

17.2.8. Настройка Source NAT

Source NAT (SNAT) представляет собой механизм, осуществляющий замену исходного IP-адреса в заголовках IP-пакетов, проходящих через сетевой шлюз. При передаче трафика из внутренней (локальной) сети в внешнюю (публичную) сеть исходный адрес заменяется на один из назначенных публичных IP-адресов шлюза. В ряде случаев осуществляется дополнительное преобразование исходного порта (NATP – Network Address and Port Translation), что обеспечивает корректное направление обратного трафика. При поступлении пакетов из публичной сети в локальную происходит обратная процедура – восстановление оригинальных значений IP-адреса и порта для обеспечения корректной маршрутизации внутри внутренней сети.

Алгоритм Source NAT описан в разделе *Пример настройки Destination NAT*.

17.2.8.1. Пример настройки

Задача:

- предоставить доступ в Интернет хостам, находящимся в локальной сети;
- клиентская подсеть: 192.0.2.0/24;
- публичный IP-адрес – VIP-адрес на интерфейсе.

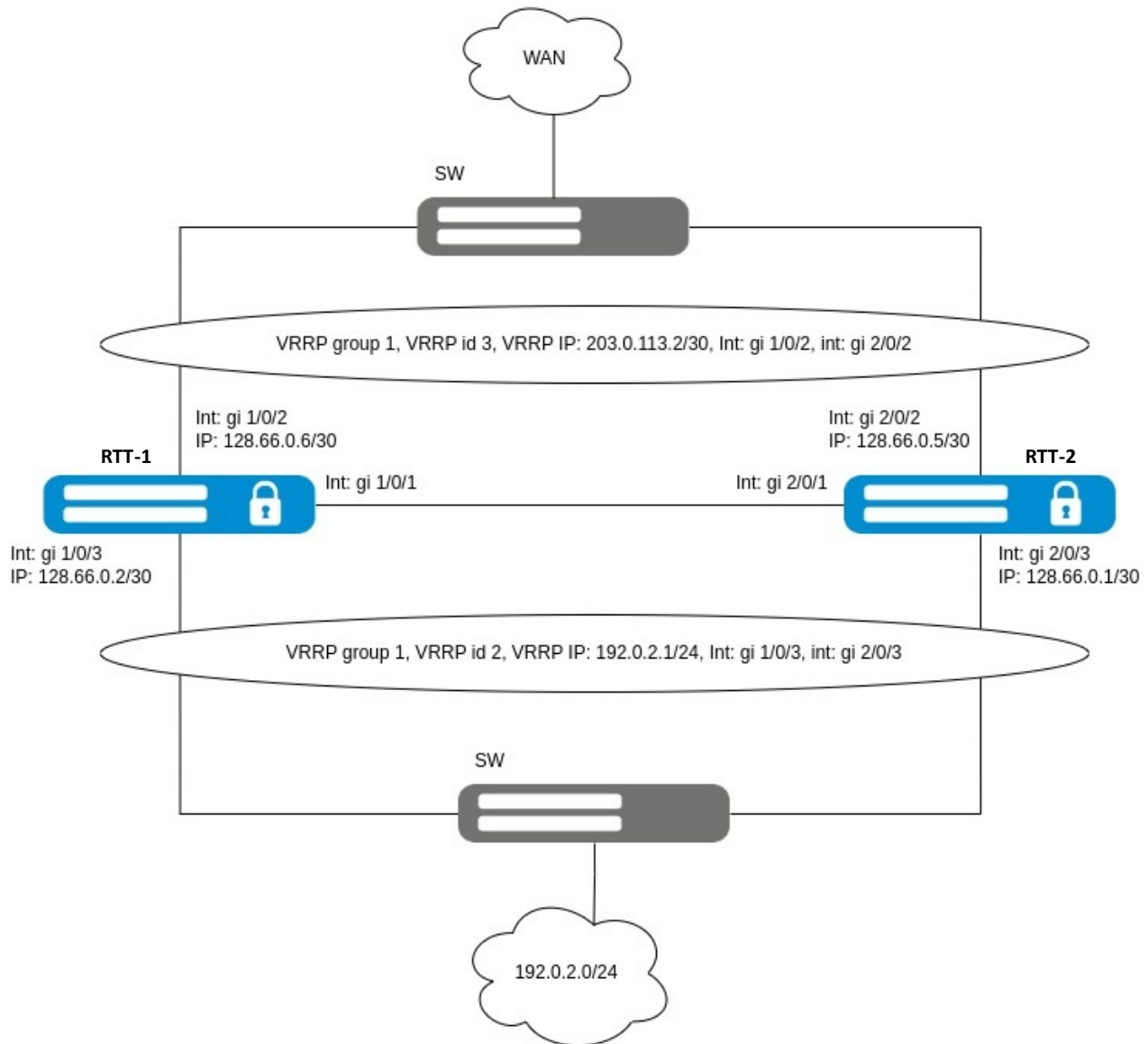


Схема реализации Source NAT

Исходная конфигурация кластера:

```
cluster
cluster-interface bridge 1
unit 1
mac-address a2:00:00:10:c0:00
exit
unit 2
mac-address a2:00:00:10:d0:00
exit
enable
exit

hostname RTT-1 unit 1
hostname RTT-2 unit 2
```



```
security zone SYNC
exit
security zone LAN
exit
security zone WAN
exit

bridge 1
  vlan 1
    security-zone SYNC
    ip address 198.51.100.254/24 unit 1
    ip address 198.51.100.253/24 unit 2
  vrrp 1
    ip address 198.51.100.1/24
    priority 254 unit 1
    priority 253 unit 2
    group 1
    enable
  exit
  enable
exit

interface gigabitethernet 1/0/1
  mode switchport
  spanning-tree disable
exit
interface gigabitethernet 1/0/2
  security-zone WAN
  ip address 128.66.0.6/30
  vrrp 3
    ip address 203.0.113.2/30
    group 1
    enable
  exit
exit
interface gigabitethernet 1/0/3
  security-zone LAN
  ip address 128.66.0.2/30
  vrrp 2
    ip address 192.0.2.1/24
    group 1
    enable
  exit
exit
interface gigabitethernet 2/0/1
  mode switchport
  spanning-tree disable
exit
interface gigabitethernet 2/0/2
  security-zone WAN
  ip address 128.66.0.5/30
  vrrp 3
    ip address 203.0.113.2/30
    group 1
    enable
  exit
exit
interface gigabitethernet 2/0/3
```

```
security-zone LAN
ip address 128.66.0.1/30
vrrp 2
    ip address 192.0.2.1/24
    group 1
    enable
exit
exit

security zone-pair SYNC self
    rule 1
        action permit
        match protocol icmp
        enable
    exit
exit
security zone-pair LAN self
    rule 1
        action permit
        match protocol vrrp
        enable
    exit
exit
security zone-pair WAN self
    rule 1
        action permit
        match protocol vrrp
        enable
    exit
exit
security zone-pair LAN WAN
    rule 1
        action permit
        enable
    exit
exit
```

Решение:

Создадим список IP-адресов, которые будут иметь возможность выхода в Интернет:

RTT-1

```
RTT-1(config)# object-group network INTERNET_USERS
RTT-1(config-object-group-network)# ip address-range 192.0.2.2-192.0.2.255
RTT-1(config-object-group-network)# exit
```

Создадим пул исходных NAT-адресов, в который включим виртуальный IP-адрес (VIP), назначенный WAN-интерфейсу:

RTT-1

```
RTT-1(config)# nat source
RTT-1(config-snat)# pool TRANSLATE_ADDRESS
RTT-1(config-snat-pool)# ip address-range 203.0.113.2
RTT-1(config-snat-pool)# exit
```

Добавим набор правил SNAT. В атрибутах набора укажем применение правил исключительно для пакетов, направляемых в зону WAN. При этом правила осуществляют проверку адреса источника на принадлежность к пулу INTERNET_USERS и выполняют трансляцию исходного адреса в VIP IP-адрес интерфейса:

RTT-1

```
RTT-1(config-snat)# ruleset SNAT
RTT-1(config-snat-ruleset)# to zone WAN
RTT-1(config-snat-ruleset)# rule 1
RTT-1(config-snat-rule)# match source-address object-group INTERNET_USERS
RTT-1(config-snat-rule)# action source-nat pool TRANSLATE_ADDRESS
RTT-1(config-snat-rule)# enable
RTT-1(config-snat-rule)# exit
RTT-1(config-snat-ruleset)# exit
RTT-1(config-snat)# exit
```

Просмотр таблицы NAT трансляций осуществляется посредством следующей команды:

RTT-1

```
RTT-1# show ip nat translations
Prot  Inside source      Inside destination  Outside source      Outside destination  Pkts  Bytes
----  -
tcp    192.0.2.10:45838    203.0.113.1:22      203.0.113.2:45838   203.0.113.1:22      --     --
```

17.2.9. Настройка Destination NAT

Функция Destination NAT (DNAT) выполняет преобразование IP-адреса назначения в заголовках пакетов, проходящих через сетевой шлюз. DNAT применяется для перенаправления трафика, адресованного на IP-адрес в публичном сегменте сети, на «реальный» IP-адрес сервера, расположенного в локальной сети за шлюзом.

Алгоритм настройки Destination NAT описан в разделе **Конфигурирование Destination NAT**.

17.2.9.1. Пример настройки

Задача:

- организовать публичный доступа к серверу, находящемуся в частной сети и не имеющему публичного сетевого адреса;
- сервер доступен по адресу: 192.0.2.10/24;

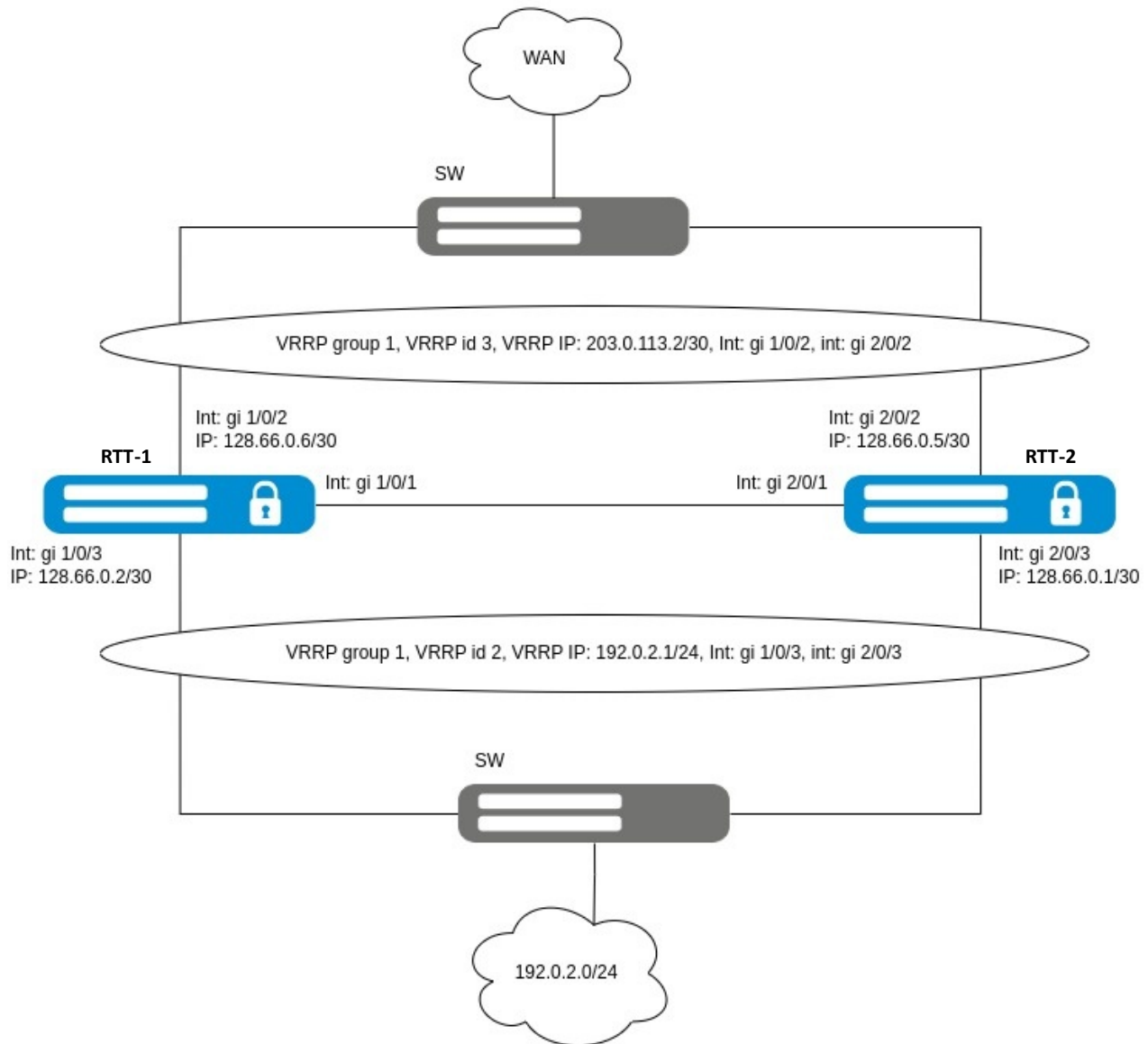


Схема реализации Destination NAT

Исходная конфигурация кластера:

```
cluster
cluster-interface bridge 1
unit 1
    mac-address a2:00:00:10:c0:00
exit
unit 2
    mac-address a2:00:00:10:d0:00
exit
enable
exit

hostname RTT-1 unit 1
hostname RTT-2 unit 2
```

```
security zone SYNC
exit
security zone LAN
exit
security zone WAN
exit

bridge 1
  vlan 1
    security-zone SYNC
    ip address 198.51.100.254/24 unit 1
    ip address 198.51.100.253/24 unit 2
  vrrp 1
    ip address 198.51.100.1/24
    priority 254 unit 1
    priority 253 unit 2
    group 1
    enable
  exit
  enable
exit

interface gigabitethernet 1/0/1
  mode switchport
  spanning-tree disable
exit
interface gigabitethernet 1/0/2
  security-zone WAN
  ip address 128.66.0.6/30
  vrrp 3
    ip address 203.0.113.2/30
    group 1
    enable
  exit
exit
interface gigabitethernet 1/0/3
  security-zone LAN
  ip address 128.66.0.2/30
  vrrp 2
    ip address 192.0.2.1/24
    group 1
    enable
  exit
exit
interface gigabitethernet 2/0/1
  mode switchport
  spanning-tree disable
exit
interface gigabitethernet 2/0/2
  security-zone WAN
  ip address 128.66.0.5/30
  vrrp 3
    ip address 203.0.113.2/30
    group 1
    enable
  exit
exit
interface gigabitethernet 2/0/3
```

```
security-zone LAN
ip address 128.66.0.1/30
vrrp 2
    ip address 192.0.2.1/24
    group 1
    enable
exit
exit

security zone-pair SYNC self
    rule 1
        action permit
        match protocol icmp
        enable
    exit
exit
security zone-pair LAN self
    rule 1
        action permit
        match protocol vrrp
        enable
    exit
exit
security zone-pair WAN self
    rule 1
        action permit
        match protocol vrrp
        enable
    exit
exit
```

Решение:

Создадим профиль адреса сервера из WAN-сети, с которого будем принимать:

RTT-1

```
RTT-1(config)# object-group network INTERNAL
RTT-1(config-object-group-network)# ip address-range 203.0.113.2
RTT-1(config-object-group-network)# exit
```

Создадим профиль сервиса, доступ к которому будем предоставлять:

RTT-1

```
RTT-1(config)# object-group service SERVER_DMZ
RTT-1(config-object-group-service)# port-range 22
RTT-1(config-object-group-service)# exit
```

Войдем в режим конфигурирования функции DNAT и создадим пул адресов, в которые будут транслироваться адреса пакетов, поступающие на адрес 1.2.3.4 из внешней сети:

RTT-1

```
RTT-1(config)# nat destination
```

```
RTT-1(config-dnat)# pool DMZ
RTT-1(config-dnat-pool)# ip address 192.0.2.10
RTT-1(config-dnat-pool)# exit
```

Создадим набор правил «DNAT», в соответствии с которыми будет производиться трансляция адресов. В атрибутах набора укажем, что правила применяются только для пакетов, пришедших из зоны WAN. Набор правил включает в себя требования соответствия данных по адресу и порту назначения (`match destination-address`, `match destination-port`) и по протоколу. Кроме этого, в наборе задано действие, применяемое к данным, удовлетворяющим всем правилам (`action destination-nat`):

RTT-1

```
RTT-1(config-dnat)# ruleset DNAT_SERVER_DMZ
RTT-1(config-dnat-ruleset)# from zone WAN
RTT-1(config-dnat-ruleset)# rule 1
RTT-1(config-dnat-rule)# match protocol tcp
RTT-1(config-dnat-rule)# match destination-address object-group INTERNAL
RTT-1(config-dnat-rule)# match destination-port object-group SERVER_DMZ
RTT-1(config-dnat-rule)# action destination-nat pool DMZ
RTT-1(config-dnat-rule)# enable
RTT-1(config-dnat-rule)# exit
RTT-1(config-dnat-ruleset)# exit
RTT-1(config-dnat)# exit
```

Добавим правило, которое проверяет применение правил исключительно к пакетам, поступающим из зоны WAN. Набор правил включает требования соответствия по адресу назначения (`match destination-address`) и протоколу. Дополнительно в наборе определено действие (`action destination-nat`), которое применяется к данным, удовлетворяющим указанным критериям:

RTT-1

```
RTT-1(config)# security zone-pair WAN LAN
RTT-1(config-security-zone-pair)# rule 1
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# match protocol tcp
RTT-1(config-security-zone-pair-rule)# match destination-port object-group
SERVER_DMZ
RTT-1(config-security-zone-pair-rule)# match destination-nat
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
```

Просмотр таблицы NAT-трансляций осуществляется посредством следующей команды:

RTT-1

```
RTT-1# show ip nat translations
```

Prot	Inside source	Inside destination	Outside source	Outside destination	Pkts	Bytes
tcp	203.0.113.1:41296	192.0.2.10:22	203.0.113.1:41296	203.0.113.2:22	--	--

17.2.10. Настройка BGP

Протокол BGP предназначен для обмена информацией о достижимости подсетей между автономными системами (далее АС), то есть группами маршрутизаторов под единым техническим управлением, использующими протокол внутридоменной маршрутизации для определения маршрутов внутри себя и протокол междоменной маршрутизации для определения маршрутов доставки пакетов в другие АС. Передаваемая информация включает в себя список АС, к которым имеется доступ через данную систему. Выбор наилучших маршрутов осуществляется исходя из правил, принятых в сети.

Алгоритм настройки описан в разделе **Настройка BGP**.

17.2.10.1. Пример настройки eBGP с общим IP-адресом**Задача:**

Настроить BGP-протокол в кластере маршрутизаторов RTT-1 и RTT-2 со следующими параметрами:

- соседство устанавливается только с Active-устройством;
- клиентская подсеть: 192.0.2.0/24;
- анонсирование подсетей, подключенных напрямую;
- собственная AS 64500;
- соседство – подсеть 203.0.113.0/24, vrrp IP-адрес для подключения 203.0.113.1, IP-адрес соседа 203.0.113.2, 64501.

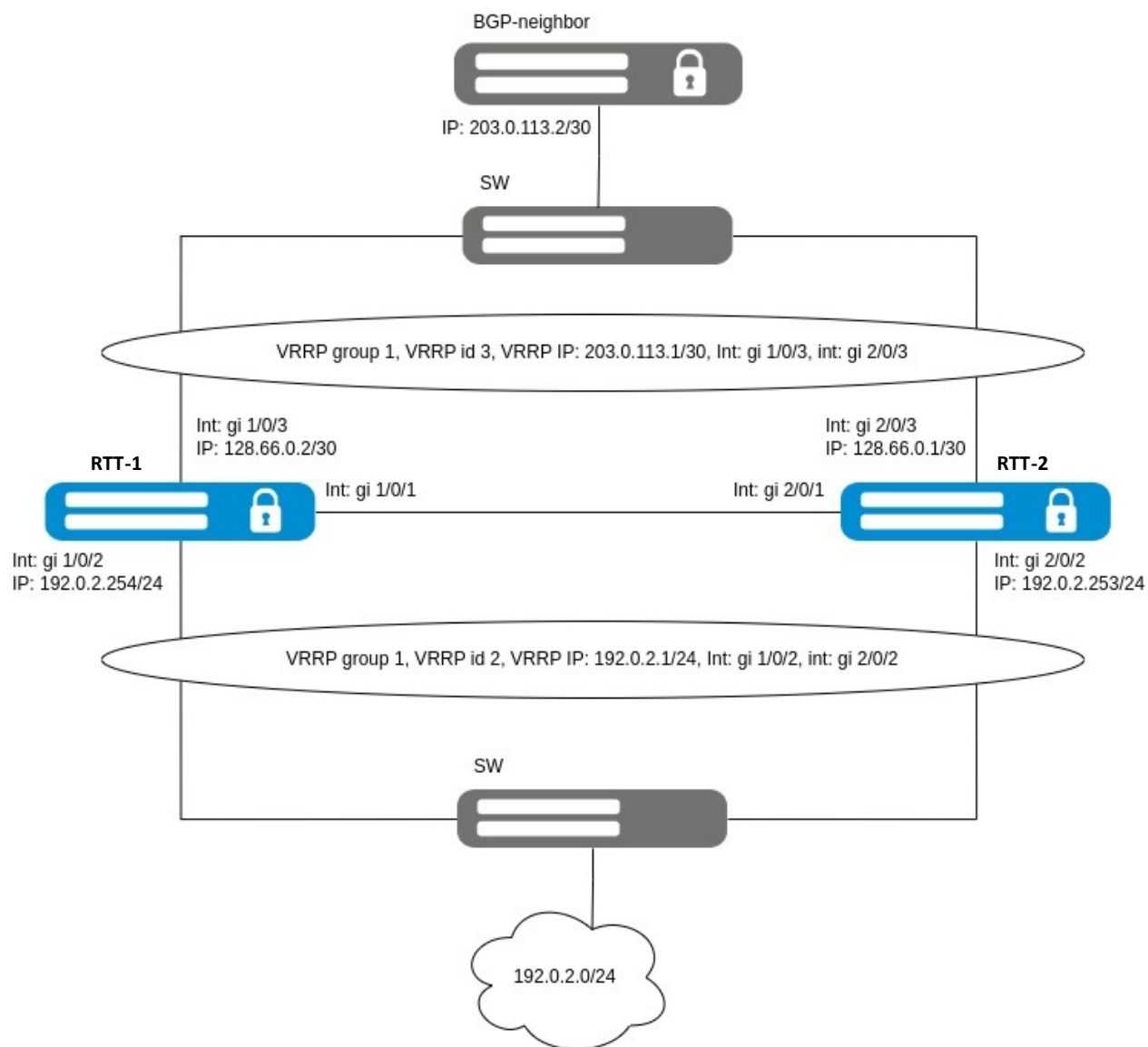


Схема реализации eBGP с общим IP-адресом

Исходная конфигурация кластера:

RTT-1

```
cluster
  cluster-interface bridge 1
  unit 1
    mac-address a2:00:00:10:c0:00
  exit
  unit 2
    mac-address a2:00:00:10:d0:00
  exit
  enable
exit
```

```
hostname RTT-1 unit 1
hostname RTT-2 unit 2

security zone SYNC
exit
security zone LAN
exit
security zone WAN
exit

bridge 1
  vlan 1
    security-zone SYNC
    ip address 198.51.100.254/24 unit 1
    ip address 198.51.100.253/24 unit 2
  vrrp 1
    ip address 198.51.100.1/24
    priority 254 unit 1
    priority 253 unit 2
    group 1
    enable
  exit
  enable
exit

interface gigabitethernet 1/0/1
  mode switchport
  spanning-tree disable
exit
interface gigabitethernet 1/0/2
  security-zone LAN
  ip address 192.0.2.254/24
  vrrp 2
    ip address 192.0.2.1/24
    group 1
    enable
  exit
exit
interface gigabitethernet 1/0/3
  security-zone WAN
  ip address 128.66.0.2/30
  vrrp 3
    ip address 203.0.113.1/30
    group 1
    enable
  exit
exit
interface gigabitethernet 2/0/1
  mode switchport
  spanning-tree disable
exit
interface gigabitethernet 2/0/2
  security-zone LAN
  ip address 192.0.2.253/24
  vrrp 2
    ip address 192.0.2.1/24
    group 1
    enable
  exit
```

```
exit
interface gigabitethernet 2/0/3
  security-zone WAN
  ip address 128.66.0.1/30
  vrrp 3
    ip address 203.0.113.1/30
    group 1
    enable
  exit
exit

security zone-pair SYNC self
  rule 1
    action permit
    match protocol icmp
    enable
  exit
exit

security zone-pair LAN self
  rule 1
    action permit
    match protocol vrrp
    enable
  exit
  rule 2
    action permit
    match protocol ah
    enable
  exit
exit

security zone-pair WAN self
  rule 1
    action permit
    match protocol vrrp
    enable
  exit
exit
```

Решение:

Настроим firewall для приема маршрутизатором BGP-трафика из зоны безопасности WAN:

```
RTT-1(config)# object-group service og_bgp
RTT-1(config-object-group-service)# port-range 179
RTT-1(config-object-group-service)# exit
RTT-1(config)# security zone WAN
RTT-1(config-security-zone)# exit
RTT-1(config)# security zone-pair WAN self
RTT-1(config-security-zone-pair)# rule 2
RTT-1(config-security-zone-pair-rule)# match protocol tcp
RTT-1(config-security-zone-pair-rule)# match destination-port object-group
og_bgp
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
```

Создадим route-map, который будет использоваться в дальнейшем при настройке разрешающих анонсов роутерам из другой AS. В route-map запретим анонсировать подсеть для cluster-interface:

```
RTT-1(config)# route-map bgp-out
RTT-1(config-route-map)# rule 1
RTT-1(config-route-map-rule)# match ip address 198.51.100.0/24
RTT-1(config-route-map-rule)# action deny
RTT-1(config-route-map-rule)# exit
RTT-1(config-route-map)# rule 2
RTT-1(config-route-map-rule)# action permit
RTT-1(config-route-map-rule)# exit
RTT-1(config-route-map)# exit
```

Создадим BGP-процесс для AS 64500 и войдем в режим конфигурирования параметров процесса:

```
RTT-1(config)# router bgp 64500
```

Сконфигурируем анонсирование подсетей, подключенных напрямую:

```
RTT-1(config-bgp)# address-family ipv4 unicast
RTT-1(config-bgp-af)# redistribute connected
RTT-1(config-bgp-af)# exit
```

Создадим eBGP с вышестоящим роутером:

```
RTT-1(config-bgp)# neighbor 203.0.113.2
RTT-1(config-bgp-neighbor)# remote-as 64501
RTT-1(config-bgp-neighbor)# update-source 203.0.113.1
```

И включим обмен IPv4-маршрутами:

```
RTT-1(config-bgp-neighbor)# address-family ipv4 unicast
RTT-1(config-bgp-neighbor-af)# route-map bgp-out out
RTT-1(config-bgp-neighbor-af)# enable
RTT-1(config-bgp-neighbor-af)# exit
```

Включим работу протокола:

```
RTT-1(config-bgp-neighbor)# enable
RTT-1(config-bgp-neighbor)# exit
RTT-1(config-bgp)# enable
RTT-1(config-bgp)# exit
```

Применим конфигурацию на Active-устройстве.

Информацию о BGP-пирах можно посмотреть командой **show bgp neighbors**:

```
RTT-1# show bgp neighbors
BGP neighbor is 203.0.113.1
  BGP state:                               Established
  Type:                                     Static neighbor
  Neighbor address:                        203.0.113.1
  Neighbor AS:                             64501
  Neighbor ID:                             203.0.113.1
```

```

Neighbor caps:                refresh enhanced-refresh restart-aware
AS4
Session:                      external AS4
Source address:               203.0.113.2
Weight:                       0
Hold timer:                   124/180
Keepalive timer:              27/60
RR client:                    No
Address family ipv4 unicast:
  Send-label:                  No
  Default originate:           No
  Default information originate: No
  Outgoing route-map:          bgp-out
  Preference:                  170
  Remove private AS:           No
  Next-hop self:               No
  Next-hop unchanged:          No
  Uptime (d,h:m:s):            00,00:03:13
RTT-2# show bgp neighbors
BGP neighbor is 203.0.113.2
  BGP state:                   Active
  Type:                        Static neighbor
  Neighbor address:             203.0.113.1
  Neighbor AS:                  64501
  Connect delay:                2/5
  Last error:                   Socket: Network is unreachable

```

Таблицу маршрутов протокола BGP можно просмотреть с помощью команды:

```

RTT-1# show bgp ipv4 unicast neighbor 203.0.113.1 advertise-routes
Status codes: u - unicast, b - broadcast, m - multicast, a - anycast
                * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop          Metric  LocPrf    Weight  Path
*> u 192.0.2.0/24    203.0.113.2          --      --        --      64500 ?
* u 192.0.2.0/24    203.0.113.2          --      --        --      64500 ?
*> u 128.66.0.0/30   203.0.113.2          --      --        --      64500 ?
*> u 203.0.113.0/30  203.0.113.2          --      --        --      64500 ?
RTT-1# show bgp ipv4 unicast neighbor 203.0.113.1 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    Weight  Path
*> u 0.0.0.0/0       203.0.113.1          --      100        0       64501 ?

```



В случае выхода из строя Active-устройства BGP будет полностью переустанавливаться со Standby-устройством.

17.2.10.2. *Пример настройки eBGP с каждым участником кластера по индивидуальным IP-адресам*

Задача:

Настроить BGP-протокол в кластере маршрутизаторов RTT-1 и RTT-2 со следующими параметрами:

- соседство устанавливается с каждым маршрутизатором в кластере индивидуально;
- клиентская подсеть: 192.0.2.0/24;
- анонсирование подсетей, подключенных напрямую;
- собственная AS 64500;
- соседство для RTT-1 – подсеть 203.0.113.0/30, IP-адрес для подключения 203.0.113.1, IP-адрес соседа 203.0.113.2, 64501;
- соседство для RTT-2 – подсеть 203.0.113.4/30, IP-адрес для подключения 203.0.113.5, IP-адрес соседа 203.0.113.6, 64502.

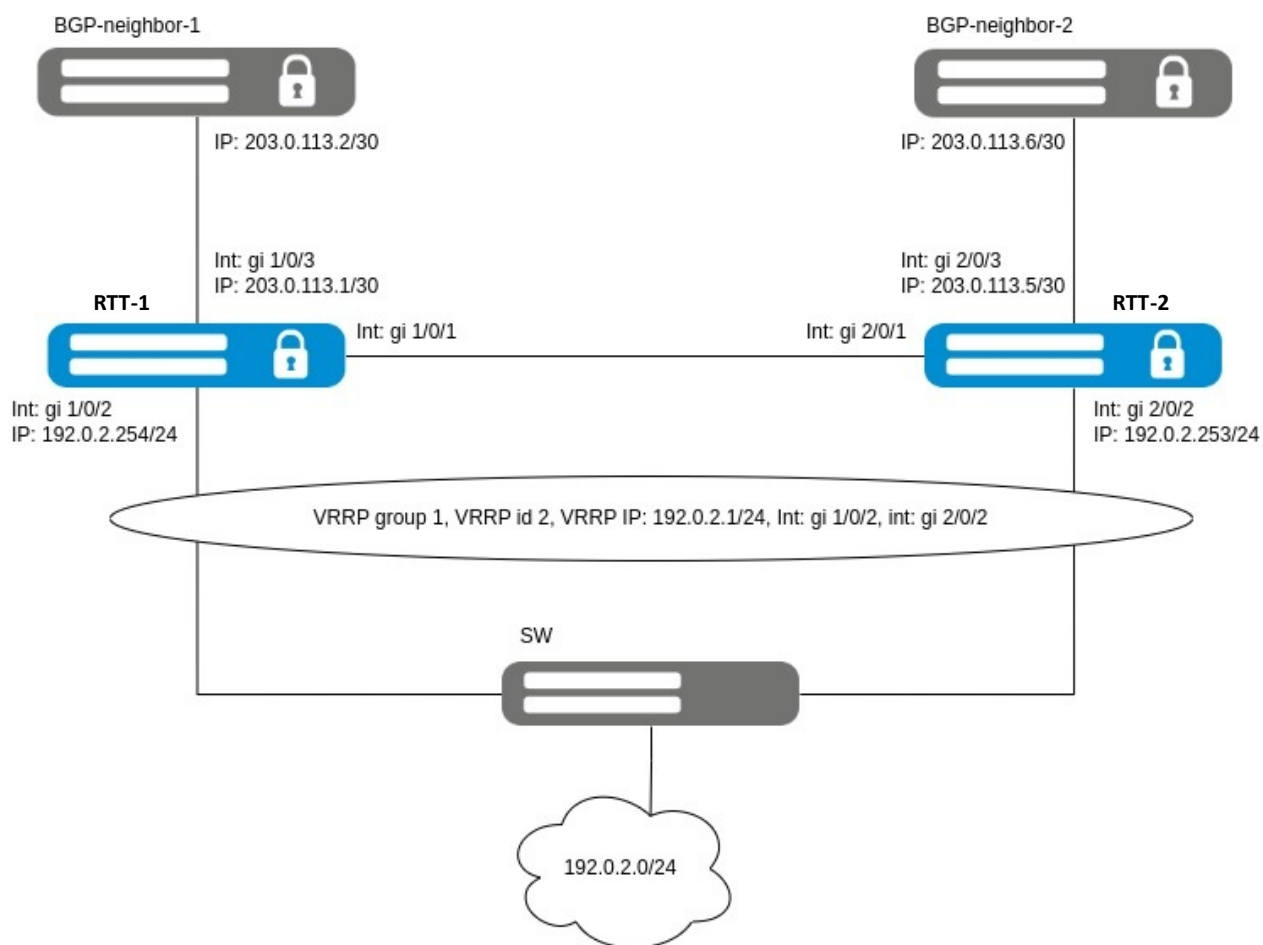


Схема реализации eBGP с каждым участником кластера по индивидуальным IP-адресам

Исходные конфигурации маршрутизаторов в кластере:

RTT-1

```
cluster
cluster-interface bridge 1
unit 1
mac-address a2:00:00:10:c0:00
```

```
exit
unit 2
    mac-address a2:00:00:10:d0:00
exit
enable
exit

hostname RTT-1 unit 1
hostname RTT-2 unit 2

security zone SYNC
exit
security zone LAN
exit
security zone WAN
exit

bridge 1
    vlan 1
        security-zone SYNC
        ip address 198.51.100.254/24 unit 1
        ip address 198.51.100.253/24 unit 2
        vrrp 1
            ip address 198.51.100.1/24
            priority 254 unit 1
            priority 253 unit 2
            group 1
            enable
        exit
    enable
exit

interface gigabitethernet 1/0/1
    mode switchport
    spanning-tree disable
exit
interface gigabitethernet 1/0/2
    security-zone LAN
    ip address 192.0.2.254/24
    vrrp 2
        ip address 192.0.2.1/24
        group 1
        enable
    exit
exit

interface gigabitethernet 1/0/3
    security-zone WAN
    ip address 203.0.113.1/30
exit
interface gigabitethernet 2/0/1
    mode switchport
    spanning-tree disable
exit
interface gigabitethernet 2/0/2
    security-zone LAN
    ip address 192.0.2.253/24
    vrrp 2
```

```
ip address 192.0.2.1/24
group 1
enable
exit
exit

interface gigabitethernet 2/0/3
security-zone WAN
ip address 203.0.113.5/30
exit

security zone-pair SYNC self
rule 1
action permit
match protocol icmp
enable
exit
exit
security zone-pair LAN self
rule 1
action permit
match protocol vrrp
enable
exit
rule 2
action permit
match protocol ah
enable
exit
exit
security zone-pair WAN self
rule 1
action permit
match protocol vrrp
enable
exit
exit
```

Решение:

Настроим firewall для приема маршрутизатором BGP-трафика из зоны безопасности WAN:

```
RTT-1(config)# object-group service og_bgp
RTT-1(config-object-group-service)# port-range 179
RTT-1(config-object-group-service)# exit
RTT-1(config)# security zone-pair WAN self
RTT-1(config-security-zone-pair)# rule 2
RTT-1(config-security-zone-pair-rule)# match protocol tcp
RTT-1(config-security-zone-pair-rule)# match destination-port object-group
og_bgp
RTT-1(config-security-zone-pair-rule)# action permit
RTT-1(config-security-zone-pair-rule)# enable
RTT-1(config-security-zone-pair-rule)# exit
RTT-1(config-security-zone-pair)# exit
```

Создадим track для последующего управления анонсами маршрутов в кластере:


```
RTT-1(config)# track 1
RTT-1(config-track)# track vrrp id 1 state not master
RTT-1(config-track)# enable
RTT-1(config-track)# exit
```

Создадим route-map, который будет использоваться в дальнейшем при настройке разрешающих анонсов роутерам из другой AS. В route-map запретим анонсировать подсеть для cluster-interface, а также настроим управление as-path prepend для управления анонсами bgp:

```
RTT-1(config)# route-map bgp-out
RTT-1(config-route-map)# rule 1
RTT-1(config-route-map-rule)# match ip address 198.51.100.0/24
RTT-1(config-route-map-rule)# action deny
RTT-1(config-route-map-rule)# exit
RTT-1(config-route-map)# rule 2
RTT-1(config-route-map-rule)# action set as-path prepend 64500 track 1
RTT-1(config-route-map-rule)# action permit
RTT-1(config-route-map-rule)# exit
RTT-1(config-route-map)# exit
```

Создадим BGP-процесс для AS 64500 для RTT-1 и войдем в режим конфигурирования параметров процесса:

```
RTT-1(config)# router bgp 64500 unit 1
```

Сконфигурируем анонсирование подсетей, подключенных напрямую:

```
RTT-1(config-bgp)# address-family ipv4 unicast
RTT-1(config-bgp-af)# redistribute connected
RTT-1(config-bgp-af)# exit
```

Создадим eBGP с вышестоящим роутером:

```
RTT-1(config-bgp)# neighbor 203.0.113.2
RTT-1(config-bgp-neighbor)# remote-as 64501
RTT-1(config-bgp-neighbor)# update-source 203.0.113.1
```

И включим обмен IPv4-маршрутами:

```
RTT-1(config-bgp-neighbor)# address-family ipv4 unicast
RTT-1(config-bgp-neighbor-af)# route-map bgp-out out
RTT-1(config-bgp-neighbor-af)# enable
RTT-1(config-bgp-neighbor-af)# exit
```

Включим работу протокола:

```
RTT-1(config-bgp-neighbor)# enable
RTT-1(config-bgp-neighbor)# exit
RTT-1(config-bgp)# enable
RTT-1(config-bgp)# exit
```

Создадим BGP процесс для AS 64500 для RTT-2 и войдем в режим конфигурирования параметров процесса:

```
RTT-1(config)# router bgp 64500 unit 2
```

Сконфигурируем анонсирование подсетей, подключенных напрямую:

```
RTT-1(config-bgp)# address-family ipv4 unicast
RTT-1(config-bgp-af)# redistribute connected
RTT-1(config-bgp-af)# exit
```

Создадим eBGP с вышестоящим роутером:

```
RTT-1(config-bgp)# neighbor 203.0.113.6
RTT-1(config-bgp-neighbor)# remote-as 64502
RTT-1(config-bgp-neighbor)# update-source 203.0.113.5
```

И включим обмен IPv4-маршрутами:

```
RTT-1(config-bgp-neighbor)# address-family ipv4 unicast
RTT-1(config-bgp-neighbor-af)# route-map bgp-out out
RTT-1(config-bgp-neighbor-af)# enable
RTT-1(config-bgp-neighbor-af)# exit
```

Включим работу протокола:

```
RTT-1(config-bgp-neighbor)# enable
RTT-1(config-bgp-neighbor)# exit
RTT-1(config-bgp)# enable
RTT-1(config-bgp)# exit
```

Применим конфигурацию на Active-устройстве.

Информацию о BGP-пирах можно посмотреть командой:

```
RTT-1# show bgp neighbors
BGP neighbor is 203.0.113.2
  BGP state:                               Established
  Type:                                       Static neighbor
  Neighbor address:                         203.0.113.2
  Neighbor AS:                             64501
  Neighbor ID:                             203.0.113.2
  Neighbor caps:                           refresh enhanced-refresh restart-aware
AS4
  Session:                                 external AS4
  Source address:                         203.0.113.1
  Weight:                                  0
  Hold timer:                             107/180
  Keepalive timer:                        20/60
  RR client:                              No
  Address family ipv4 unicast:
    Send-label:                           No
    Default originate:                    No
    Default information originate:        No
    Outgoing route-map:                   bgp-out
    Preference:                           170
    Remove private AS:                    No
    Next-hop self:                        No
```

```

Next-hop unchanged: No
Uptime (d,h:m:s): 00,00:00:28
RTT-2# show bgp neighbors
BGP neighbor is 203.0.113.6
  BGP state: Established
  Type: Static neighbor
  Neighbor address: 203.0.113.6
  Neighbor AS: 64502
  Neighbor ID: 203.0.113.6
  Neighbor caps: refresh enhanced-refresh restart-aware
AS4
  Session: external AS4
  Source address: 203.0.113.5
  Weight: 0
  Hold timer: 144/180
  Keepalive timer: 29/60
  RR client: No
  Address family ipv4 unicast:
    Send-label: No
    Default originate: No
    Default information originate: No
    Outgoing route-map: bgp-out
    Preference: 170
    Remove private AS: No
    Next-hop self: No
    Next-hop unchanged: No
  Uptime (d,h:m:s): 00,00:00:20

```

Таблицу маршрутов протокола BGP можно просмотреть с помощью команды:

```

RTT-1# show bgp ipv4 unicast neighbor 203.0.113.2 advertise-routes
Status codes: u - unicast, b - broadcast, m - multicast, a - anycast
               * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop          Metric  LocPrf    Weight Path
*> u 192.0.2.0/24    203.0.113.1          --      --        -- 64500 ?
* u 192.0.2.0/24    203.0.113.1          --      --        -- 64500 ?
*> u 203.0.113.0/30 203.0.113.1          --      --        -- 64500 ?
RTT-1# show bgp ipv4 unicast neighbor 203.0.113.2 routes
Status codes: u - unicast, b - broadcast, m - multicast, a - anycast
               * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop          Metric  LocPrf    Weight Path
*> u 0.0.0.0/0       203.0.113.2          --      100        0 64501 ?
RTT-2# show bgp ipv4 unicast neighbor 203.0.113.6 advertise-routes
Status codes: u - unicast, b - broadcast, m - multicast, a - anycast
               * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop          Metric  LocPrf    Weight Path
*> u 192.0.2.0/24    203.0.113.5          --      --        -- 64500 64500 ?
*> u 203.0.113.4/30 203.0.113.5          --      --        -- 64500 64500 ?
RTT-2# show bgp ipv4 unicast neighbor 203.0.113.6 routes
Status codes: u - unicast, b - broadcast, m - multicast, a - anycast
               * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop          Metric  LocPrf    Weight Path

```

17.2.11. Настройка DMVPN

DMVPN (Dynamic Multipoint Virtual Private Network) — технология для создания виртуальных частных сетей, с возможностью динамического создания туннелей между узлами. Преимуществом данного решения является высокая масштабируемость и легкость настройки при подключении филиалов к головному офису. DMVPN используется в топологии Hub-and-Spoke, и позволяет строить прямые VPN-туннели Spoke-to-Spoke в дополнение к обычным Spoke-to-Hub туннелям. Это означает, что филиалы смогут общаться друг с другом напрямую, без необходимости прохождения трафика через Hub.

Чтобы установить такое соединение, клиенты (NHC) по зашифрованному IPsec-туннелю отправляют соответствие своего внутреннего (туннельного) адреса и внешнего (NBMA) адреса на NHRP-сервер (NHS). Когда клиент захочет соединиться с другим NHC, он посылает на сервер запрос, чтобы узнать его внешний адрес. Получив ответ от сервера, клиент теперь самостоятельно может устанавливать соединение с удалённым филиалом.

Алгоритм настройки описан в разделе **Настройка DMVPN**.

17.2.11.1. Пример настройки в кластере DMVPN Single Hub Dual Cloud схемы

Задача:

Организовать DMVPN между офисами компании, используя mGRE-туннели, NHRP (Next Hop Resolution Protocol), протокол динамической маршрутизации (BGP), IPsec. В данном примере будет HUB-маршрутизатор, который находится в кластере, и два филиала. HUB — это DMVPN-сервер (NHS), а филиалы — DMVPN-клиенты (NHC).

HUB внешний IP-адрес через Cloud_one — 198.51.100.2/30;

HUB внешний IP-адрес через Cloud_two — 198.51.100.6/30;

SPOKE-1 внешний IP-адрес — 198.51.100.10/30;

SPOKE-2 внешний IP-адрес — 198.51.100.14/30.

Параметры IPsec VPN:

IKE:

- группа Диффи-Хеллмана: 19;
- алгоритм шифрования: AES256;
- алгоритм аутентификации: SHA2-256.

IPsec:

- группа Диффи-Хеллмана: 19;
- алгоритм шифрования: AES256;
- алгоритм аутентификации: SHA2-256.

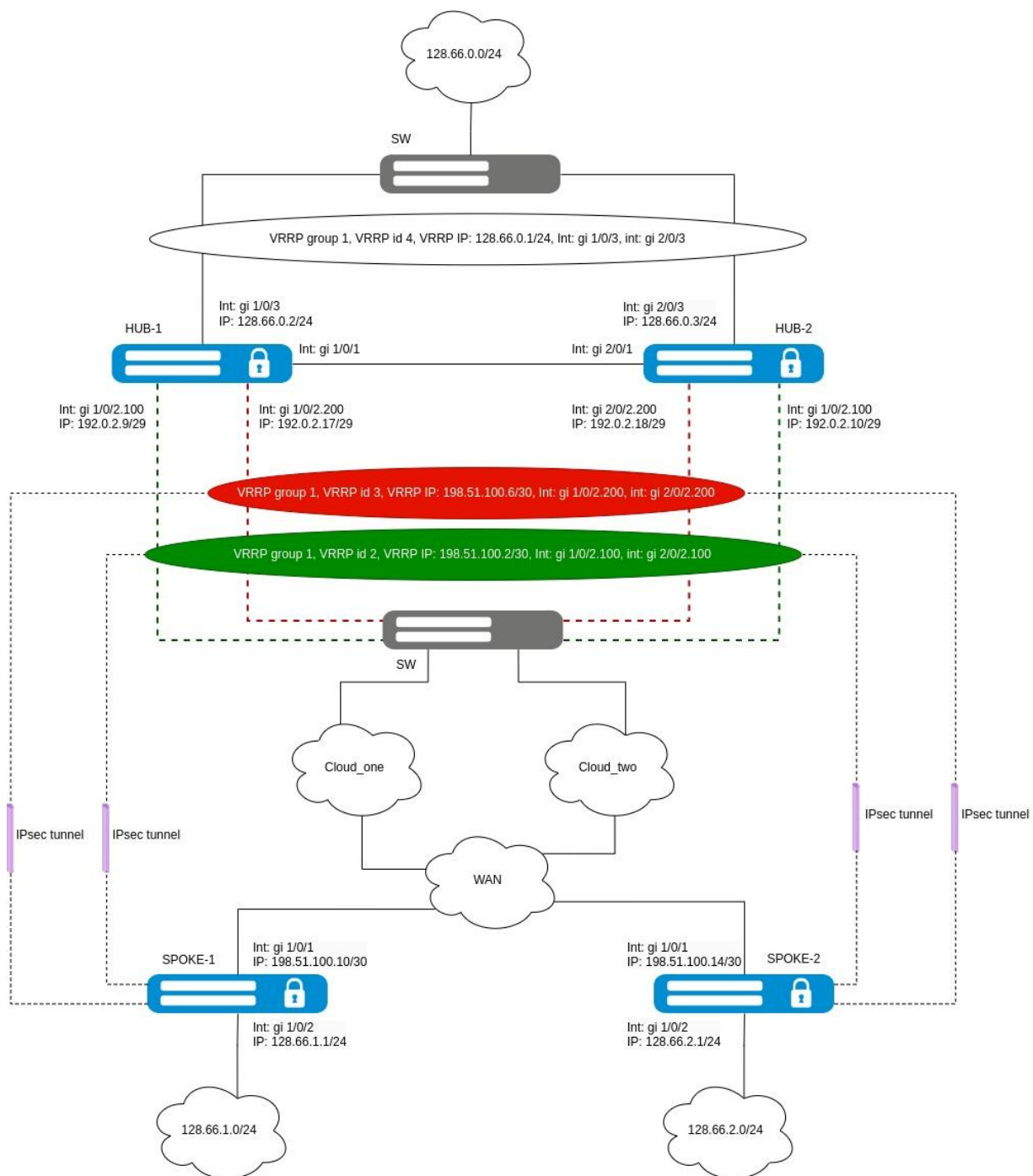


Схема реализации DMVPN Single Hub Dual Cloud в кластере

Исходная конфигурация CLUSTER-HUB:

HUB-1

```
cluster
cluster-interface bridge 1
unit 1
```

```
    mac-address a2:00:00:10:a0:00
exit
unit 2
    mac-address a2:00:00:10:b0:00
exit
enable
exit

hostname HUB-1 unit 1
hostname HUB-2 unit 2

security zone SYNC
exit
security zone LAN
exit
security zone WAN
exit

ip access-list extended LOCAL_1
    rule 1
        action permit
        match source-address 198.51.100.2 255.255.255.255
        enable
    exit
exit
ip access-list extended LOCAL_2
    rule 1
        action permit
        match source-address 198.51.100.6 255.255.255.255
        enable
    exit
exit

route-map PBR_LOCAL
    rule 1
        match ip access-group LOCAL_1
        action set ip next-hop verify-availability 198.51.100.1 1
    exit
    rule 2
        match ip access-group LOCAL_2
        action set ip next-hop verify-availability 198.51.100.5 1
    exit
exit
route-map DMVPN_BGP_OUT_CLOUD_TWO
    rule 1
        match ip address 0.0.0.0/0
        action set metric bgp 2000
    exit
exit
route-map DMVPN_BGP_OUT_CLOUD_ONE
    rule 1
        match ip address 0.0.0.0/0
        action set metric bgp 1000
    exit
exit

ip local policy route-map PBR_LOCAL

bridge 1
```

```
vlan 1
security-zone SYNC
ip address 192.0.2.5/29 unit 1
ip address 192.0.2.6/29 unit 2
vrrp 1
    ip address 192.0.2.1/29
    priority 5 unit 1
    priority 6 unit 2
    group 1
    enable
exit
enable
exit

interface gigabitethernet 1/0/1
    mode switchport
    spanning-tree disable
exit
interface gigabitethernet 1/0/2.100
    security-zone WAN
    ip address 192.0.2.9/29
    vrrp 2
        ip address 198.51.100.2/30
        group 1
        enable
    exit
    wan load-balance nexthop 198.51.100.1
    wan load-balance enable
exit
interface gigabitethernet 1/0/2.200
    security-zone WAN
    ip address 192.0.2.17/29
    vrrp 3
        ip address 198.51.100.6/30
        group 1
        enable
    exit
    wan load-balance nexthop 198.51.100.5
    wan load-balance enable
exit
interface gigabitethernet 1/0/3
    security-zone LAN
    ip address 128.66.0.2/24
    vrrp 4
        ip address 128.66.0.1/24
        group 1
        enable
    exit
exit
interface gigabitethernet 2/0/1
    mode switchport
    spanning-tree disable
exit
interface gigabitethernet 2/0/2.100
    security-zone WAN
    ip address 192.0.2.10/29
    vrrp 2
        ip address 198.51.100.2/30
```

```
    group 1
    enable
  exit
  wan load-balance nexthop 198.51.100.1
  wan load-balance enable
exit
interface gigabitethernet 2/0/2.200
  security-zone WAN
  ip address 192.0.2.18/29
  vrrp 3
    ip address 198.51.100.6/30
    group 1
    enable
  exit
  wan load-balance nexthop 198.51.100.5
  wan load-balance enable
exit
interface gigabitethernet 2/0/3
  security-zone LAN
  ip address 128.66.0.3/24
  vrrp 4
    ip address 128.66.0.1/24
    group 1
    enable
  exit
exit

security zone-pair SYNC self
  rule 1
    action permit
    match protocol vrrp
    enable
  exit
exit
security zone-pair LAN self
  rule 1
    action permit
    match protocol vrrp
    enable
  exit
  rule 2
    action permit
    match protocol ah
    enable
  exit
exit
security zone-pair WAN self
  rule 1
    action permit
    match protocol vrrp
    enable
  exit
  rule 2
    action permit
    match protocol ah
    enable
  exit
exit
```



```
ip route 0.0.0.0/0 wan load-balance rule 1

wan load-balance rule 1
  outbound interface gigabitethernet 1/0/2.100
  outbound interface gigabitethernet 1/0/2.200
  outbound interface gigabitethernet 2/0/2.200
  outbound interface gigabitethernet 2/0/2.100
  enable
exit
```

Исходная конфигурация SPOKE-1:

SPOKE-1

```
hostname SPOKE-1

security zone LAN
exit
security zone WAN
exit

interface gigabitethernet 1/0/1
  security-zone WAN
  ip address 198.51.100.10/30
exit
interface gigabitethernet 1/0/2
  security-zone LAN
  ip address 128.66.1.1/24
exit

ip route 198.51.100.0/30 198.51.100.9
ip route 198.51.100.12/30 198.51.100.9
ip route 198.51.100.4/30 198.51.100.9
```

Исходная конфигурация SPOKE-2:

SPOKE-2

```
hostname SPOKE-2

security zone LAN
exit
security zone WAN
exit

interface gigabitethernet 1/0/1
  security-zone WAN
  ip address 198.51.100.14/30
exit
interface gigabitethernet 1/0/2
  security-zone LAN
  ip address 128.66.2.1/24
exit

ip route 198.51.100.0/30 198.51.100.13
ip route 198.51.100.4/30 198.51.100.13
```

```
ip route 198.51.100.8/30 198.51.100.13
```

Решение:

Конфигурирование HUB

Создадим туннели mGRE, каждый через свой CLOUD, определим принадлежность к зоне безопасности, настроим NHRP и включим туннель и NHRP командой **enable**:

```
HUB-1(config)# security zone DMVPN_C_ONE
HUB-1(config-security-zone)# exit
HUB-1(config)# security zone DMVPN_C_TWO
HUB-1(config-security-zone)# exit
HUB-1(config)# tunnel gre 1
HUB-1(config-gre)# key 1000
HUB-1(config-gre)# ttl 64
HUB-1(config-gre)# mtu 1400
HUB-1(config-gre)# multipoint
HUB-1(config-gre)# security-zone DMVPN_C_ONE
HUB-1(config-gre)# local address 198.51.100.2
HUB-1(config-gre)# ip address 203.0.113.1/25
HUB-1(config-gre)# ip tcp adjust-mss 1360
HUB-1(config-gre)# ip nhrp redirect
HUB-1(config-gre)# ip nhrp multicast dynamic
HUB-1(config-gre)# ip nhrp enable
HUB-1(config-gre)# enable
HUB-1(config-gre)# exit
HUB-1(config)# tunnel gre 2
HUB-1(config-gre)# key 2000
HUB-1(config-gre)# ttl 64
HUB-1(config-gre)# mtu 1400
HUB-1(config-gre)# multipoint
HUB-1(config-gre)# security-zone DMVPN_C_TWO
HUB-1(config-gre)# local address 198.51.100.6
HUB-1(config-gre)# ip address 203.0.113.129/25
HUB-1(config-gre)# ip tcp adjust-mss 1360
HUB-1(config-gre)# ip nhrp redirect
HUB-1(config-gre)# ip nhrp multicast dynamic
HUB-1(config-gre)# ip nhrp enable
HUB-1(config-gre)# enable
HUB-1(config-gre)# exit
```

Произведём настройку протокола динамической маршрутизации для Hub. В примере это будет eBGP, для которого необходимо явно разрешить анонсирование подсетей.

Так как в примере используется два CLOUD, необходимо сделать один из них более приоритетным, используя route-map.

Для ускорения переключения в случае выхода из строя Active устройства в кластере включим также bfd для BGP, а также уменьшим таймер error-wait.

```
HUB-1(config)# route-map DMVPN_BGP_OUT_CLOUD_ONE
HUB-1(config-route-map)# rule 1
HUB-1(config-route-map-rule)# match ip address 0.0.0.0/0
HUB-1(config-route-map-rule)# action set metric bgp 1000
```

```
HUB-1(config-route-map-rule)# exit
HUB-1(config-route-map)# exit
HUB-1(config)# route-map DMVPN_BGP_OUT_CLOUD_TWO
HUB-1(config-route-map)# rule 1
HUB-1(config-route-map-rule)# match ip address 0.0.0.0/0
HUB-1(config-route-map-rule)# action set metric bgp 2000
HUB-1(config-route-map-rule)# exit
HUB-1(config-route-map)# exit
HUB-1(config)# router bgp 64500
HUB-1(config-bgp)# default-information-originate
HUB-1(config-bgp)# timers error-wait 5 10
HUB-1(config-bgp)# peer-group DMVPN_CLOUD_ONE
HUB-1(config-bgp-group)# remote-as 64501
HUB-1(config-bgp-group)# update-source 203.0.113.1
HUB-1(config-bgp-group)# fall-over bfd
HUB-1(config-bgp-group)# address-family ipv4 unicast
HUB-1(config-bgp-group-af)# route-map DMVPN_BGP_OUT_CLOUD_ONE out
HUB-1(config-bgp-group-af)# next-hop-self
HUB-1(config-bgp-group-af)# enable
HUB-1(config-bgp-group-af)# exit
HUB-1(config-bgp-group)# exit
HUB-1(config-bgp)# peer-group DMVPN_CLOUD_TWO
HUB-1(config-bgp-group)# remote-as 64501
HUB-1(config-bgp-group)# update-source 203.0.113.129
HUB-1(config-bgp-group)# fall-over bfd
HUB-1(config-bgp-group)# address-family ipv4 unicast
HUB-1(config-bgp-group-af)# route-map DMVPN_BGP_OUT_CLOUD_TWO out
HUB-1(config-bgp-group-af)# next-hop-self
HUB-1(config-bgp-group-af)# enable
HUB-1(config-bgp-group-af)# exit
HUB-1(config-bgp-group)# exit
HUB-1(config-bgp)# listen-range 203.0.113.0/25
HUB-1(config-bgp-listen)# peer-group DMVPN_CLOUD_ONE
HUB-1(config-bgp-listen)# enable
HUB-1(config-bgp-listen)# exit
HUB-1(config-bgp)# listen-range 203.0.113.128/25
HUB-1(config-bgp-listen)# peer-group DMVPN_CLOUD_TWO
HUB-1(config-bgp-listen)# enable
HUB-1(config-bgp-listen)# exit
HUB-1(config-bgp)# address-family ipv4 unicast
HUB-1(config-bgp-af)# redistribute static
HUB-1(config-bgp-af)# exit
HUB-1(config-bgp)# enable
HUB-1(config-bgp)# exit
```

Произведём настройку IPsec для Hub, для начала настроим `ike proposal`, `ike policy` и `ike gateway`. В `ike gateway` дополнительно настроим `drd`, для ускорения перестроения туннелей в случае, если выйдет из строя Active-устройство:

```
HUB-1(config)# security ike proposal ike_proposal
HUB-1(config-ike-proposal)# authentication algorithm sha2-256
HUB-1(config-ike-proposal)# encryption algorithm aes256
HUB-1(config-ike-proposal)# dh-group 19
HUB-1(config-ike-proposal)# exit
HUB-1(config)#
HUB-1(config)# security ike policy ike_policy
HUB-1(config-ike-policy)# pre-shared-key ascii-text encrypted 8CB5107EA7005AFF
```

```
HUB-1(config-ike-policy)# proposal ike_proposal
HUB-1(config-ike-policy)# exit
HUB-1(config)# security ike gateway ike_gateway_cloud_one
HUB-1(config-ike-gw)# version v2-only
HUB-1(config-ike-gw)# ike-policy ike_policy
HUB-1(config-ike-gw)# local address 198.51.100.2
HUB-1(config-ike-gw)# local network 198.51.100.2/32 protocol gre
HUB-1(config-ike-gw)# remote address any
HUB-1(config-ike-gw)# remote network any protocol gre
HUB-1(config-ike-gw)# mode policy-based
HUB-1(config-ike-gw)# mobike disable
HUB-1(config-ike-gw)# dead-peer-detection action clear
HUB-1(config-ike-gw)# dead-peer-detection interval 10
HUB-1(config-ike-gw)# dead-peer-detection retransmit timeout 5
HUB-1(config-ike-gw)# dead-peer-detection retransmit tries 2
HUB-1(config-ike-gw)# exit
HUB-1(config)# security ike gateway ike_gateway_cloud_two
HUB-1(config-ike-gw)# version v2-only
HUB-1(config-ike-gw)# ike-policy ike_policy
HUB-1(config-ike-gw)# local address 198.51.100.6
HUB-1(config-ike-gw)# local network 198.51.100.6/32 protocol gre
HUB-1(config-ike-gw)# remote address any
HUB-1(config-ike-gw)# remote network any protocol gre
HUB-1(config-ike-gw)# mode policy-based
HUB-1(config-ike-gw)# mobike disable
HUB-1(config-ike-gw)# dead-peer-detection action clear
HUB-1(config-ike-gw)# dead-peer-detection interval 10
HUB-1(config-ike-gw)# dead-peer-detection retransmit timeout 5
HUB-1(config-ike-gw)# dead-peer-detection retransmit tries 2
HUB-1(config-ike-gw)# exit
HUB-1(config)#
HUB-1(config)# security ike session uniqueids replace
```

Затем настроим IPsec proposal, IPsec policy и IPsec vpn туннели через каждый CLOUD:

```
HUB-1(config)# security ipsec proposal ipsec_proposal
HUB-1(config-ipsec-proposal)# authentication algorithm sha2-256
HUB-1(config-ipsec-proposal)# encryption algorithm aes256
HUB-1(config-ipsec-proposal)# pfs dh-group 19
HUB-1(config-ipsec-proposal)# exit
HUB-1(config)# security ipsec policy ipsec_policy
HUB-1(config-ipsec-policy)# proposal ipsec_proposal
HUB-1(config-ipsec-policy)# exit
HUB-1(config)# security ipsec vpn ipsec_dynamic_cloud_one
HUB-1(config-ipsec-vpn)# type transport
HUB-1(config-ipsec-vpn)# ike establish-tunnel route
HUB-1(config-ipsec-vpn)# ike gateway ike_gateway_cloud_one
HUB-1(config-ipsec-vpn)# ike ipsec-policy ipsec_policy
HUB-1(config-ipsec-vpn)# enable
HUB-1(config-ipsec-vpn)# exit
HUB-1(config)# security ipsec vpn ipsec_dynamic_cloud_two
HUB-1(config-ipsec-vpn)# type transport
HUB-1(config-ipsec-vpn)# ike establish-tunnel route
HUB-1(config-ipsec-vpn)# ike gateway ike_gateway_cloud_two
HUB-1(config-ipsec-vpn)# ike ipsec-policy ipsec_policy
HUB-1(config-ipsec-vpn)# enable
HUB-1(config-ipsec-vpn)# exit
```

Скорректируем правила зоны безопасности WAN, разрешим протоколы для GRE over IPSec-туннеля:

```
HUB-1(config)# object-group service ISAKMP_PORT
HUB-1(config-object-group-service)# port-range 500
HUB-1(config-object-group-service)# port-range 4500
HUB-1(config-object-group-service)# exit
HUB-1(config)# security zone-pair WAN self
HUB-1(config-security-zone-pair)# rule 3
HUB-1(config-security-zone-pair-rule)# action permit
HUB-1(config-security-zone-pair-rule)# match protocol udp
HUB-1(config-security-zone-pair-rule)# match destination-port object-group
ISAKMP_PORT
HUB-1(config-security-zone-pair-rule)# enable
HUB-1(config-security-zone-pair-rule)# exit
HUB-1(config-security-zone-pair)# rule 4
HUB-1(config-security-zone-pair-rule)# action permit
HUB-1(config-security-zone-pair-rule)# match protocol esp
HUB-1(config-security-zone-pair-rule)# enable
HUB-1(config-security-zone-pair-rule)# exit
HUB-1(config-security-zone-pair)# rule 5
HUB-1(config-security-zone-pair-rule)# action permit
HUB-1(config-security-zone-pair-rule)# match protocol gre
HUB-1(config-security-zone-pair-rule)# enable
HUB-1(config-security-zone-pair-rule)# exit
HUB-1(config-security-zone-pair)# exit
```

Настроим правила зон безопасности DMVPN_C_ONE и DMVPN_C_TWO, разрешим прохождение трафика для протоколов BGP, BFD, ICMP:

```
HUB-1(config)# object-group service BGP
HUB-1(config-object-group-service)# port-range 179
HUB-1(config-object-group-service)# exit
HUB-1(config)# object-group service BFD
HUB-1(config-object-group-service)# port-range 3784
HUB-1(config-object-group-service)# exit
HUB-1(config)# security zone-pair DMVPN_C_ONE self
HUB-1(config-security-zone-pair)# rule 1
HUB-1(config-security-zone-pair-rule)# action permit
HUB-1(config-security-zone-pair-rule)# match protocol icmp
HUB-1(config-security-zone-pair-rule)# enable
HUB-1(config-security-zone-pair-rule)# exit
HUB-1(config-security-zone-pair)# rule 2
HUB-1(config-security-zone-pair-rule)# action permit
HUB-1(config-security-zone-pair-rule)# match protocol tcp
HUB-1(config-security-zone-pair-rule)# match destination-port object-group BGP
HUB-1(config-security-zone-pair-rule)# enable
HUB-1(config-security-zone-pair-rule)# exit
HUB-1(config-security-zone-pair)# rule 3
HUB-1(config-security-zone-pair-rule)# action permit
HUB-1(config-security-zone-pair-rule)# match protocol udp
HUB-1(config-security-zone-pair-rule)# match destination-port object-group BFD
HUB-1(config-security-zone-pair-rule)# enable
HUB-1(config-security-zone-pair-rule)# exit
HUB-1(config-security-zone-pair)# exit
HUB-1(config)# security zone-pair DMVPN_C_TWO self
```

```
HUB-1(config-security-zone-pair)# rule 1
HUB-1(config-security-zone-pair-rule)# action permit
HUB-1(config-security-zone-pair-rule)# match protocol icmp
HUB-1(config-security-zone-pair-rule)# enable
HUB-1(config-security-zone-pair-rule)# exit
HUB-1(config-security-zone-pair)# rule 2
HUB-1(config-security-zone-pair-rule)# action permit
HUB-1(config-security-zone-pair-rule)# match protocol tcp
HUB-1(config-security-zone-pair-rule)# match destination-port object-group BGP
HUB-1(config-security-zone-pair-rule)# enable
HUB-1(config-security-zone-pair-rule)# exit
HUB-1(config-security-zone-pair)# rule 3
HUB-1(config-security-zone-pair-rule)# action permit
HUB-1(config-security-zone-pair-rule)# match protocol udp
HUB-1(config-security-zone-pair-rule)# match destination-port object-group BFD
HUB-1(config-security-zone-pair-rule)# enable
HUB-1(config-security-zone-pair-rule)# exit
HUB-1(config-security-zone-pair)# exit
```

Скорректируем правила зоны безопасности LAN, разрешим прохождение трафика между зонами LAN и DMVPN_C_ONE/DMVPN_C_TWO:

```
HUB-1(config)# security zone-pair LAN DMVPN_C_ONE
HUB-1(config-security-zone-pair)# rule 1
HUB-1(config-security-zone-pair-rule)# action permit
HUB-1(config-security-zone-pair-rule)# enable
HUB-1(config-security-zone-pair-rule)# exit
HUB-1(config-security-zone-pair)# exit
HUB-1(config)# security zone-pair LAN DMVPN_C_TWO
HUB-1(config-security-zone-pair)# rule 1
HUB-1(config-security-zone-pair-rule)# action permit
HUB-1(config-security-zone-pair-rule)# enable
HUB-1(config-security-zone-pair-rule)# exit
HUB-1(config-security-zone-pair)# exit
HUB-1(config)# security zone-pair DMVPN_C_ONE LAN
HUB-1(config-security-zone-pair)# rule 1
HUB-1(config-security-zone-pair-rule)# action permit
HUB-1(config-security-zone-pair-rule)# enable
HUB-1(config-security-zone-pair-rule)# exit
HUB-1(config-security-zone-pair)# exit
HUB-1(config)# security zone-pair DMVPN_C_TWO LAN
HUB-1(config-security-zone-pair)# rule 1
HUB-1(config-security-zone-pair-rule)# action permit
HUB-1(config-security-zone-pair-rule)# enable
HUB-1(config-security-zone-pair-rule)# exit
HUB-1(config-security-zone-pair)# exit
```

Конфигурирование SPOKE-1

Создадим туннели mGRE, каждый через свой CLOUD, определим принадлежность к зоне безопасности, настроим NHRP и включим туннель и NHRP командой **enable**:

```
SPOKE-1(config)# security zone DMVPN_C_TWO
SPOKE-1(config-security-zone)# exit
SPOKE-1(config)# security zone DMVPN_C_ONE
SPOKE-1(config-security-zone)# exit
SPOKE-1(config)# tunnel gre 1
```

```
SPOKE-1(config-gre)# key 1000
SPOKE-1(config-gre)# ttl 64
SPOKE-1(config-gre)# mtu 1400
SPOKE-1(config-gre)# multipoint
SPOKE-1(config-gre)# security-zone DMVPN_C_ONE
SPOKE-1(config-gre)# local address 198.51.100.10
SPOKE-1(config-gre)# ip address 203.0.113.2/25
SPOKE-1(config-gre)# ip tcp adjust-mss 1360
SPOKE-1(config-gre)# ip nhrp holding-time 60
SPOKE-1(config-gre)# ip nhrp shortcut
SPOKE-1(config-gre)# ip nhrp map 203.0.113.1 198.51.100.2
SPOKE-1(config-gre)# ip nhrp nhs 203.0.113.1
SPOKE-1(config-gre)# ip nhrp multicast nhs
SPOKE-1(config-gre)# ip nhrp enable
SPOKE-1(config-gre)# enable
SPOKE-1(config-gre)# exit
SPOKE-1(config)# tunnel gre 2
SPOKE-1(config-gre)# key 2000
SPOKE-1(config-gre)# ttl 64
SPOKE-1(config-gre)# mtu 1400
SPOKE-1(config-gre)# multipoint
SPOKE-1(config-gre)# security-zone DMVPN_C_TWO
SPOKE-1(config-gre)# local address 198.51.100.10
SPOKE-1(config-gre)# ip address 203.0.113.130/25
SPOKE-1(config-gre)# ip tcp adjust-mss 1360
SPOKE-1(config-gre)# ip nhrp holding-time 60
SPOKE-1(config-gre)# ip nhrp shortcut
SPOKE-1(config-gre)# ip nhrp map 203.0.113.129 198.51.100.6
SPOKE-1(config-gre)# ip nhrp nhs 203.0.113.129
SPOKE-1(config-gre)# ip nhrp multicast nhs
SPOKE-1(config-gre)# ip nhrp enable
SPOKE-1(config-gre)# enable
SPOKE-1(config-gre)# exit
```

Произведём настройку протокола динамической маршрутизации для SPOKE-1. В примере это будет eBGP, для которого необходимо явно разрешить анонсирование подсетей. Анонсируем LAN подсети в сторону HUB используя network в address-family.

Для ускорения переключения в случае выхода из строя Active-устройства в кластере включим также bfd для BGP, а также уменьшим таймер error-wait.

```
SPOKE-1(config)# route-map DMVPN_BGP_OUT
SPOKE-1(config-route-map)# rule 1
SPOKE-1(config-route-map-rule)# exit
SPOKE-1(config-route-map)# exit
SPOKE-1(config)# router bgp 64501
SPOKE-1(config-bgp)# timers error-wait 5 10
SPOKE-1(config-bgp)# neighbor 203.0.113.1
SPOKE-1(config-bgp-neighbor)# remote-as 64500
SPOKE-1(config-bgp-neighbor)# allow-local-as 10
SPOKE-1(config-bgp-neighbor)# update-source 203.0.113.2
SPOKE-1(config-bgp-neighbor)# fall-over bfd
SPOKE-1(config-bgp-neighbor)# address-family ipv4 unicast
SPOKE-1(config-bgp-neighbor-af)# route-map DMVPN_BGP_OUT out
SPOKE-1(config-bgp-neighbor-af)# enable
SPOKE-1(config-bgp-neighbor-af)# exit
```

```
SPOKE-1(config-bgp-neighbor)# enable
SPOKE-1(config-bgp-neighbor)# exit
SPOKE-1(config-bgp)# neighbor 203.0.113.129
SPOKE-1(config-bgp-neighbor)# remote-as 64500
SPOKE-1(config-bgp-neighbor)# allow-local-as 10
SPOKE-1(config-bgp-neighbor)# update-source 203.0.113.130
SPOKE-1(config-bgp-neighbor)# fall-over bfd
SPOKE-1(config-bgp-neighbor)# address-family ipv4 unicast
SPOKE-1(config-bgp-neighbor-af)# route-map DMVPN_BGP_OUT out
SPOKE-1(config-bgp-neighbor-af)# enable
SPOKE-1(config-bgp-neighbor-af)# exit
SPOKE-1(config-bgp-neighbor)# enable
SPOKE-1(config-bgp-neighbor)# exit
SPOKE-1(config-bgp)# address-family ipv4 unicast
SPOKE-1(config-bgp-af)# network 128.66.1.0/24
SPOKE-1(config-bgp-af)# exit
SPOKE-1(config-bgp)# enable
SPOKE-1(config-bgp)# exit
```

Произведём настройку IPsec для SPOKE-1, настроим `ike proposal`, `ike policy` и `ike gateway`. В `ike gateway` дополнительно настроим `dpd` для ускорения перестроения туннелей, в случае если выйдет из строя Active-устройство:

```
SPOKE-1(config)# security ike proposal ike_proposal
SPOKE-1(config-ike-proposal)# authentication algorithm sha2-256
SPOKE-1(config-ike-proposal)# encryption algorithm aes256
SPOKE-1(config-ike-proposal)# dh-group 19
SPOKE-1(config-ike-proposal)# exit
SPOKE-1(config)# security ike policy ike_policy
SPOKE-1(config-ike-policy)# pre-shared-key ascii-text encrypted 8CB5107EA7005AFF
SPOKE-1(config-ike-policy)# proposal ike_proposal
SPOKE-1(config-ike-policy)# exit
SPOKE-1(config)# security ike gateway ike_gateway_cloud_one
SPOKE-1(config-ike-gw)# version v2-only
SPOKE-1(config-ike-gw)# ike-policy ike_policy
SPOKE-1(config-ike-gw)# local address 198.51.100.10
SPOKE-1(config-ike-gw)# local network 198.51.100.10/32 protocol gre
SPOKE-1(config-ike-gw)# remote address 198.51.100.2
SPOKE-1(config-ike-gw)# remote network 198.51.100.2/32 protocol gre
SPOKE-1(config-ike-gw)# mode policy-based
SPOKE-1(config-ike-gw)# mobike disable
SPOKE-1(config-ike-gw)# dead-peer-detection action clear
SPOKE-1(config-ike-gw)# dead-peer-detection interval 10
SPOKE-1(config-ike-gw)# dead-peer-detection retransmit timeout 5
SPOKE-1(config-ike-gw)# dead-peer-detection retransmit tries 2
SPOKE-1(config-ike-gw)# exit
SPOKE-1(config)# security ike gateway ike_gateway_cloud_two
SPOKE-1(config-ike-gw)# version v2-only
SPOKE-1(config-ike-gw)# ike-policy ike_policy
SPOKE-1(config-ike-gw)# local address 198.51.100.10
SPOKE-1(config-ike-gw)# local network 198.51.100.10/32 protocol gre
SPOKE-1(config-ike-gw)# remote address 198.51.100.6
SPOKE-1(config-ike-gw)# remote network 198.51.100.6/32 protocol gre
SPOKE-1(config-ike-gw)# mode policy-based
SPOKE-1(config-ike-gw)# mobike disable
SPOKE-1(config-ike-gw)# dead-peer-detection action clear
SPOKE-1(config-ike-gw)# dead-peer-detection interval 10
SPOKE-1(config-ike-gw)# dead-peer-detection retransmit timeout 5
```



```
SPOKE-1(config-ike-gw)# dead-peer-detection retransmit tries 2
SPOKE-1(config-ike-gw)# exit
SPOKE-1(config)# security ike gateway ike_gateway_to_spokes
SPOKE-1(config-ike-gw)# version v2-only
SPOKE-1(config-ike-gw)# ike-policy ike_policy
SPOKE-1(config-ike-gw)# local address 198.51.100.10
SPOKE-1(config-ike-gw)# local network 198.51.100.10/32 protocol gre
SPOKE-1(config-ike-gw)# remote id any
SPOKE-1(config-ike-gw)# remote address any
SPOKE-1(config-ike-gw)# remote network any protocol gre
SPOKE-1(config-ike-gw)# mode policy-based
SPOKE-1(config-ike-gw)# mobike disable
SPOKE-1(config-ike-gw)# dead-peer-detection action clear
SPOKE-1(config-ike-gw)# dead-peer-detection interval 10
SPOKE-1(config-ike-gw)# dead-peer-detection retransmit timeout 5
SPOKE-1(config-ike-gw)# dead-peer-detection retransmit tries 2
SPOKE-1(config-ike-gw)# exit
```

Затем настроим IPsec proposal, IPsec policy и IPsec vpn туннели через каждый CLOUD:

```
SPOKE-1(config)# security ipsec proposal ipsec_proposal
SPOKE-1(config-ipsec-proposal)# authentication algorithm sha2-256
SPOKE-1(config-ipsec-proposal)# encryption algorithm aes256
SPOKE-1(config-ipsec-proposal)# pfs dh-group 19
SPOKE-1(config-ipsec-proposal)# exit
SPOKE-1(config)# security ipsec policy ipsec_policy
SPOKE-1(config-ipsec-policy)# proposal ipsec_proposal
SPOKE-1(config-ipsec-policy)# exit
SPOKE-1(config)# security ipsec vpn ipsec_dynamic_to_spoke
SPOKE-1(config-ipsec-vpn)# type transport
SPOKE-1(config-ipsec-vpn)# ike establish-tunnel route
SPOKE-1(config-ipsec-vpn)# ike gateway ike_gateway_to_spokes
SPOKE-1(config-ipsec-vpn)# ike ipsec-policy ipsec_policy
SPOKE-1(config-ipsec-vpn)# enable
SPOKE-1(config-ipsec-vpn)# exit
SPOKE-1(config)# security ipsec vpn ipsec_static_cloud_one
SPOKE-1(config-ipsec-vpn)# type transport
SPOKE-1(config-ipsec-vpn)# ike establish-tunnel route
SPOKE-1(config-ipsec-vpn)# ike gateway ike_gateway_cloud_one
SPOKE-1(config-ipsec-vpn)# ike ipsec-policy ipsec_policy
SPOKE-1(config-ipsec-vpn)# enable
SPOKE-1(config-ipsec-vpn)# exit
SPOKE-1(config)# security ipsec vpn ipsec_static_cloud_two
SPOKE-1(config-ipsec-vpn)# type transport
SPOKE-1(config-ipsec-vpn)# ike establish-tunnel route
SPOKE-1(config-ipsec-vpn)# ike gateway ike_gateway_cloud_two
SPOKE-1(config-ipsec-vpn)# ike ipsec-policy ipsec_policy
SPOKE-1(config-ipsec-vpn)# enable
SPOKE-1(config-ipsec-vpn)# exit
```

Скорректируем правила зоны безопасности WAN, разрешим протоколы для GRE over IPsec-туннеля:

```
SPOKE-1(config)# object-group service ISAKMP_PORT
SPOKE-1(config-object-group-service)# port-range 500
SPOKE-1(config-object-group-service)# port-range 4500
```

```
SPOKE-1(config-object-group-service)# exit
SPOKE-1(config)# security zone-pair WAN self
SPOKE-1(config-security-zone-pair)# rule 1
SPOKE-1(config-security-zone-pair-rule)# action permit
SPOKE-1(config-security-zone-pair-rule)# match protocol udp
SPOKE-1(config-security-zone-pair-rule)# match destination-port object-group
ISAKMP_PORT
SPOKE-1(config-security-zone-pair-rule)# enable
SPOKE-1(config-security-zone-pair-rule)# exit
SPOKE-1(config-security-zone-pair)# rule 2
SPOKE-1(config-security-zone-pair-rule)# action permit
SPOKE-1(config-security-zone-pair-rule)# match protocol esp
SPOKE-1(config-security-zone-pair-rule)# enable
SPOKE-1(config-security-zone-pair-rule)# exit
SPOKE-1(config-security-zone-pair)# rule 3
SPOKE-1(config-security-zone-pair-rule)# action permit
SPOKE-1(config-security-zone-pair-rule)# match protocol gre
SPOKE-1(config-security-zone-pair-rule)# enable
SPOKE-1(config-security-zone-pair-rule)# exit
SPOKE-1(config-security-zone-pair)# exit
```

Настроим правила зон безопасности DMVPN_C_ONE и DMVPN_C_TWO, разрешим прохождение трафика для протоколов BGP, BFD, ICMP:

```
SPOKE-1(config)# object-group service BGP
SPOKE-1(config-object-group-service)# port-range 179
SPOKE-1(config-object-group-service)# exit
SPOKE-1(config)# object-group service BFD
SPOKE-1(config-object-group-service)# port-range 3784
SPOKE-1(config-object-group-service)# exit
SPOKE-1(config)# security zone-pair DMVPN_C_ONE self
SPOKE-1(config-security-zone-pair)# rule 1
SPOKE-1(config-security-zone-pair-rule)# action permit
SPOKE-1(config-security-zone-pair-rule)# match protocol icmp
SPOKE-1(config-security-zone-pair-rule)# enable
SPOKE-1(config-security-zone-pair-rule)# exit
SPOKE-1(config-security-zone-pair)# rule 2
SPOKE-1(config-security-zone-pair-rule)# action permit
SPOKE-1(config-security-zone-pair-rule)# match protocol tcp
SPOKE-1(config-security-zone-pair-rule)# match destination-port object-group BGP
SPOKE-1(config-security-zone-pair-rule)# enable
SPOKE-1(config-security-zone-pair-rule)# exit
SPOKE-1(config-security-zone-pair)# rule 3
SPOKE-1(config-security-zone-pair-rule)# action permit
SPOKE-1(config-security-zone-pair-rule)# match protocol udp
SPOKE-1(config-security-zone-pair-rule)# match destination-port object-group BFD
SPOKE-1(config-security-zone-pair-rule)# enable
SPOKE-1(config-security-zone-pair-rule)# exit
SPOKE-1(config-security-zone-pair)# exit
SPOKE-1(config)# security zone-pair DMVPN_C_TWO self
SPOKE-1(config-security-zone-pair)# rule 1
SPOKE-1(config-security-zone-pair-rule)# action permit
SPOKE-1(config-security-zone-pair-rule)# match protocol icmp
SPOKE-1(config-security-zone-pair-rule)# enable
SPOKE-1(config-security-zone-pair-rule)# exit
SPOKE-1(config-security-zone-pair)# rule 2
SPOKE-1(config-security-zone-pair-rule)# action permit
SPOKE-1(config-security-zone-pair-rule)# match protocol tcp
```

```
SPOKE-1(config-security-zone-pair-rule)# match destination-port object-group BGP
SPOKE-1(config-security-zone-pair-rule)# enable
SPOKE-1(config-security-zone-pair-rule)# exit
SPOKE-1(config-security-zone-pair)# rule 3
SPOKE-1(config-security-zone-pair-rule)# action permit
SPOKE-1(config-security-zone-pair-rule)# match protocol udp
SPOKE-1(config-security-zone-pair-rule)# match destination-port object-group BFD
SPOKE-1(config-security-zone-pair-rule)# enable
SPOKE-1(config-security-zone-pair-rule)# exit
SPOKE-1(config-security-zone-pair)# exit
```

Скорректируем правила зоны безопасности LAN, разрешим прохождение трафика между зонами LAN и DMVPN_C_ONE/DMVPN_C_TWO:

```
SPOKE-1(config)# security zone-pair LAN DMVPN_C_ONE
SPOKE-1(config-security-zone-pair)# rule 1
SPOKE-1(config-security-zone-pair-rule)# action permit
SPOKE-1(config-security-zone-pair-rule)# enable
SPOKE-1(config-security-zone-pair-rule)# exit
SPOKE-1(config-security-zone-pair)# exit
SPOKE-1(config)# security zone-pair LAN DMVPN_C_TWO
SPOKE-1(config-security-zone-pair)# rule 1
SPOKE-1(config-security-zone-pair-rule)# action permit
SPOKE-1(config-security-zone-pair-rule)# enable
SPOKE-1(config-security-zone-pair-rule)# exit
SPOKE-1(config-security-zone-pair)# exit
SPOKE-1(config)# security zone-pair DMVPN_C_ONE LAN
SPOKE-1(config-security-zone-pair)# rule 1
SPOKE-1(config-security-zone-pair-rule)# action permit
SPOKE-1(config-security-zone-pair-rule)# match protocol icmp
SPOKE-1(config-security-zone-pair-rule)# enable
SPOKE-1(config-security-zone-pair-rule)# exit
SPOKE-1(config-security-zone-pair)# exit
SPOKE-1(config)# security zone-pair DMVPN_C_TWO LAN
SPOKE-1(config-security-zone-pair)# rule 1
SPOKE-1(config-security-zone-pair-rule)# action permit
SPOKE-1(config-security-zone-pair-rule)# match protocol icmp
SPOKE-1(config-security-zone-pair-rule)# enable
SPOKE-1(config-security-zone-pair-rule)# exit
SPOKE-1(config-security-zone-pair)# exit
```

Конфигурирование SPOKE-2

Создадим туннели mGRE, каждый через свой CLOUD, определим принадлежность к зоне безопасности, настроим NHRP и включим туннель и NHRP командой **enable**:

```
SPOKE-2(config)# security zone DMVPN_C_TWO
SPOKE-2(config-security-zone)# exit
SPOKE-2(config)# security zone DMVPN_C_ONE
SPOKE-2(config-security-zone)# exit
SPOKE-2(config)# tunnel gre 1
SPOKE-2(config-gre)# key 1000
SPOKE-2(config-gre)# ttl 64
SPOKE-2(config-gre)# mtu 1400
SPOKE-2(config-gre)# multipoint
SPOKE-2(config-gre)# security-zone DMVPN_C_ONE
```

```
SPOKE-2(config-gre)# local address 198.51.100.14
SPOKE-2(config-gre)# ip address 203.0.113.3/25
SPOKE-2(config-gre)# ip tcp adjust-mss 1360
SPOKE-2(config-gre)# ip nhrp holding-time 60
SPOKE-2(config-gre)# ip nhrp shortcut
SPOKE-2(config-gre)# ip nhrp map 203.0.113.1 198.51.100.2
SPOKE-2(config-gre)# ip nhrp nhs 203.0.113.1
SPOKE-2(config-gre)# ip nhrp multicast nhs
SPOKE-2(config-gre)# ip nhrp enable
SPOKE-2(config-gre)# enable
SPOKE-2(config-gre)# exit
SPOKE-2(config)# tunnel gre 2
SPOKE-2(config-gre)# key 2000
SPOKE-2(config-gre)# ttl 64
SPOKE-2(config-gre)# mtu 1400
SPOKE-2(config-gre)# multipoint
SPOKE-2(config-gre)# security-zone DMVPN_C_TWO
SPOKE-2(config-gre)# local address 198.51.100.14
SPOKE-2(config-gre)# ip address 203.0.113.131/25
SPOKE-2(config-gre)# ip tcp adjust-mss 1360
SPOKE-2(config-gre)# ip nhrp holding-time 60
SPOKE-2(config-gre)# ip nhrp shortcut
SPOKE-2(config-gre)# ip nhrp map 203.0.113.129 198.51.100.6
SPOKE-2(config-gre)# ip nhrp nhs 203.0.113.129
SPOKE-2(config-gre)# ip nhrp multicast nhs
SPOKE-2(config-gre)# ip nhrp enable
SPOKE-2(config-gre)# enable
SPOKE-2(config-gre)# exit
```

Произведём настройку протокола динамической маршрутизации для SPOKE-1. В примере это будет eBGP, для которого необходимо явно разрешить анонсирование подсетей. Анонсируем LAN-подсети в сторону HUB, используя network в address-family.

Для ускорения переключения в случае выхода из строя Active-устройства в кластере включим также bfd для BGP, а также уменьшим таймер error-wait.

```
SPOKE-2(config)# route-map DMVPN_BGP_OUT
SPOKE-2(config-route-map)# rule 1
SPOKE-2(config-route-map-rule)# exit
SPOKE-2(config-route-map)# exit
SPOKE-2(config)# router bgp 64501
SPOKE-2(config-bgp)# timers error-wait 5 10
SPOKE-2(config-bgp)# neighbor 203.0.113.1
SPOKE-2(config-bgp-neighbor)# remote-as 64500
SPOKE-2(config-bgp-neighbor)# allow-local-as 10
SPOKE-2(config-bgp-neighbor)# update-source 203.0.113.3
SPOKE-2(config-bgp-neighbor)# fall-over bfd
SPOKE-2(config-bgp-neighbor)# address-family ipv4 unicast
SPOKE-2(config-bgp-neighbor-af)# route-map DMVPN_BGP_OUT out
SPOKE-2(config-bgp-neighbor-af)# enable
SPOKE-2(config-bgp-neighbor-af)# exit
SPOKE-2(config-bgp-neighbor)# enable
SPOKE-2(config-bgp-neighbor)# exit
SPOKE-2(config-bgp)# neighbor 203.0.113.129
SPOKE-2(config-bgp-neighbor)# remote-as 64500
SPOKE-2(config-bgp-neighbor)# allow-local-as 10
SPOKE-2(config-bgp-neighbor)# update-source 203.0.113.131
```

```
SPOKE-2(config-bgp-neighbor)# fall-over bfd
SPOKE-2(config-bgp-neighbor)# address-family ipv4 unicast
SPOKE-2(config-bgp-neighbor-af)# route-map DMVPN_BGP_OUT out
SPOKE-2(config-bgp-neighbor-af)# enable
SPOKE-2(config-bgp-neighbor-af)# exit
SPOKE-2(config-bgp-neighbor)# enable
SPOKE-2(config-bgp-neighbor)# exit
SPOKE-2(config-bgp)# address-family ipv4 unicast
SPOKE-2(config-bgp-af)# network 128.66.2.0/24
SPOKE-2(config-bgp-af)# exit
SPOKE-2(config-bgp)# enable
SPOKE-2(config-bgp)# exit
```

Произведём настройку IPsec для SPOKE-1, настроим `ike proposal`, `ike policy` и `ike gateway`. В `ike gateway` дополнительно настроим `dpd`, для ускорения перестроения туннелей, в случае если выйдет из строя Active-устройство:

```
SPOKE-2(config)# security ike proposal ike_proposal
SPOKE-2(config-ike-proposal)# authentication algorithm sha2-256
SPOKE-2(config-ike-proposal)# encryption algorithm aes256
SPOKE-2(config-ike-proposal)# dh-group 19
SPOKE-2(config-ike-proposal)# exit
SPOKE-2(config)# security ike policy ike_policy
SPOKE-2(config-ike-policy)# pre-shared-key ascii-text encrypted 8CB5107EA7005AFF
SPOKE-2(config-ike-policy)# proposal ike_proposal
SPOKE-2(config-ike-policy)# exit
SPOKE-2(config)# security ike gateway ike_gateway_cloud_one
SPOKE-2(config-ike-gw)# version v2-only
SPOKE-2(config-ike-gw)# ike-policy ike_policy
SPOKE-2(config-ike-gw)# local address 198.51.100.14
SPOKE-2(config-ike-gw)# local network 198.51.100.14/32 protocol gre
SPOKE-2(config-ike-gw)# remote address 198.51.100.2
SPOKE-2(config-ike-gw)# remote network 198.51.100.2/32 protocol gre
SPOKE-2(config-ike-gw)# mode policy-based
SPOKE-2(config-ike-gw)# mobike disable
SPOKE-2(config-ike-gw)# dead-peer-detection action clear
SPOKE-2(config-ike-gw)# dead-peer-detection interval 10
SPOKE-2(config-ike-gw)# dead-peer-detection retransmit timeout 5
SPOKE-2(config-ike-gw)# dead-peer-detection retransmit tries 2
SPOKE-2(config-ike-gw)# exit
SPOKE-2(config)# security ike gateway ike_gateway_cloud_two
SPOKE-2(config-ike-gw)# version v2-only
SPOKE-2(config-ike-gw)# ike-policy ike_policy
SPOKE-2(config-ike-gw)# local address 198.51.100.14
SPOKE-2(config-ike-gw)# local network 198.51.100.14/32 protocol gre
SPOKE-2(config-ike-gw)# remote address 198.51.100.6
SPOKE-2(config-ike-gw)# remote network 198.51.100.6/32 protocol gre
SPOKE-2(config-ike-gw)# mode policy-based
SPOKE-2(config-ike-gw)# mobike disable
SPOKE-2(config-ike-gw)# dead-peer-detection action clear
SPOKE-2(config-ike-gw)# dead-peer-detection interval 10
SPOKE-2(config-ike-gw)# dead-peer-detection retransmit timeout 5
SPOKE-2(config-ike-gw)# dead-peer-detection retransmit tries 2
SPOKE-2(config-ike-gw)# exit
SPOKE-2(config)# security ike gateway ike_gateway_to_spokes
SPOKE-2(config-ike-gw)# version v2-only
SPOKE-2(config-ike-gw)# ike-policy ike_policy
```

```
SPOKE-2(config-ike-gw)# local address 198.51.100.14
SPOKE-2(config-ike-gw)# local network 198.51.100.14/32 protocol gre
SPOKE-2(config-ike-gw)# remote id any
SPOKE-2(config-ike-gw)# remote address any
SPOKE-2(config-ike-gw)# remote network any protocol gre
SPOKE-2(config-ike-gw)# mode policy-based
SPOKE-2(config-ike-gw)# mobike disable
SPOKE-2(config-ike-gw)# dead-peer-detection action clear
SPOKE-2(config-ike-gw)# dead-peer-detection interval 10
SPOKE-2(config-ike-gw)# dead-peer-detection retransmit timeout 5
SPOKE-2(config-ike-gw)# dead-peer-detection retransmit tries 2
SPOKE-2(config-ike-gw)# exit
```

Затем настроим IPsec proposal, IPsec policy и IPsec vpn туннели через каждый CLOUD:

```
SPOKE-2(config)# security ipsec proposal ipsec_proposal
SPOKE-2(config-ipsec-proposal)# authentication algorithm sha2-256
SPOKE-2(config-ipsec-proposal)# encryption algorithm aes256
SPOKE-2(config-ipsec-proposal)# pfs dh-group 19
SPOKE-2(config-ipsec-proposal)# exit
SPOKE-2(config)# security ipsec policy ipsec_policy
SPOKE-2(config-ipsec-policy)# proposal ipsec_proposal
SPOKE-2(config-ipsec-policy)# exit
SPOKE-2(config)# security ipsec vpn ipsec_dynamic_to_spoke
SPOKE-2(config-ipsec-vpn)# type transport
SPOKE-2(config-ipsec-vpn)# ike establish-tunnel route
SPOKE-2(config-ipsec-vpn)# ike gateway ike_gateway_to_spokes
SPOKE-2(config-ipsec-vpn)# ike ipsec-policy ipsec_policy
SPOKE-2(config-ipsec-vpn)# enable
SPOKE-2(config-ipsec-vpn)# exit
SPOKE-2(config)# security ipsec vpn ipsec_static_cloud_one
SPOKE-2(config-ipsec-vpn)# type transport
SPOKE-2(config-ipsec-vpn)# ike establish-tunnel route
SPOKE-2(config-ipsec-vpn)# ike gateway ike_gateway_cloud_one
SPOKE-2(config-ipsec-vpn)# ike ipsec-policy ipsec_policy
SPOKE-2(config-ipsec-vpn)# enable
SPOKE-2(config-ipsec-vpn)# exit
SPOKE-2(config)# security ipsec vpn ipsec_static_cloud_two
SPOKE-2(config-ipsec-vpn)# type transport
SPOKE-2(config-ipsec-vpn)# ike establish-tunnel route
SPOKE-2(config-ipsec-vpn)# ike gateway ike_gateway_cloud_two
SPOKE-2(config-ipsec-vpn)# ike ipsec-policy ipsec_policy
SPOKE-2(config-ipsec-vpn)# enable
SPOKE-2(config-ipsec-vpn)# exit
```

Скорректируем правила зоны безопасности WAN, разрешим протоколы для GRE over IPsec-туннеля:

```
SPOKE-2(config)# object-group service ISAKMP_PORT
SPOKE-2(config-object-group-service)# port-range 500
SPOKE-2(config-object-group-service)# port-range 4500
SPOKE-2(config-object-group-service)# exit
SPOKE-2(config)# security zone-pair WAN self
SPOKE-2(config-security-zone-pair)# rule 1
SPOKE-2(config-security-zone-pair-rule)# action permit
SPOKE-2(config-security-zone-pair-rule)# match protocol udp
```

```
SPOKE-2 (config-security-zone-pair-rule) # match destination-port object-group
ISAKMP_PORT
SPOKE-2 (config-security-zone-pair-rule) # enable
SPOKE-2 (config-security-zone-pair-rule) # exit
SPOKE-2 (config-security-zone-pair) # rule 2
SPOKE-2 (config-security-zone-pair-rule) # action permit
SPOKE-2 (config-security-zone-pair-rule) # match protocol esp
SPOKE-2 (config-security-zone-pair-rule) # enable
SPOKE-2 (config-security-zone-pair-rule) # exit
SPOKE-2 (config-security-zone-pair) # rule 3
SPOKE-2 (config-security-zone-pair-rule) # action permit
SPOKE-2 (config-security-zone-pair-rule) # match protocol gre
SPOKE-2 (config-security-zone-pair-rule) # enable
SPOKE-2 (config-security-zone-pair-rule) # exit
SPOKE-2 (config-security-zone-pair) # exit
```

Настроим правила зон безопасности DMVPN_C_ONE и DMVPN_C_TWO, разрешим прохождение трафика для протоколов BGP, BFD, ICMP:

```
SPOKE-2 (config) # object-group service BGP
SPOKE-2 (config-object-group-service) # port-range 179
SPOKE-2 (config-object-group-service) # exit
SPOKE-2 (config) # object-group service BFD
SPOKE-2 (config-object-group-service) # port-range 3784
SPOKE-2 (config-object-group-service) # exit
SPOKE-2 (config) # security zone-pair DMVPN_C_ONE self
SPOKE-2 (config-security-zone-pair) # rule 1
SPOKE-2 (config-security-zone-pair-rule) # action permit
SPOKE-2 (config-security-zone-pair-rule) # match protocol icmp
SPOKE-2 (config-security-zone-pair-rule) # enable
SPOKE-2 (config-security-zone-pair-rule) # exit
SPOKE-2 (config-security-zone-pair) # rule 2
SPOKE-2 (config-security-zone-pair-rule) # action permit
SPOKE-2 (config-security-zone-pair-rule) # match protocol tcp
SPOKE-2 (config-security-zone-pair-rule) # match destination-port object-group BGP
SPOKE-2 (config-security-zone-pair-rule) # enable
SPOKE-2 (config-security-zone-pair-rule) # exit
SPOKE-2 (config-security-zone-pair) # rule 3
SPOKE-2 (config-security-zone-pair-rule) # action permit
SPOKE-2 (config-security-zone-pair-rule) # match protocol udp
SPOKE-2 (config-security-zone-pair-rule) # match destination-port object-group BFD
SPOKE-2 (config-security-zone-pair-rule) # enable
SPOKE-2 (config-security-zone-pair-rule) # exit
SPOKE-2 (config-security-zone-pair) # exit
SPOKE-2 (config) # security zone-pair DMVPN_C_TWO self
SPOKE-2 (config-security-zone-pair) # rule 1
SPOKE-2 (config-security-zone-pair-rule) # action permit
SPOKE-2 (config-security-zone-pair-rule) # match protocol icmp
SPOKE-2 (config-security-zone-pair-rule) # enable
SPOKE-2 (config-security-zone-pair-rule) # exit
SPOKE-2 (config-security-zone-pair) # rule 2
SPOKE-2 (config-security-zone-pair-rule) # action permit
SPOKE-2 (config-security-zone-pair-rule) # match protocol tcp
SPOKE-2 (config-security-zone-pair-rule) # match destination-port object-group BGP
SPOKE-2 (config-security-zone-pair-rule) # enable
SPOKE-2 (config-security-zone-pair-rule) # exit
SPOKE-2 (config-security-zone-pair) # rule 3
```



```
SPOKE-2 (config-security-zone-pair-rule) # action permit
SPOKE-2 (config-security-zone-pair-rule) # match protocol udp
SPOKE-2 (config-security-zone-pair-rule) # match destination-port object-group BFD
SPOKE-2 (config-security-zone-pair-rule) # enable
SPOKE-2 (config-security-zone-pair-rule) # exit
SPOKE-2 (config-security-zone-pair) # exit
```

Скорректируем правила зоны безопасности LAN, разрешим прохождение трафика между зонами LAN и DMVPN_C_ONE/DMVPN_C_TWO:

```
SPOKE-2 (config) # security zone-pair LAN DMVPN_C_ONE
SPOKE-2 (config-security-zone-pair) # rule 1
SPOKE-2 (config-security-zone-pair-rule) # action permit
SPOKE-2 (config-security-zone-pair-rule) # enable
SPOKE-2 (config-security-zone-pair-rule) # exit
SPOKE-2 (config-security-zone-pair) # exit
SPOKE-2 (config) # security zone-pair LAN DMVPN_C_TWO
SPOKE-2 (config-security-zone-pair) # rule 1
SPOKE-2 (config-security-zone-pair-rule) # action permit
SPOKE-2 (config-security-zone-pair-rule) # enable
SPOKE-2 (config-security-zone-pair-rule) # exit
SPOKE-2 (config-security-zone-pair) # exit
SPOKE-2 (config) # security zone-pair DMVPN_C_ONE LAN
SPOKE-2 (config-security-zone-pair) # rule 1
SPOKE-2 (config-security-zone-pair-rule) # action permit
SPOKE-2 (config-security-zone-pair-rule) # match protocol icmp
SPOKE-2 (config-security-zone-pair-rule) # enable
SPOKE-2 (config-security-zone-pair-rule) # exit
SPOKE-2 (config-security-zone-pair) # exit
SPOKE-2 (config) # security zone-pair DMVPN_C_TWO LAN
SPOKE-2 (config-security-zone-pair) # rule 1
SPOKE-2 (config-security-zone-pair-rule) # action permit
SPOKE-2 (config-security-zone-pair-rule) # match protocol icmp
SPOKE-2 (config-security-zone-pair-rule) # enable
SPOKE-2 (config-security-zone-pair-rule) # exit
SPOKE-2 (config-security-zone-pair) # exit
```

Проверка работы:

Состояние IPsec-туннелей можно посмотреть командой:

HUB-1# show security ipsec vpn status

Name	Local host	Remote host	Initiator spi	Responder spi	State
ipsec_dynamic_cloud_one	198.51.100.2	198.51.100.14	0x22d11891e06edf92	0x40469d552e93e47c	Established
ipsec_dynamic_cloud_two	198.51.100.6	198.51.100.14	0x61f7a205eeef5d06	0x141239e7309d351c	Established
ipsec_dynamic_cloud_one	198.51.100.2	198.51.100.10	0x3dbf984518584d5e	0x08563e2683776071	Established
ipsec_dynamic_cloud_two	198.51.100.6	198.51.100.10	0x500adbe8428c7d35	0x9c83c7a2255cb0ed	Established

SPOKE-1# show security ipsec vpn status

Name	Local host	Remote host	Initiator spi	Responder spi	State
ipsec_static_cloud_one	198.51.100.10	198.51.100.2	0x3dbf984518584d5e	0x08563e2683776071	Established
ipsec_static_cloud_two	198.51.100.10	198.51.100.6	0x500adbe8428c7d35	0x9c83c7a2255cb0ed	Established

SPOKE-2# show security ipsec vpn status

Name	Local host	Remote host	Initiator spi	Responder spi	State
ipsec_static_cloud_one	198.51.100.14	198.51.100.2	0x22d11891e06edf92	0x40469d552e93e47c	Established
ipsec_static_cloud_two	198.51.100.14	198.51.100.6	0x61f7a205eeef5d06	0x141239e7309d351c	Established

Состояние NHRP-записей можно посмотреть командой:

HUB-1# show ip nhrp peers

Flags: E - unique, R - nhs, U - used, L - lower-up
 C - connected, G - group, Q - qos, N - nat
 P - protected, I - Redirect-ignored, X - undefined

Tunnel address	NBMA address	Tunnel	Expire (h:m:s)	Created (d,h:m:s)	Type	Flags
203.0.113.2	198.51.100.10	gre 1	00:00:51	00,00:04:41	dynamic	LCP
203.0.113.3	198.51.100.14	gre 1	00:00:48	00,00:04:44	dynamic	LCP
203.0.113.130	198.51.100.10	gre 2	00:00:51	00,00:04:41	dynamic	LCP
203.0.113.131	198.51.100.14	gre 2	00:00:48	00,00:04:44	dynamic	LCP

SPOKE-1# show ip nhrp peers

Flags: E - unique, R - nhs, U - used, L - lower-up
 C - connected, G - group, Q - qos, N - nat
 P - protected, I - Redirect-ignored, X - undefined

Tunnel address	NBMA address	Tunnel	Expire (h:m:s)	Created (d,h:m:s)	Type	Flags
203.0.113.1	198.51.100.2	gre 1	--	00,00:00:13	static	RULCP
203.0.113.129	198.51.100.6	gre 2	--	00,00:00:13	static	RULCP

SPOKE-2# show ip nhrp peers

Flags: E - unique, R - nhs, U - used, L - lower-up
 C - connected, G - group, Q - qos, N - nat
 P - protected, I - Redirect-ignored, X - undefined

Tunnel address	NBMA address	Tunnel	Expire (h:m:s)	Created (d,h:m:s)	Type	Flags
203.0.113.1	198.51.100.2	gre 1	--	00,00:00:16	static	RULCP
203.0.113.129	198.51.100.6	gre 2	--	00,00:00:16	static	RULCP

18. УПРАВЛЕНИЕ УДАЛЕННЫМ ДОСТУПОМ

18.1. Настройка сервера удаленного доступа к корпоративной сети по PPTP-протоколу

PPTP (англ. Point-to-Point Tunneling Protocol) – туннельный протокол типа точка-точка, позволяющий компьютеру устанавливать защищённое соединение с сервером за счёт создания специального туннеля в обычной незащищенной сети. PPTP помещает (инкапсулирует) кадры PPP в IP-пакеты для передачи по глобальной IP-сети, например, Интернет. PPTP может также использоваться для организации туннеля между двумя локальными сетями. PPTP использует дополнительное TCP-соединение для обслуживания туннеля.

18.1.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать профиль PPTP-сервера.	<code>rtt(config)# remote-access pptp <NAME></code>	<NAME> – имя профиля PPTP-сервера, задаётся строкой до 31 символа.
2	Указать описание конфигурируемого сервера (необязательно).	<code>rtt(config-pptp-server)# description <DESCRIPTION></code>	<DESCRIPTION> – описание PPTP-сервера, задаётся строкой до 255 символов.
3	Указать IP-адрес, который должен обрабатывать PPTP-сервер.	<code>rtt(config-pptp-server)# outside-address { object-group <OBJ-GROUP-NETWORK-NAME> ip-address <ADDR> interface { <IF> <TUN> } }</code>	<OBJ-GROUP-NETWORK-NAME> – имя профиля, содержащего IP-адрес, который должен слушать PPTP-сервер, задаётся строкой до 31 символа; <ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]; <IF> – тип и идентификатор интерфейса маршрутизатора; <TUN> – тип и номер туннеля маршрутизатора.
4	Указать IP-адрес локального шлюза.	<code>rtt(config-pptp-server)# local-address { object-group <OBJ-GROUP-NETWORK-NAME> ip-address <ADDR> }</code>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который содержит IP-адрес локального шлюза, задаётся строкой до 31 символа; <ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Шаг	Описание	Команда	Ключи
5	Указать список IP-адресов, из которого PPTP выдаются динамические IP-адреса удаленным пользователям.	<pre> rtt(config-pptp-server)# remote-address { object-group <OBJ-GROUP-NETWORK-NAME> address-range <FROM-ADDR>-<TO-ADDR> } </pre>	<p><OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который содержит список IP-адресов удаленных пользователей, задаётся строкой до 31 символа;</p> <p><FROM-ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p>
6	Выбрать режим аутентификации PPTP-клиентов.	<pre> rtt(config-pptp-server)# authentication mode { local radius } </pre>	<ul style="list-style-type: none"> local – аутентификация пользователя по локальной базе. radius – аутентификация пользователя по базе RADIUS-сервера. На маршрутизаторе должен быть сконфигурирован механизм взаимодействия с RADIUS-сервером см. раздел Алгоритм настройки AAA по протоколу RADIUS.
7	Разрешить необходимые методы аутентификации удаленных пользователей.	<pre> rtt(config-pptp-server)# authentication method <METHOD> </pre>	<p><METHOD> – метод аутентификации, принимает значения [chap, mschap, mschap-v2, eap, pap].</p> <p>По умолчанию разрешен только chap.</p>
8	Указать имя пользователя (при использовании локальной аутентификации пользователей).	<pre> rtt(config-pptp-server) username <NAME> </pre>	<p><NAME> – имя пользователя, задаётся строкой до 12 символов.</p>
9	Указать пароль пользователя (при использовании локальной аутентификации пользователей).	<pre> rtt(config-pptp-user) password ascii-text { <PASSWORD> encrypted <PASSWORD> } </pre>	<p><PASSWORD> – пароль пользователя, задается строкой до 32 символов.</p>
10	Активировать пользователя (при использовании локальной аутентификации пользователей).	<pre> rtt(config-pptp-user) enable </pre>	
11	Включить PPTP-сервер в зону безопасности и настроить правила взаимодействия между зонами или отключить firewall (см. раздел Конфигурирование Firewall).	<pre> rtt(config-pptp-server)# security-zone <NAME> </pre>	<p><NAME> – имя зоны безопасности, задаётся строкой до 31 символа.</p>

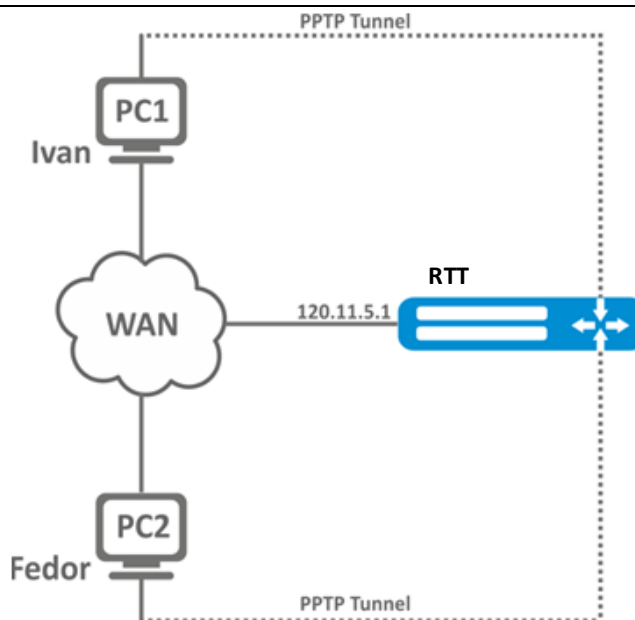
Шаг	Описание	Команда	Ключи
12	Включить сервер.	<code>rtt(config-pptp-server)# enable</code>	
13	Указать DSCP-приоритет исходящих пакетов (необязательно).	<code>rtt(config-pptp-server)# dscp <DSCP></code>	<DSCP>— dscp-приоритет исходящих пакетов [0..63].
14	Включить шифрование MPPE для PPTP-соединений (необязательно).	<code>rtt(config-pptp-server)# encryption mppe</code>	
15	Указать размер MTU (MaximumTransmissionUnit) для сервера (не обязательно). MTU более 1500 будет активно только в случае применения команды "system jumbo-frames".	<code>rtt(config-pptp-server) mtu <MTU></code>	<MTU> – значение MTU, принимает значения в диапазоне [1280..1500]. Значение по умолчанию: 1500.
16	Указать список DNS-серверов, которые будут использовать удаленные пользователи (необязательно).	<code>rtt(config-pptp-server)# dns-servers object-group <OBJ-GROUP-NETWORK -NAME ></code>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, содержащего адреса необходимых DNS-серверов, задаётся строкой до 31 символа.
17	Указать список WINS-серверов, которые будут использовать удаленные пользователи (необязательно).	<code>rtt(config-pptp-server)# wins-servers object-group <OBJ-GROUP-NETWORK-NAME ></code>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, содержащего адреса необходимых WINS-серверов, задаётся строкой до 31 символа.

18.1.2. Пример настройки

Задача:

Настроить PPTP-сервер на маршрутизаторе.

- адрес PPTP-сервера – 120.11.5.1;
- шлюз внутри туннеля для подключающихся клиентов – 10.10.10.1;
- пул IP-адресов для выдачи 10.10.10.5-10.10.10.25;
- DNS-серверы: 8.8.8.8, 8.8.8.4;
- учетные записи для подключения – fedor, ivan.



Решение:

Создадим профиль адресов, содержащий адрес, который должен слушать сервер:

```

rtt# configure
rtt(config)# object-group network pptp_outside
rtt(config-object-group-network)# ip address-range 120.11.5.1
rtt(config-object-group-network)# exit
  
```

Создадим профиль адресов, содержащий адрес локального шлюза:

```

rtt(config)# object-group network pptp_local
rtt(config-object-group-network)# ip address-range 10.10.10.1
rtt(config-object-group-network)# exit
  
```

Создадим профиль адресов, содержащий адреса клиентов:

```

rtt(config)# object-group network pptp_remote
rtt(config-object-group-network)# ip address-range 10.10.10.5-10.10.10.25
rtt(config-object-group-network)# exit
  
```

Создадим профиль адресов, содержащий адреса, которые будут использовать удаленные пользователи:

```

rtt(config)# object-group network pptp_dns
rtt(config-object-group-network)# ip address-range 8.8.8.8,8.8.8.4
rtt(config-object-group-network)# exit
  
```

Создадим PPTP-сервер и привяжем вышеуказанные профили:

```

rtt(config)# remote-access pptp remote-workers
rtt(config-pptp)# local-address object-group pptp_local
rtt(config-pptp)# remote-address object-group pptp_remote
  
```

```
rtt(config-pptp)# outside-address object-group pptp_outside
rtt(config-pptp)# dns-servers object-group pptp_dns
```

Выберем метод аутентификации пользователей PPTP-сервера:

```
rtt(config-pptp)# authentication mode local
```

Укажем зону безопасности, к которой будут относиться сессии пользователей:

```
rtt(config-pptp)# security-zone VPN
```

Создадим PPTP-пользователей *Ivan* и *Fedor* для PPTP-сервера:

```
rtt(config-pptp)# username ivan
rtt(config-pptp-user)# password ascii-text password1
rtt(config-pptp-user)# enable
rtt(config-pptp-user)# exit
rtt(config-pptp)# username fedor
rtt(config-pptp-user)# password ascii-text password2
rtt(config-pptp-user)# enable
rtt(config-pptp-user)# exit
rtt(config-pptp)# exit
```

Включим PPTP-сервер:

```
rtt(config-pptp)# enable
```

После применения конфигурации маршрутизатор будет прослушивать 120.11.5.1:1723. Состояние сессий PPTP-сервера можно посмотреть командой:

```
rtt# show remote-access status pptp server remote-workers
```

Счетчики сессий PPTP-сервера можно посмотреть командой:

```
rtt# show remote-access counters pptp server remote-workers
```

Очистить счетчики сессий PPTP-сервера можно командой:

```
rtt# clear remote-access counters pptp server remote-workers
```

Завершить сессию пользователя *fedor* PPTP-сервера можно одной из следующих команд:

```
rtt# clear remote-access session pptp username fedor
rtt# clear remote-access session pptp server remote-workers username fedor
```

Конфигурацию PPTP-сервера можно посмотреть командой:

```
rtt# show remote-access configuration pptp remote-workers
```



Помимо создания PPTP-сервера необходимо в firewall открыть TCP-порт 1723 для обслуживания соединений и разрешить протокол GRE(47) для туннельного трафика.

18.2. Настройка сервера удаленного доступа к корпоративной сети по L2TP over IPsec протоколу

L2TP (англ. Layer 2 Tunneling Protocol – протокол туннелирования второго уровня) – туннельный протокол, использующийся для поддержки виртуальных частных сетей. L2TP помещает (инкапсулирует) кадры PPP в IP-пакеты для передачи по глобальной IP-сети, например, Интернет. L2TP может также использоваться для организации туннеля между двумя локальными сетями. L2TP использует дополнительное UDP-соединение для обслуживания туннеля. L2TP-протокол не предоставляет средств шифрования данных и поэтому он обычно используется в связке с группой протоколов IPsec, которая предоставляет безопасность на пакетном уровне.

18.2.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать профиль L2TP-сервера.	<code>rtt(config)# remote-access l2tp <NAME></code>	<NAME> – имя профиля L2TP-сервера, задаётся строкой до 31 символа.
2	Указать описание конфигурируемого сервера (необязательно).	<code>rtt(config-l2tp-server)# description <DESCRIPTION></code>	<DESCRIPTION> – описание L2TP-сервера, задаётся строкой до 255 символов.
3	Указать IP-адрес, который должен слушать L2TP-сервер.	<code>rtt(config-l2tp-server)# outside-address { object-group <NAME> ip-address <ADDR> interface { <IF> <TUN> } }</code>	<p><OBJ-GROUP-NETWORK-NAME> – имя профиля, содержащего IP-адрес, который должен слушать L2TP-сервер, задаётся строкой до 31 символа;</p> <p><ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><IF> – тип и идентификатор интерфейса маршрутизатора;</p> <p><TUN> – тип и номер туннеля маршрутизатора.</p>
4	Указать IP-адрес локального шлюза либо отключить firewall для PPTP-сервера.	<code>rtt(config-l2tp-server)# local-address { object-group <OBJ-GROUP-NETWORK-NAME> ip-address <ADDR> }</code>	<p><OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который содержит IP-адрес локального шлюза, задаётся строкой до 31 символа;</p> <p><ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p>

Шаг	Описание	Команда	Ключи
5	Указать список IP-адресов из которого L2TP выдаются динамические IP-адреса удаленным пользователям.	<pre> rtt(config-l2tp-server)# remote-address { object-group <OBJ-GROUP-NETWORK -NAME> address-range <FROM-ADDR>-<TO-ADDR> } </pre>	<p><OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который содержит список IP-адресов удаленных пользователей, задаётся строкой до 31 символа;</p> <p><FROM-ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p>
6	Выбрать режим аутентификации L2TP-клиентов.	<pre> rtt(config-l2tp-server)# authentication mode { local radius } </pre>	<ul style="list-style-type: none"> • local – аутентификация пользователя по локальной базе. • radius – аутентификация пользователя по базе RADIUS-сервера. На маршрутизаторе должен быть сконфигурирован механизм взаимодействия с RADIUS-сервером см. раздел Алгоритм настройки AAA по протоколу RADIUS.
7	Разрешить необходимые методы аутентификации удаленных пользователей.	<pre> rtt(config-l2tp-server)# authentication method <METHOD> </pre>	<p><METHOD> – метод аутентификации, принимает значения [chap, mschap, mschap-v2, eap, pap].</p> <p>По умолчанию разрешен только chap.</p>
8	Включить L2TP-сервер в зону безопасности и настроить правила взаимодействия между зонами (см. раздел Конфигурирование Firewall).	<pre> rtt(config-l2tp-server)# security-zone <NAME> </pre>	<p><NAME> – имя зоны безопасности, задаётся строкой до 31 символа.</p>
9	Указать имя пользователя (при использовании локальной базы аутентификации).	<pre> rtt(config-l2tp-server) username <NAME> </pre>	<p><NAME> – имя пользователя, задаётся строкой до 12 символов.</p>
10	Указать пароль пользователя (при использовании локальной базы аутентификации).	<pre> rtt(config-l2tp-user) password ascii-text { <PASSWORD> encrypted <PASSWORD> } </pre>	<p><PASSWORD> – пароль пользователя, задается строкой до 32 символов.</p>
11	Включить пользователя (при использовании локальной базы аутентификации).	<pre> rtt(config-l2tp-user) enable </pre>	

Шаг	Описание	Команда	Ключи
12	Выбрать метод аутентификации по ключу для IKE-соединения (по умолчанию).	<code>rtt(config-l2tp-server)# ipsec authentication method pre-shared-key</code>	
13	Указать общий секретный ключ для аутентификации, который должен совпадать у обоих сторон, устанавливающих туннель.	<code>rtt(config-l2tp-server)# ipsec authentication pre-shared-key { ascii-text { <TEXT> encrypted <ENCRYPTED-TEXT> } hexadecimal { <HEX> encrypted <ENCRYPTED-HEX> } }</code>	<p><TEXT> – строка [1..64] ASCII-символов;</p> <p><HEX> – число размером [1..32] байт задаётся строкой [2..128] символов в шестнадцатеричном формате (0xYYYY...) или (YYYY...).</p> <p><ENCRYPTED-TEXT> – зашифрованный пароль размером [1..32] байт, задаётся строкой [2..128] символов;</p> <p><ENCRYPTED-HEX> – зашифрованное число размером [2..64] байт, задаётся строкой [2..256] символов.</p>
14	Ограничить используемые методы аутентификации и шифрования протокола ike (необязательно).	<code>rtt(config-l2tp-server)# ipsec ike proposal <NAME></code>	<NAME> – имя ранее созданного профиля протокола IKE, задаётся строкой до 31 символа.
15	Ограничить используемые методы аутентификации и шифрования протокола ipsec (необязательно).	<code>rtt(config-l2tp-server)# ipsec proposal <NAME></code>	<NAME> – имя ранее созданного профиля IPsec, задаётся строкой до 31 символа.
16	Включить сервер.	<code>rtt(config-l2tp-server)# enable</code>	
17	Указать DSCP-приоритет исходящих пакетов.	<code>rtt(config-l2tp-server)# dscp <DSCP></code>	<DSCP> – DSCP-приоритет исходящих пакетов [0..63].
18	Указать размер MTU (MaximumTransmissionUnit) для сервера (необязательно). MTU более 1500 будет активно только в случае применения команды "system jumbo-frames".	<code>rtt(config-l2tp-server) mtu <MTU></code>	<p><MTU> – значение MTU, принимает значения в диапазоне [1280..1500].</p> <p>Значение по умолчанию: 1500.</p>
19	Указать список DNS-серверов, которые будут использовать удаленные пользователи (необязательно).	<code>rtt(config-l2tp-server)# dns-servers object-group <OBJ-GROUP-NETWORK -NAME ></code>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, содержащего, который содержит адреса необходимых DNS-серверов, задаётся строкой до 31 символа.
20	Указать список WINS-серверов, которые будут использовать удаленные пользователи (необязательно).	<code>rtt(config-l2tp-server)# wins-servers object-group <OBJ-GROUP-NETWORK -NAME ></code>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который содержит адреса необходимых WINS-серверов, задаётся строкой до 31 символа.

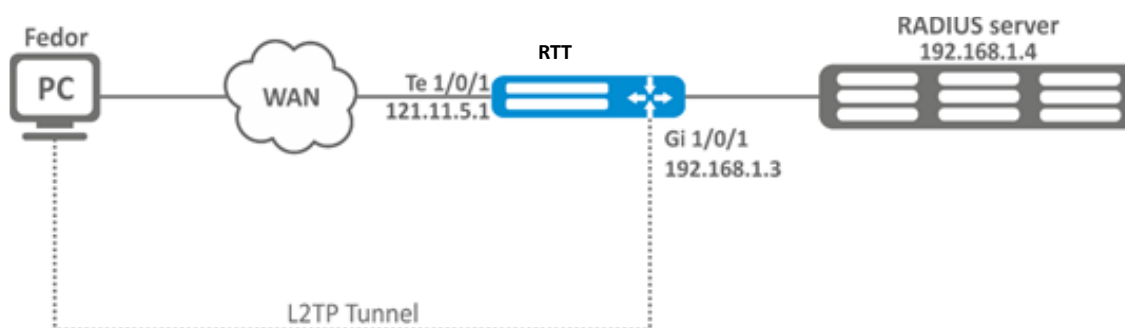
18.2.2. Пример настройки

Задача:

Настроить L2TP-сервер на маршрутизаторе для подключения удаленных пользователей к ЛВС. Аутентификация пользователей проходит на RADIUS-сервере.

- адрес L2TP-сервера – 120.11.5.1;
- шлюз внутри туннеля – 10.10.10.1;
- адрес RADIUS-сервера – 192.168.1.4.

Для IPsec используется метод аутентификации по ключу: ключ — «password».



Решение:

Предварительно нужно выполнить следующие действия:

- Настроить подключение к RADIUS-серверу;
- Настроить зоны для интерфейсов te1/0/1 и gi1/0/1;
- Указать IP-адреса для интерфейсов te1/0/1 и te1/0/1.

Создадим профиль адресов, содержащий адрес локального шлюза:

```
rtt(config)# object-group network l2tp_local
rtt(config-object-group-network)# ip address-range 10.10.10.1
rtt(config-object-group-network)# exit
```

Создадим профиль адресов, содержащий DNS-серверы:

```
rtt(config)# object-group network pptp_dns
rtt(config-object-group-network)# ip address-range 8.8.8.8
rtt(config-object-group-network)# ip address-range 8.8.4.4
rtt(config-object-group-network)# exit
```

Создадим L2TP-сервер и привяжем к нему вышеуказанные профили:

```
rtt(config)# remote-access l2tp remote-workers
rtt(config-l2tp)# local-address ip-address 10.10.10.1
rtt(config-l2tp)# outside-address ip-address 120.11.5.1
```

```
rtt(config-l2tp)# dns-server object-group l2tp_dns
```

Выберем метод аутентификации пользователей L2TP-сервера:

```
rtt(config-l2tp)# authentication mode radius
```

Укажем зону безопасности, к которой будут относиться сессии пользователей:

```
rtt(config-l2tp)# security-zone VPN
```

Выберем метод аутентификации первой фазы IKE и зададим ключ аутентификации:

```
rtt(config-l2tp)# ipsec authentication method psk
rtt(config-l2tp)# ipsec authentication pre-shared-key ascii-text password
```

Включим L2TP-сервер:

```
rtt(config-l2tp)# enable
```

После применения конфигурации маршрутизатор будет прослушивать IP-адрес 120.11.5.1 и порт 1701. Состояние сессий L2TP-сервера можно посмотреть командой:

```
rtt# show remote-access status l2tp server remote-workers
```

Счетчики сессий L2TP-сервера можно посмотреть командой:

```
rtt# show remote-access counters l2tp server remote-workers
```

Очистить счетчики сессий L2TP-сервера можно командой:

```
rtt# clear remote-access counters l2tp server remote-workers
```

Завершить сессию пользователя fedor L2TP-сервера можно одной из следующих команд:

```
rtt# clear remote-access session l2tp username fedor
rtt# clear remote-access session l2tp server remote-workers username fedor
```

Конфигурацию L2TP-сервера можно посмотреть командой:

```
rtt# show remote-access configuration l2tp remote-workers
```

Помимо создания L2TP-сервера необходимо в firewall открыть UDP-порты 500, 1701, 4500 для обслуживания соединений и разрешить протоколы ESP (50) и GRE (47) для туннельного трафика.

18.3. Настройка сервера удаленного доступа к корпоративной сети по OpenVPN-протоколу

OpenVPN — полнофункциональное средство для построения виртуальных частных сетей (Virtual Private Networks, VPN), организации удалённого доступа и решения ряда других задач, связанных с безопасностью передачи данных, базирующееся на SSL.

18.3.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать профиль OpenVPN-сервера.	<code>rtt(config)# remote-access openvpn <NAME></code>	<NAME> – имя профиля OpenVPN-сервера, задаётся строкой до 31 символа.
2	Указать описание конфигурируемого сервера (необязательно).	<code>rtt(config-openvpn-server)# description <DESCRIPTION></code>	<DESCRIPTION> – описание OpenVPN-сервера, задаётся строкой до 255 символов.
3	Указать подсеть, из которой выдаются IP-адреса пользователям (только для tunnel ip).	<code>rtt(config-openvpn-server)# network <ADDR/LEN></code>	<ADDR/LEN> – адрес подсети, имеет следующий формат: AAA.BBB.CCC.DDD/EE – IP-адрес подсети с маской в форме префикса, где AAA-DDD принимают значения [0..255] и EE принимает значения [16..29].
4	Указать инкапсулируемый протокол.	<code>rtt(config-openvpn-server)# protocol <PROTOCOL></code>	<PROTOCOL> – тип инкапсуляции, возможные значения: <ul style="list-style-type: none"> • TCP-инкапсуляция в TCP-сегменты; • UDP-инкапсуляция в UDP-дейтаграммы.
5	Определить тип соединения с частной сетью через OpenVPN-сервер.	<code>rtt(config-openvpn-server)# tunnel <TYPE></code>	<TYPE> – инкапсулирующий протокол, принимает значения: <ul style="list-style-type: none"> • ip – соединение точка-точка; • ethernet – подключение к L2-домену.

Шаг	Описание	Команда	Ключи
6	Указать список IP-адресов, из которого OpenVPN-сервером выдаются динамические IP-адреса удаленным пользователям в режиме L2 (только для tunnel ethernet).	<code>rtt(config-openvpn-server)# address-range <FROM-ADDR>-<TO-ADDR></code>	<p><FROM-ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p>
7	Включить клиентские соединения по OpenVPN в L2-домен (только для tunnel ethernet).	<code>rtt(config-openvpn-server)# bridge-group <BRIDGE-ID></code>	<BRIDGE-ID> – идентификационный номер моста.
8	Указать сертификаты и ключи.	<code>rtt(config-openvpn-server)# crypto <CERTIFICATE-TYPE> <NAME></code>	<p><CERTIFICATE-TYPE> – тип сертификата или ключа, может принимать следующие значения:</p> <ul style="list-style-type: none"> • са – сертификат удостоверяющего сервера; • crl – список отозванных сертификатов; • dh – ключ Диффи-Хеллмана; • cert – публичный сертификат сервера; • private-key – приватный ключ сервера; • ta – HMAC-ключ. <p><NAME> – имя сертификата или ключа, задаётся строкой до 31 символа.</p>
9	<p>Указать контейнер PKCS12 (необязательно.)</p> <p>Примечание: контейнер обязательно должен включать в себя сертификат удостоверяющего центра, публичный сертификат сервера и приватный ключ сервера.</p> <p>Применение сертификатов или контейнера является взаимоисключающим, т. е. необходимо указывать или сертификаты, или контейнер.</p>	<code>rtt(config-openvpn-server)# crypto pfx <NAME> [password ascii-text <PASSWORD>]</code>	<p><NAME> – имя PKCS12-контейнера, задаётся строкой до 31 символа.</p> <p><PASSWORD> – пароль от PKCS12-контейнера.</p>

Шаг	Описание	Команда	Ключи
10	Выбрать алгоритм шифрования, используемый при передаче данных.	<code>rtt(config-openvpn-server)# encryption algorithm <ALGORITHM></code>	<ALGORITHM> – идентификатор протокола шифрования, принимает значения: 3des,blowfish128, aes128.
11	Включить OpenVPN-сервер в зону безопасности и настроить правила взаимодействия между зонами (см. раздел Конфигурирование Firewall).	<code>rtt(config-openvpn-server)# security-zone <NAME></code>	<NAME> – имя зоны безопасности, задаётся строкой до 31 символа.
12	Определить дополнительные параметры для указанного пользователя OpenVPN-сервера (при использовании локальной базы для аутентификации пользователей).	<code>rtt(config-openvpn-server)# username < NAME ></code>	<NAME> – имя пользователя, задаётся строкой до 31 символа.
13	Определить подсеть для указанного пользователя OpenVPN-сервера.	<code>rtt(config-openvpn-user)# subnet <ADDR/LEN></code>	<ADDR/LEN> – адрес подсети, имеет следующий формат: AAA.BBB.CCC.DDD/EE – IP-адрес подсети с маской в форме префикса, где AAA-DDD принимают значения [0..255] и EE принимает значения [16..32].
14	Определить статический IP-адрес для указанного пользователя OpenVPN-сервера.	<code>rtt(config-openvpn-user)# ip address <ADDR></code>	<ADDR> – адрес имеет следующий формат: AAA.BBB.CCC.DDD – IP-адрес подсети, где AAA-DDD принимают значения [0..255].
15	Включить профиль OpenVPN-сервера.	<code>rtt(config-openvpn-server)# enable</code>	
16	Включить блокировку передачи данных между клиентами (необязательно).	<code>rtt(config-openvpn-server)# client-isolation</code>	
17	Устанавливается максимальное количество одновременных пользовательских сессий (необязательно).	<code>rtt(config-openvpn-server)# client-max <VALUE></code>	<VALUE> – максимальное количество пользователей, принимает значения [1..65535].
18	Включается механизм сжатия передаваемых данных между клиентами и сервером OpenVPN (необязательно).	<code>rtt(config-openvpn-server)# compression</code>	
19	Указать список DNS-серверов, которые будут использовать удаленные пользователи (необязательно).	<code>rtt(config-openvpn-server)# dns-server <ADDR></code>	<ADDR> – IP-адрес DNS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

Шаг	Описание	Команда	Ключи
20	Указать TCP-/UDP-порт, который будет прослушиваться OpenVPN-сервером (необязательно).	rtt(config-openvpn-server)# port <PORT>	<PORT> – TCP-/UDP-порт, принимает значения [1..65535]. Значение по умолчанию: 1194.
21	Включить анонсирование маршрута по умолчанию для OpenVPN-соединений, что приводит к замене маршрута по умолчанию на клиентской стороне (необязательно).	rtt(config-openvpn-server)# redirect-gateway	
22	Включить анонсирование указанных подсетей, шлюзом является IP-адрес OpenVPN-сервера (необязательно).	rtt(config-openvpn-server)# route <ADDR/LEN>	<ADDR/LEN> – адрес подсети, имеет следующий формат: 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].
23	Указать временной интервал, по истечению которого встречная сторона считается недоступной (необязательно).	rtt(config-openvpn-server)# timers holdtime <TIME>	<TIME> – время в секундах, принимает значения [2..65535]. Значение по умолчанию: 120.
24	Указать временной интервал, по истечению которого идет проверка соединения со встречной стороной (необязательно).	rtt(config-openvpn-server)# timers keepalive <TIME>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 10.
25	Разрешить подключаться к OpenVPN-серверу нескольким пользователям с одним сертификатом.	rtt(config-openvpn-server)# duplicate-cn	
26	Указать список WINS-серверов, которые будут использовать удаленные пользователи (необязательно).	rtt(config-openvpn-server)# wins-server <ADDR>	<ADDR> – IP-адрес WINS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Шаг	Описание	Команда	Ключи
27	Изменить алгоритм аутентификации OpenVPN-клиентов (необязательно).	<code>rtt(config-openvpn-server)# authentication algorithm <ALGORITHM></code>	<p><ALGORITHM> – алгоритм аутентификации:</p> <ul style="list-style-type: none"> 8-128 bits key size: md4, rsa-md4, md5, rsa-md5, mdc2, rsa-mdc2 8-160 bits key size: sha, sha1, rsa-sha, rsa-sha1, rsa-sha1-2, dsa, dsa-sha, dsa-sha1, dsa-sha1-old, ripemd160, rsa-ripemd160, ecdsa-with-sha1 8-224 bits key size: sha-224, rsa-sha-224 8-256 bits key size: sha-256, rsa-sha-256 8-384 bits key size: sha-384, rsa-sha-384 8-512 bits key size: sha-512, rsa-sha-512, whirlpool <p>Значение по умолчанию: sha.</p>

18.3.2. Пример настройки

Задача:

Настроить OpenVPN-сервер в режиме L3 на маршрутизаторе для подключения удаленных пользователей к ЛВС.

- подсеть OpenVPN-сервера – 10.10.100.0/24;
- режим – L3;
- аутентификация на основе сертификатов.



Решение:

Предварительно нужно выполнить следующие действия:

- Подготовить сертификаты и ключи:
 - Сертификат Удостоверяющего Центра (CA),
 - Ключ и сертификат для OpenVPN-сервера,
 - Ключ Диффи-Хэллмана и HMAC для TLS.

ИЛИ

- Контейнер PKCS12, содержащий в себе сертификат Удостоверяющего Центра (CA) и ключ, и сертификат для OpenVPN-туннеля.
- Настроить зону для интерфейса te1/0/1.
- Указать IP-адреса для интерфейса te1/0/1.

Импортируем по tftp сертификаты и ключи:

```
rtt# copy tftp://192.168.16.10:/ca.crt crypto:cert/ca.crt
rtt# copy tftp://192.168.16.10:/dh.pem crypto:dh/dh.pem
rtt# copy tftp://192.168.16.10:/server.key crypto:private-key/server.key
rtt# copy tftp://192.168.16.10:/server.crt crypto:cert/server.crt
rtt# copy tftp://192.168.16.10:/ta.key crypto:ta/ta.key
```

Или импортируем PKCS12-контейнер:

```
rtt# copy tftp://192.168.0.1:/container.p12 crypto:pfx/cont.p12
```

Создадим OpenVPN-сервер и подсеть, в которой он будет работать:

```
rtt(config)# remote-access openvpn AP
rtt(config-openvpn)# network 10.10.100.0/24
```

Укажем тип соединения L3 и протокол инкапсуляции:

```
rtt(config-openvpn)# tunnel ip
rtt(config-openvpn)# protocol tcp
```

Объявим подсети ЛВС, которые будут доступны через OpenVPN-соединение и укажем DNS-сервер:

```
rtt(config-openvpn)# route 10.10.0.0/20
rtt(config-openvpn)# dns-server 10.10.1.1
```

Укажем ранее импортированные сертификаты и ключи, которые будут использоваться OpenVPN-сервером:

```
rtt(config-openvpn)# crypto ca ca.crt
rtt(config-openvpn)# crypto dh dh.pem
rtt(config-openvpn)# crypto private-key server.key
rtt(config-openvpn)# crypto cert server.crt
rtt(config-openvpn)# crypto ta ta.key
```

Или укажем импортированный контейнер, DH и TA ключи указываются:

```
rtt(config-openvpn)# crypto dh dh.pem
rtt(config-openvpn)# crypto ta ta.key
rtt(config-openvpn)# crypto pfx pfx.p12
```

Укажем зону безопасности, к которой будут относиться сессии пользователей:

```
rtt(config-openvpn)# security-zone VPN
```

Выберем алгоритм шифрования aes128:

```
rtt(config-openvpn)# encryption algorithm aes128
```

Включим OpenVPN-сервер:

```
rtt(config-openvpn)# enable
```

После применения конфигурации маршрутизатор будет прослушивать порт 1194 (используется по умолчанию).

Состояние сессий OpenVPN-сервера можно посмотреть командой:

```
rtt# show remote-access status openvpn server AP
```

Счетчики сессий OpenVPN-сервера можно посмотреть командой:

```
rtt# show remote-access counters openvpn server AP
```

Очистить счетчики сессий OpenVPN-сервера можно командой:

```
rtt# clear remote-access counters openvpn server AP
```

Завершить сессию пользователя fedor OpenVPN-сервера можно одной из следующих команд:

```
rtt# clear remote-access session openvpn username fedor
rtt# clear remote-access session openvpn server AP username fedor
```

Конфигурацию OpenVPN-сервера можно посмотреть командой:

```
rtt# show remote-access configuration openvpn AP
```



Помимо создания OpenVPN-сервера необходимо в firewall открыть TCP-порт 1194.

18.4. Настройка сервера удаленного доступа к корпоративной сети по WireGuard-протоколу

WireGuard – простой, быстрый и современный VPN, использующий современную криптографию (ChaCha20, Poly1305, Curve25519, BLAKE2s, SipHash24, HKDF). WireGuard надежно инкапсулирует IP-пакеты поверх UDP. В основе WireGuard лежит концепция под названием «Маршрутизация криптоключей», которая работает путем связывания открытых ключей со списком IP-адресов туннеля, разрешенным находиться внутри туннеля. Каждый сетевой интерфейс имеет закрытый ключ и список пиров. У каждого узла есть открытый ключ. Открытые ключи короткие и простые и используются узлами для аутентификации друг друга. Их можно передавать для использования в файлах конфигурации любым внешним методом аналогично тому, как можно отправить открытый ключ SSH для доступа к серверу.

18.4.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать профиль WireGuard-сервера.	<code>rtt(config)# remote-access wireguard <NAME></code>	<NAME> – имя профиля WireGuard-сервера, задаётся строкой до 31 символа.
2	Указать описание конфигурируемого сервера (необязательно).	<code>rtt(config-wireguard-server)# description <DESCRIPTION></code>	<DESCRIPTION> – описание WireGuard-сервера, задаётся строкой до 255 символов.
3	Определить статический IP-адрес конфигурируемого сервера.	<code>rtt(config-wireguard-server)# local-address <ADDR/LEN></code>	<ADDR/LEN> – IP-адрес и длина маски подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].
4	Указать UDP-порт, который будет прослушиваться WireGuard-сервером (необязательно).	<code>rtt(config-wireguard-server)# port <PORT></code>	<PORT> – UDP-порт, принимает значения [1..65535].
5	Отключить функции Firewall или включить WireGuard-сервер в зону безопасности и настроить правила взаимодействия между зонами (см. раздел Конфигурирование Firewall).	<code>rtt(config-wireguard-server)# ip firewall disable</code>	
		<code>rtt(config-wireguard-server)# security-zone <NAME></code>	<NAME> – имя зоны безопасности, задаётся строкой до 31 символа.
6	Задать MTU (необязательно).	<code>rtt(config-wireguard-server)# mtu <MTU></code>	<MTU> – 552–10000. Значение по умолчанию: 1500.

Шаг	Описание	Команда	Ключи
7	Указать приватный ключ WireGuard-сервера.	<code>rtt (config-wireguard-server) # private-key <NAME></code>	<NAME> – имя приватного ключа, задаётся строкой до 31 символа.
8	Включить профиль WireGuard-сервера.	<code>rtt (config-wireguard-server) # enable</code>	
9	Перейти к настройке разрешённых туннелей WireGuard-сервера.	<code>rtt (config-wireguard-server) # peer <COUNT></code>	<COUNT> – номер соответствующего пира, принимает значения [1..16].
10	Указать описание туннеля (необязательно).	<code>rtt (config-wireguard-server-peer) # description <DESCRIPTION></code>	<DESCRIPTION> – описание WireGuard-сервера, задаётся строкой до 255 символов.
11	Указать публичный ключ туннеля.	<code>rtt (config-wireguard-server-peer) # public-key <NAME></code>	<NAME> – имя публичного ключа, задаётся строкой до 31 символа.
12	Указать pre-shared-key для настраиваемого туннеля (необязательно).	<code>rtt (config-wireguard-server-peer) # pre-shared-key <TYPE> <WORD></code>	<p><TYPE> – тип аргумента, устанавливаемый в качестве симметричного ключа:</p> <ul style="list-style-type: none"> • ascii-text – указать симметричный ключ в виде ASCII-текста, который будет сконвертирован в формат Base64; • base64 – указать симметричный ключ в формате Base64; • encrypted – указать симметричный ключ в зашифрованном виде. <p><WORD> – вводимый симметричный ключ. В зависимости от типа аргумента имеет длину 32 символа для ascii-text, 44 символа для формата base64 и 64 символа для зашифрованного вида.</p>

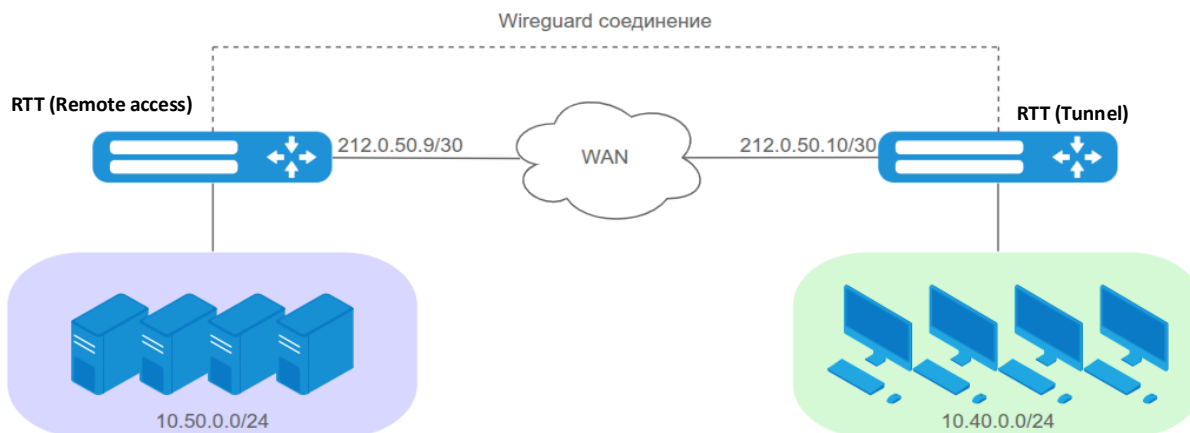
Шаг	Описание	Команда	Ключи
13	Указать список IP-адресов, которым будет разрешено находиться внутри туннеля.	<pre>rtt(config-wireguard- server-peer)# access- addresses <TYPE> {<FROM-ADDR> - <TO-ADDR> <OBJ-GROUP-NETWORK-NAME> <ADDR/LEN>}</pre>	<p><TYPE> – тип аргумента, устанавливаемый в качестве адреса:</p> <ul style="list-style-type: none"> • address-range – указать диапазон IPv4-адресов; • object-group – указать имя профиля; • prefix – указать адрес подсети и префикс. <p><FROM-ADDR> – начальный IP-адрес диапазона;</p> <p><TO-ADDR> – конечный IP-адрес диапазона;</p> <p><OBJ-GROUP-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа;</p> <p><ADDR/LEN> – IP-адрес и маска подсети.</p>
14	Включить туннель.	<pre>rtt(config-wireguard- server-peer)# enable</pre>	

18.4.2. Пример настройки

Задача:

Настроить WireGuard-сервер на маршрутизаторе для подключения удаленных пользователей к серверам организации.

- адресация внутри туннеля – 110.0.0.0/30;
- порт подключения к серверу – 43020;
- адрес WireGuard-сервера внутри туннеля – 110.0.0.1.



Решение:

Создадим ключевую пару x25519, которая будет использоваться в работе WireGuard:

```
rtt# crypto generate private-key x25519 filename wg_server_private.key
rtt# crypto generate public-key x25519 private-key wg_server_private.key
filename wg_server_public.key
```

Для успешной работы необходимо совершить обмен открытыми криптографическими ключами с удаленной стороной любым удобным способом. На данном этапе настройке открытым криптографическим ключом является файл с именем **wg_server_public.key**, который хранится в **crypto:public-key**:

```
rtt# show crypto certificates public-key
File name
-----
wg_server_public.key
```

Создадим object-group network со списком IP-адресов, которым будет разрешено прохождение через туннель:

```
rtt(config)# object-group network WG_CLIENTS
rtt(config-object-group-network)# ip address-range 10.40.0.10-10.40.0.20
```

Создадим профиль WireGuard-сервера, зададим локальный адрес сервера, порт для прослушивания и выставим MTU:

```
rtt(config)# remote-access wireguard WG
rtt(config-wireguard-server)# local-address 110.0.0.1/30
rtt(config-wireguard-server)# port 43020
rtt(config-wireguard-server)# mtu 1420
```

Укажем приватный ключ сервера и отключим Firewall:

```
rtt(config-wireguard-server)# private-key wg_server_private.key
rtt(config-wireguard-server)# ip firewall disable
```

Перейдём в настройки разрешённого туннеля, укажем связку публичного ключа клиента и разрешённого IP-адреса:

```
rtt(config-wireguard-server)# peer 1
rtt(config-wireguard-server-peer)# public-key wg_client_public.key
rtt(config-wireguard-server-peer)# access-addresses object-group WG_CLIENTS
```

Для усиления криптостойкости установим заранее известный симметричный ключ:

```
rtt(config-wireguard-server-peer)# pre-shared-key base64
r4u48oYTouJ+j1GrAtVWRIZqlQ2YLjEZEvc+Yttc6R4=
```

Включим туннель и WireGuard-сервер:

```
rtt(config-wireguard-server-peer)# enable
rtt(config-wireguard-server-peer)# exit
rtt(config-wireguard-server)# enable
```

После применения конфигурации маршрутизатор будет прослушивать порт 43020.

Счётчики сессий WireGuard-сервера можно посмотреть командой:

```
rtt# show remote-access counters wireguard server WG
```

Очистить счётчики сессий WireGuard-сервера можно командой:

```
rtt# clear remote-access counters wireguard server WG
```

Конфигурацию WireGuard-сервера можно посмотреть командой:

```
rtt# show remote-access configuration wireguard WG
```

18.4.3. Пример настройки правил Firewall для совместной работы с WireGuard-сервером

Задача:

Настроить правила Firewall таким образом, чтобы разрешить обращения клиентов на 80 порт сервера 10.50.0.10, остальное взаимодействие запретить.

Решение:

Создадим зоны безопасности WAN, SERVERS, WIREGUARD и назначим их на соответствующие интерфейсы и включим Firewall:

```
rtt(config)# security zone WAN
rtt(config-security-zone)# exit
rtt(config)# security zone SERVERS
rtt(config-security-zone)# exit
rtt(config)# security zone WIREGUARD
```



```
rtt(config-security-zone)# exit
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# security-zone WAN
rtt(config-if-gi)# no ip firewall disable
rtt(config-if-gi)# exit
rtt(config)# interface gigabitethernet 1/0/2
rtt(config-if-gi)# security-zone SERVERS
rtt(config-if-gi)# no ip firewall disable
rtt(config-if-gi)# exit
rtt(config)# remote-access wireguard WG
rtt(config-wireguard-server)# security-zone WIREGUARD
rtt(config-wireguard-server)# no ip firewall disable
rtt(config-wireguard-server)# exit
```

Создадим правила, разрешающие работу WireGuard:

```
rtt(config)# security zone-pair WAN self
rtt(config-security-zone-pair)# rule 10
rtt(config-security-zone-pair-rule)# description "Permit wireguard-traffic"
rtt(config-security-zone-pair-rule)# action permit
rtt(config-security-zone-pair-rule)# match protocol udp
rtt(config-security-zone-pair-rule)# match destination-port port-range 43020
rtt(config-security-zone-pair-rule)# enable
rtt(config-security-zone-pair-rule)# exit
rtt(config-security-zone-pair)# rule 100
rtt(config-security-zone-pair-rule)# action deny
rtt(config-security-zone-pair-rule)# enable
rtt(config-security-zone-pair-rule)# exit
rtt(config-security-zone-pair)# exit
```

Создадим правила, разрешающие обращения на 80 порт сервера 10.50.0.10 из туннеля WireGuard:

```
rtt(config)# security zone-pair WIREGUARD SERVERS
rtt(config-security-zone-pair)# rule 10
rtt(config-security-zone-pair-rule)# action permit
rtt(config-security-zone-pair-rule)# match protocol tcp
rtt(config-security-zone-pair-rule)# match destination-address address-range
10.50.0.10
rtt(config-security-zone-pair-rule)# match destination-port port-range 80
rtt(config-security-zone-pair-rule)# enable
rtt(config-security-zone-pair-rule)# exit
rtt(config-security-zone-pair)# rule 100
rtt(config-security-zone-pair-rule)# action deny
rtt(config-security-zone-pair-rule)# enable
rtt(config-security-zone-pair-rule)# exit
rtt(config-security-zone-pair)# exit
```

Проверим работоспособность:

```
rtt@client:~$ ping 10.50.0.10 -c 4
PING 10.50.0.10 (10.50.0.10) 56(84) bytes of data.

--- 10.50.0.10 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3090ms
```

```

rtt@client:~$ hping3 10.50.0.10 -c 4 -S -p 80
HPING 10.50.0.10 (ens3 10.50.0.10): S set, 40 headers + 0 data bytes
len=46 ip=10.50.0.10 ttl=62 DF id=0 sport=80 flags=RA seq=0 win=0 rtt=2.0 ms
len=46 ip=10.50.0.10 ttl=62 DF id=0 sport=80 flags=RA seq=1 win=0 rtt=2.8 ms
len=46 ip=10.50.0.10 ttl=62 DF id=0 sport=80 flags=RA seq=2 win=0 rtt=2.6 ms
len=46 ip=10.50.0.10 ttl=62 DF id=0 sport=80 flags=RA seq=3 win=0 rtt=2.3 ms

--- 10.50.0.10 hping statistic ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 2.0/2.4/2.8 ms

```

18.5. Настройка клиента удаленного доступа по протоколу PPPoE

PPPoE – это туннелирующий протокол (tunneling protocol), который позволяет инкапсулировать IP PPP через соединения Ethernet и обладает программными возможностями PPP-соединений, что позволяет использовать его для виртуальных соединений на соседнюю Ethernet-машину и устанавливать соединение точка-точка, которое используется для транспортировки IP-пакетов, а также работает с возможностями PPP. Это позволяет применять традиционное PPP-ориентированное ПО для настройки соединения, которое использует не последовательный канал, а пакетно-ориентированную сеть (например, Ethernet), чтобы организовать классическое соединение с логином и паролем для Интернет-соединений. Кроме того, IP-адрес по другую сторону соединения назначается, только когда PPPoE-соединение открыто, позволяя динамическое переиспользование IP-адресов.

18.5.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать PPPoE-туннель и перейти в режим конфигурирования PPPoE-клиента.	<code>rtt(config)# tunnel pppoe <PPPoE></code>	<PPPoE> – порядковый номер туннеля от 1 до 10.
2	Указать описание конфигурируемого клиента (необязательно).	<code>rtt(config-pppoe)# description <DESCRIPTION></code>	<DESCRIPTION> – описание PPPoE-туннеля, задаётся строкой до 255 символов.
3	Указать имя экземпляра VRF, в котором будут использоваться PPPoE-клиент (необязательно).	<code>rtt(config-pppoe)# ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.
4	Указать интерфейс, через который будет устанавливаться PPPoE соединение.	<code>rtt(config-pppoe)# interface <IF></code>	<IF> – интерфейс или группа интерфейсов.
5	Указать имя пользователя и пароль для подключения к PPPoE-серверу.	<code>rtt(config-pppoe)# username <NAME> password ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED- TEXT> }</code>	<NAME> – имя пользователя, задаётся строкой до 31 символа; <CLEAR-TEXT> – пароль, задаётся строкой [8 .. 64] символов; <ENCRYPTED-TEXT> – зашифрованный пароль, задаётся строкой [16..128] символов.

Шаг	Описание	Команда	Ключи
6	Включить PPPoE-туннель в зону безопасности и настроить правила взаимодействия между зонами (см. раздел Конфигурирование Firewall).	<code>rtt(config-pppoe)# security-zone <NAME></code>	<NAME> – имя зоны безопасности, задаётся строкой до 31 символа.
7	Активировать конфигурируемый профиль.	<code>rtt(config-pppoe)# enable</code>	
8	Указать метод аутентификации (необязательно).	<code>rtt(config-pppoe)# authentication method <METHOD></code>	<METHOD> – метод аутентификации, возможные значения: chap, mschap, mschap-v2, eap, pap. Значение по умолчанию: chap.
9	Игнорировать dns-сервер через данный РРрЕ-туннель (необязательно).	<code>rtt(config-pppoe)# ignore nameserver</code>	
10	Включить отказ от получения маршрута по умолчанию от РРрЕ-сервера (необязательно).	<code>rtt(config-pppoe)# ignore-default-route</code>	
11	Указать интервал времени, за который усредняется статистика о нагрузке (необязательно).	<code>rtt(config-pppoe)# load-average <TIME></code>	<TIME> – интервал времени в секундах от 5 до 150 (по умолчанию 5 с).
12	Указать размер MTU (MaximumTransmissionUnit) для РРрЕ-туннеля. MTU более 1500 будет активно только если применена команда system jumbo-frames (необязательно).	<code>rtt(config-pppoe)# mtu <MTU></code>	<MTU> – значение MTU, принимает значения в диапазоне [552..1500]. Значение по умолчанию: 1500.
13	Изменить количество неудачных data-link тестов перед разрывом сессии (необязательно).	<code>rtt(config-pppoe)# ppp failure-count <NUM></code>	<NUM> – количество неудачных data-link тестов, задается в диапазоне [1..100]. Значение по умолчанию: 10.
14	Изменить интервал времени в секундах, по истечении которого маршрутизатор отправляет keepalive-сообщение (необязательно).	<code>rtt(config-pppoe)# ppp timeout keepalive <TIME></code>	<TIME> – время в секундах, задается в диапазоне [1..32767]. Значение по умолчанию: 10.
15	Переопределить значение поля MSS (Maximum segment size) во входящих TCP-пакетах (необязательно).	<code>rtt(config-pppoe)# ip tcp adjust-mss <MSS></code>	<MSS> – значение MSS, принимает значения в диапазоне [500..1460]. Значение по умолчанию: 1460.
16	Включить запись статистики использования текущего туннеля (необязательно).	<code>rtt(config-pppoe)# history statistics</code>	

Шаг	Описание	Команда	Ключи
<p>Также для PPPoE-клиента возможно настроить:</p> <ul style="list-style-type: none"> • QoS в базовом или расширенном режимах (см. раздел Управление QoS); • Прoxy (см. раздел Проксирование HTTP/HTTPS-трафика); • Мониторинг трафика (см. разделы Настройка Netflow и Настройка sFlow). 			

18.5.2. Пример настройки

Задача:

Настроить PPPoE-клиент на маршрутизаторе.

- Учетные записи для подключения – tester;
- Пароли учетных записей – password;
- Подключение должно осуществляться с интерфейса gigabitethernet 1/0/7.



Интерфейс, с которого будет осуществляться PPPoE-соединение, должен работать в режиме routerport (кроме случаев использования bridge).

Интерфейс, с которого будет осуществляться PPPoE-соединение, должен работать в режиме routerport (кроме случаев использования bridge).



Решение:

Предварительно должен быть настроен PPPoE-сервер с соответствующими учетными записями. Также на устройстве должны быть настроены зоны безопасности и описаны правила их взаимодействия.

Зайдем в режим конфигурирования PPPoE-туннеля и зададим пользователя и пароль для подключения к PPPoE-серверу:

```

rtd# configure
rtd(config)# tunnel pppoe 1
rtd(config-pppoe)# username tester password ascii-text password

```

Укажем интерфейс, через который будет устанавливаться PPPoE-соединение:

```
rtt(config-pppoe)# interface gigabitethernet 1/0/7
rtt(config-pppoe)# enable
```

Настроим зону безопасности:

```
rtt(config-pppoe)# security-zone untrust
```

Опционально для PPPoE-туннеля можно указать следующие параметры:

Изменить метод аутентификации:

```
rtt(config-pppoe)# authentication method
METHOD Select PPP authentication method:
    chap
    mschap
    mschap-v2
    eap
    pap
```

Игнорировать полученный маршрут по умолчанию, выданные PPPoE-сервером:

```
rtt(config-pppoe)# ignore-default-route
```

Переопределить значение поля MSS (Maximum segment size) во входящих TCP-пакетах:

```
rtt(config-pppoe)# ip tcp adjust-mss 1452
```

Указать размер MTU (Maximum Transmission Unit):

```
rtt(config-pppoe)# mtu 1496
```

Изменить количество неудачных data-link тестов перед разрывом сессии:

```
rtt(config-pppoe)# ppp failure-count 15
```

Установить интервал времени в секундах, по истечении которого маршрутизатор отправляет keepalive-сообщение:

```
rtt(config-pppoe)# ppp timeout keepalive 15
```

Состояние PPPoE-туннеля можно посмотреть командой:

```
rtt# show tunnels status pppoe 1
```

Счетчики входящих и отправленных пакетов PPPoE-туннеля можно посмотреть командой:

```
rtt# show tunnels counters pppoe 1
```

Конфигурацию PPPoE-туннеля можно посмотреть командой:

```
rtt# show tunnels configuration pppoe 1
```

18.6. Настройка клиента удаленного доступа по протоколу PPTP

PPTP (англ. Point-to-Point Tunneling Protocol) – туннельный протокол типа точка-точка, позволяющий устанавливать защищённое соединение за счёт создания специального туннеля в обычной незащищенной сети. PPTP помещает (инкапсулирует) кадры PPP в IP-пакеты для передачи по глобальной IP-сети, например, Интернет. PPTP может также использоваться для организации туннеля между двумя локальными сетями. PPTP использует дополнительное TCP-соединение для обслуживания туннеля.

18.6.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать PPTP-туннель и перейти в режим его конфигурирования.	<code>rtt(config)# tunnel pptp <INDEX></code>	<INDEX> – идентификатор туннеля в диапазоне: [1..10].
2	Указать описание конфигурируемого туннеля (необязательно).	<code>rtt(config-pptp)# description <DESCRIPTION></code>	<DESCRIPTION> – описание туннеля, задается строкой до 255 символов.
3	Указать экземпляр VRF, в котором будет работать данный PPTP-туннель (не обязательно).	<code>rtt(config-pptp)# ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.
4	Включить PPTP-туннель в зону безопасности и настроить правила взаимодействия между зонами или отключить firewall (см. раздел Конфигурирование Firewall).	<code>rtt(config-pptp)# security-zone <NAME></code>	<NAME> – имя зоны безопасности, задаётся строкой до 31 символа.
		<code>rtt(config-pptp)# ip firewall disable</code>	
5	Установить удаленный IP-адрес для установки туннеля.	<code>rtt(config-pptp)# remote address <ADDR></code>	<ADDR> – IP-адрес локального шлюза, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
6	Установить размер MTU (MaximumTransmissionUnit) для туннеля (необязательно)	<code>rtt(config-pptp)# mtu <MTU></code>	<MTU> – значение MTU, принимает значения в диапазоне [552..10000]. Значение по умолчанию: 1500.

Шаг	Описание	Команда	Ключи
7	Указать пользователя и установить пароль в открытой или зашифрованной форме для аутентификации удаленной стороны.	<code>rtt(config-pptp)# username <NAME> password ascii-text { <WORD> encrypted <HEX> }</code>	<p><NAME> – имя пользователя, задается строкой до 31 символа.</p> <p><WORD> – пароль в открытой форме, задается строкой [8..64] символов, может включать символы [0-9a-fA-F].</p> <p><HEX> – пароль в зашифрованной форме, задается строкой [16..128] символов.</p>
8	Активировать туннель.	<code>rtt(config-pptp)# enable</code>	
9	Переопределить значение поля MSS (Maximum segment size) во входящих TCP-пакетах (необязательно).	<code>rtt(config-pptp)# ip tcp adjust-mss <MSS></code>	<p><MSS> – значение MSS, принимает значения в диапазоне [500..1460].</p> <p>Значение по умолчанию: 1460.</p>
10	Игнорировать dns-сервер через данный PPTP-туннель (необязательно).	<code>rtt(config-pptp)# ignore nameserver</code>	
11	Игнорировать маршрут по умолчанию через данный PPTP-туннель (необязательно)	<code>rtt(config-pptp)# ignore-default-route</code>	
12	Задать интервал времени, за который усредняется статистика о нагрузке на туннель (необязательно).	<code>rtt(config-pptp)# load-average <TIME></code>	<p><TIME> – интервал в секундах, принимает значения [5..150]</p> <p>Значение по умолчанию: 5.</p>
13	Указать метод аутентификации (необязательно).	<code>rtt(config-pptp)# authentication method <METHOD></code>	<p><METHOD> – метод аутентификации, возможные значения: chap, mschap, mschap-v2, eap, pap.</p> <p>Значение по умолчанию: chap.</p>
14	Включить запись статистики использования текущего туннеля (не обязательно).	<code>rtt(config-pptp)# history statistics</code>	
15	Изменить интервал времени в секундах, по истечении которого маршрутизатор отправляет keepalive-сообщение (необязательно).	<code>rtt(config-pptp)# ppp timeout keepalive <TIME ></code>	<p><TIME> – время в секундах, задается в диапазоне [1..32767].</p> <p>Значение по умолчанию: 10.</p>

Шаг	Описание	Команда	Ключи
16	Изменить количество неудачных data-link тестов перед разрывом сессии (необязательно).	<code>rtt(config-pptp) # pptp failure-count <NUM></code>	<p><NUM> – количество неудачных data-link тестов, задается в диапазоне [1..100].</p> <p>Значение по умолчанию: 10.</p>

18.6.2. Пример настройки

Задача:

Настроить PPTP-туннель на маршрутизаторе:

- адрес PPTP-сервера 20.20.0.1;
- учетная запись для подключения – логин: ivan, пароль: simplepass.



Решение:

Создадим туннель PPTP:

```
rtt(config)# tunnel pptp 1
```

Укажем учетную запись (пользователя Ivan) для подключения к серверу:

```
rtt(config-pptp)# username ivan password ascii-text simplepass
```

Укажем удаленный шлюз:

```
rtt(config-pptp)# remote address 20.20.0.1
```

Укажем зону безопасности:

```
rtt(config-pptp)# security-zone VPN
```

Включим туннель PPTP:

```
rtt(config-pptp)# enable
```

Состояние туннеля можно посмотреть командой:


```
rtt# show tunnels status pptp
```

Счетчики входящих и отправленных пакетов можно посмотреть командой:

```
rtt# show tunnels counters pptp
```

Конфигурацию туннеля можно посмотреть командой:

```
rtt# show tunnels configuration pptp
```

18.7. Настройка клиента удаленного доступа по протоколу L2TP

L2TP (англ. Layer 2 Tunneling Protocol – протокол туннелирования второго уровня) – туннельный протокол, использующийся для поддержки виртуальных частных сетей. L2TP помещает (инкапсулирует) кадры PPP в IP-пакеты для передачи по глобальной IP-сети, например, Интернет. L2TP может также использоваться для организации туннеля между двумя локальными сетями. L2TP использует дополнительное UDP-соединение для обслуживания туннеля. L2TP-протокол не предоставляет средств шифрования данных и поэтому он обычно используется в связке с группой протоколов IPsec, которая предоставляет безопасность на пакетном уровне.

18.7.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать L2TP-туннель и перейти в режим его конфигурирования.	<code>rtt(config)# tunnel l2tp <INDEX></code>	<INDEX> – идентификатор туннеля в диапазоне: [1..10].
2	Указать экземпляр VRF, в котором будет работать данный L2TP-туннель (необязательно).	<code>rtt(config-l2tp)# ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задаётся строкой до 31 символа.
3	Указать описание конфигурируемого туннеля (необязательно).	<code>rtt(config-l2tp)# description <DESCRIPTION></code>	<DESCRIPTION> – описание туннеля, задается строкой до 255 символов.
4	Включить L2TP-туннель в зону безопасности и настроить правила взаимодействия между зонами или отключить firewall (см. раздел Конфигурирование Firewall).	<code>rtt(config-l2tp)# security-zone <NAME></code>	<NAME> – имя зоны безопасности, задаётся строкой до 31 символа.
		<code>rtt(config-l2tp)# ip firewall disable</code>	
5	Установить удаленный IP-адрес для установки туннеля.	<code>rtt(config-l2tp)# remote address <ADDR></code>	<ADDR> – IP-адрес локального шлюза, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Шаг	Описание	Команда	Ключи
6	Указать пользователя и установить пароль в открытой или зашифрованной форме для аутентификации удалённой стороны.	<pre>rtt(config-l2tp)# username <NAME> password ascii-text { <WORD> encrypted <HEX> }</pre>	<p><NAME> – имя пользователя, задается строкой до 31 символа.</p> <p><WORD> – пароль в открытой форме, задается строкой [8..64] символов, может включать символы [0-9a-fA-F].</p> <p><HEX> – пароль в зашифрованной форме, задается строкой [16..128] символов.</p>
7	Выбрать метод аутентификации по ключу для IKE-соединения.	<pre>rtt(config-l2tp)# ipsec authentication method pre-shared-key</pre>	
8	Указать общий секретный ключ для аутентификации, который должен совпадать у обеих сторон, устанавливающих туннель.	<pre>rtt(config-l2tp)# ipsec authentication pre- shared-key { ascii-text { <TEXT> encrypted <ENCRYPTED- TEXT> } hexadecimal {<HEX> encrypted <ENCRYPTED- HEX> } }</pre>	<p><TEXT> – строка [1..64] ASCII-символов;</p> <p><HEX> – число размером [1..32] байт, задается строкой [2..128] символов в шестнадцатеричном формате (0xYYYY...) или (YYYY...);</p> <p><ENCRYPTED-TEXT> – зашифрованный пароль размером [1..32] байт, задается строкой [2..128] символов;</p> <p><ENCRYPTED-HEX> – зашифрованное число размером [2..64] байт, задается строкой [2..256] символов.</p>
9	Ограничить используемые методы аутентификации и шифрования протокола IKE (необязательно).	<pre>rtt(config-l2tp)# ipsec ike proposal <NAME></pre>	<NAME> – имя ранее созданного профиля протокола IKE, задается строкой до 31 символа.
10	Включить пересогласование ключей до разрыва IKE-соединения (необязательно)	<pre>rtt(config-l2tp)# ipsec ike rekey enable</pre>	
11	Ограничить используемые методы аутентификации и шифрования протокола IPsec (не обязательно).	<pre>rtt(config-l2tp)# ipsec proposal <NAME></pre>	<NAME> – имя ранее созданного профиля IPsec, задается строкой до 31 символа.
12	Определяется номер UDP-порта по которому устанавливается соединение с l2tp-сервером (необязательно).	<pre>rtt(config-l2tp)# port <PORT></pre>	<p><PORT> – номер UDP-порта, задается в диапазоне [1024..65535].</p> <p>Значение по умолчанию: 1701.</p>
13	Активировать туннель.	<pre>rtt(config-l2tp)# enable</pre>	

Шаг	Описание	Команда	Ключи
14	Установить размер MTU (MaximumTransmissionUnit) для туннеля (необязательно).	<code>rtt(config-l2tp) # mtu <MTU></code>	<MTU> – значение MTU, принимает значения в диапазоне [552..10000]. Значение по умолчанию: 1500.
15	Игнорировать dns-сервер через данный L2TP-туннель (необязательно).	<code>rtt(config-l2tp) # ignore nameserver</code>	
16	Игнорировать маршрут по умолчанию через данный L2TP-туннель (необязательно)	<code>rtt(config-l2tp) # ignore-default-route</code>	
17	Указать метод аутентификации (необязательно).	<code>rtt(config-l2tp) # authentication method <METHOD></code>	<METHOD> – метод аутентификации, возможные значения: chap, mschap, mschap-v2, eap, pap Значение по умолчанию: chap.
18	Задать интервал времени, за который усредняется статистика о нагрузке на туннель (необязательно).	<code>rtt(config-l2tp) # load-average <TIME></code>	<TIME> – интервал в секундах, принимает значения [5..150] Значение по умолчанию: 5.
19	Изменить интервал времени в секундах, по истечении которого маршрутизатор отправляет keepalive-сообщение (необязательно).	<code>rtt(config-l2tp) # ppp timeout keepalive <TIME></code>	<TIME> – время в секундах, задается в диапазоне [1..32767]. Значение по умолчанию: 10.
20	Изменить количество неудачных data-link тестов перед разрывом сессии (необязательно).	<code>rtt(config-l2tp) # ppp failure-count <NUM></code>	<NUM> – количество неудачных data-link тестов, задается в диапазоне [1..100]. Значение по умолчанию: 10.
Также для L2TP-клиента возможно настроить QoS в базовом или расширенном режимах (см. раздел Управление QoS).			

18.7.2. Пример настройки

Задача:

Настроить PPTP-туннель на маршрутизаторе:

- адрес PPTP-сервера 20.20.0.1;
- учетная запись для подключения – логин: ivan, пароль: simplepass



Решение:

Создадим туннель L2TP:

```
rtt(config)# tunnel l2tp 1
```

Укажем учетную запись (пользователя Ivan) для подключения к серверу:

```
rtt(config-l2tp)# username ivan password ascii-text simplepass
```

Укажем удаленный шлюз:

```
rtt(config-l2tp)# remote address 20.20.0.1
```

Укажем зону безопасности:

```
rtt(config-l2tp)# security-zone VPN
```

Укажем метод аутентификации IPsec:

```
rtt(config-l2tp)# ipsec authentication method pre-shared-key
```

Укажем ключ безопасности для IPsec:

```
rtt(config-l2tp)# ipsec authentication pre-shared-key ascii-text password
```

Включим туннель L2TP:

```
rtt(config-l2tp)# enable
```

Состояние туннеля можно посмотреть командой:

```
rtt# show tunnels status l2tp
```

Счетчики входящих и отправленных пакетов можно посмотреть командой:

```
rtt# show tunnels counters l2tp
```

Конфигурацию туннеля можно посмотреть командой:

```
rtt# show tunnels configuration l2tp
```

18.8. Настройка клиента удаленного доступа по протоколу WireGuard

WireGuard — простой, быстрый и современный VPN, использующий современную криптографию (ChaCha20, Poly1305, Curve25519, BLAKE2s, SipHash24, HKDF). WireGuard надежно инкапсулирует IP-пакеты поверх UDP. В основе WireGuard лежит концепция под названием «Маршрутизация криптоключей», которая работает путем связывания открытых ключей со списком IP-адресов туннеля, которым разрешено находиться внутри туннеля. Каждый сетевой интерфейс имеет закрытый ключ и список пиров. У каждого узла есть открытый ключ. Открытые ключи короткие и простые и используются узлами для аутентификации друг друга. Их можно передавать для использования в файлах конфигурации любым внешним методом, аналогично тому, как можно отправить открытый ключ SSH для доступа к серверу.

18.8.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать WireGuard-туннель и перейти в режим его конфигурирования.	<code>rtt(config)# tunnel wireguard <INDEX></code>	<INDEX> – идентификатор туннеля в диапазоне: [1..16].
2	Указать экземпляр VRF, в котором будет работать данный Wireguard-туннель (необязательно).	<code>rtt(config-wireguard)# ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задаётся строкой до 31 символа.
3	Указать описание конфигурируемого туннеля (необязательно).	<code>rtt(config-wireguard)# description <DESCRIPTION></code>	<DESCRIPTION> – описание туннеля, задается строкой до 255 символов.
4	Включить Wireguard-туннель в зону безопасности и настроить правила взаимодействия между зонами или отключить firewall (см. раздел Конфигурирование Firewall).	<code>rtt(config-wireguard)# security-zone <NAME></code>	<NAME> – имя зоны безопасности, задаётся строкой до 31 символа.
		<code>rtt(config-wireguard)# ip firewall disable</code>	
5	Определить статический IP-адрес конфигурируемого туннеля.	<code>rtt(config-wireguard)# ip address <ADDR/LEN></code>	<ADDR/LEN> – IP-адрес и длина маски подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].
6	Задать MTU (необязательно).	<code>rtt(config-wireguard)# mtu <MTU></code>	<MTU> – 552–10000. Значение по умолчанию: 1500.
7	Указать приватный ключ WireGuard-клиента.	<code>rtt(config-wireguard)# private-key <NAME></code>	<NAME> – имя приватного ключа, задается строкой до 31 символа.
8	Перейти к настройке разрешенных пиров	<code>rtt(config-wireguard)# peer <COUNT></code>	<COUNT> – номер соответствующего пира, принимает значения [1..16].

Шаг	Описание	Команда	Ключи
9	Указать описание пира (необязательно).	<code>rtt(config-wireguard-peer)# description <DESCRIPTION></code>	<DESCRIPTION> – описание туннеля, задается строкой до 255 символов.
10	Задать значение keepalive (необязательно).	<code>rtt(config-wireguard-peer)# keepalive timeout <SEC></code>	<SEC> – количество секунд, принимает значения [1..32767].
11	Указать публичный ключ WireGuard-пира.	<code>rtt(config-wireguard-peer)# public-key <NAME></code>	<NAME> – имя приватного ключа, задается строкой до 31 символа.
12	Указать pre-shared-key для настраиваемого пира (необязательно).	<code>rtt(config-wireguard-peer)# pre-shared-key <TYPE> <WORD></code>	<p><TYPE> – тип аргумента, устанавливаемый в качестве симметричного ключа:</p> <ul style="list-style-type: none"> • ascii-text – указать симметричный ключ в виде ASCII-текста, который будет сконvertирован в формат Base64; • base64 – указать симметричный ключ в формате Base64; • encrypted – указать симметричный ключ в зашифрованном виде. <p><WORD> - вводимый симметричный ключ. В зависимости от типа аргумента имеет длину 32 символа для ascii-text, 44 символа для формата base64 и 64 символа для зашифрованного вида.</p>
13	Указать IP-адрес удаленного пира.	<code>rtt(config-wireguard-peer)# remote address <ADDR></code>	<ADDR> – IP-адрес локального шлюза, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
14	Указать UDP-порт удаленного пира.	<code>rtt(config-wireguard-peer)# remote port <PORT></code>	<PORT> – UDP-порт, принимает значения [1..65535].

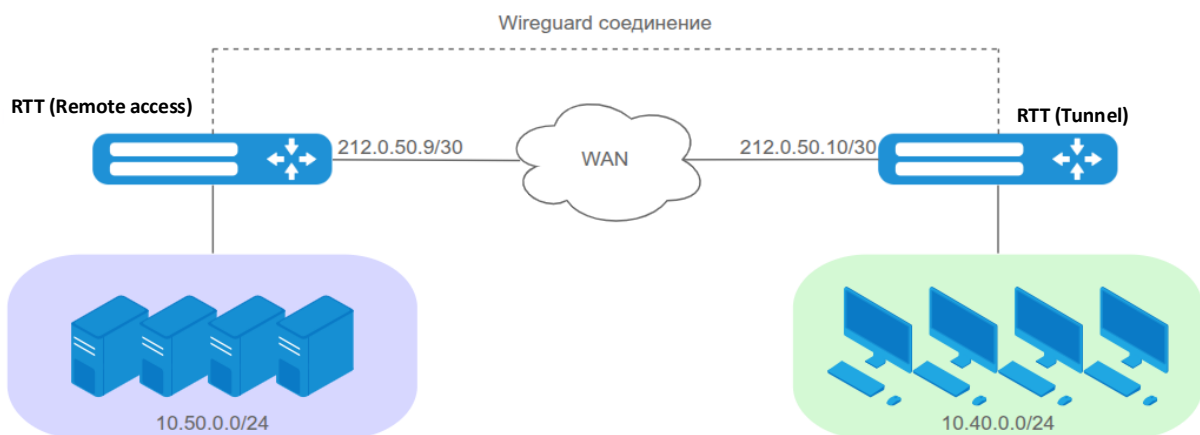
Шаг	Описание	Команда	Ключи
15	Указать список IP-адресов, которым будет разрешено находиться внутри туннеля.	<pre>rtt(config-wireguard-peer) # access-addresses <TYPE> {<FROM-ADDR> - <TO-ADDR> <OBJ-GROUP- NETWORK-NAME> <ADDR/LEN>}</pre>	<p><TYPE> – тип аргумента, устанавливаемый в качестве адреса:</p> <ul style="list-style-type: none"> • address-range – указать диапазон IPv4-адресов; • object-group – указать имя профиля; • prefix – указать адрес подсети и префикс. <p><FROM-ADDR> – начальный IP-адрес диапазона; <TO-ADDR> – конечный IP-адрес диапазона;</p> <p><OBJ-GROUP-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа;</p> <p><ADDR/LEN> – IP-адрес и маска подсети.</p>
16	Активировать пир.	<pre>rtt(config-wireguard-peer) # enable</pre>	
17	Активировать туннель.	<pre>rtt(config-wireguard) # enable</pre>	
18	Задать интервал времени, за который усредняется статистика о нагрузке на туннель (необязательно).	<pre>rtt(config-wireguard) # load-average <TIME></pre>	<p><TIME> – интервал в секундах, принимает значения [5..150]</p> <p>Значение по умолчанию: 5.</p>
19	Включить запись статистики использования текущего туннеля (необязательно).	<pre>rtt(config-wireguard) # history statistics</pre>	

18.8.2. Пример настройки

Задача:

Настроить WireGuard-клиента на маршрутизаторе:

- адресация внутри туннеля – 110.0.0.0/30;
- порт подключения к серверу – 43020;
- адрес WireGuard-клиента внутри туннеля – 110.0.0.2;
- адрес удаленного сервера – 212.0.50.9.



Решение:

Создадим ключевую пару x25519, которая будет использоваться в работе WireGuard:

```
rtt# crypto generate private-key x25519 filename wg_client_private.key
rtt# crypto generate public-key x25519 private-key wg_client_private.key
filename wg_client_public.key
```

Для успешной работы необходимо совершить обмен открытыми криптографическими ключами с удаленной стороной любым удобным способом. На данном этапе настройке открытым криптографическим ключом является файл с именем **wg_client_public.key**, который хранится в **crypto:public-key**:

```
rtt# show crypto certificates public-key
File name
-----
wg_client_public.key
```

Создадим object-group network, в которой будет указан список IP-адресов, которым будет разрешено проходить через туннель:

```
rtt(config)# object-group network WG_SERVERS
rtt(config-object-group-network)# ip address-range 10.50.0.10-10.50.0.15
```

Создадим WireGuard-туннель, зададим локальный адрес и выставим MTU:

```
rtt(config)# tunnel wireguard 1
rtt(config-wireguard)# ip address 110.0.0.2/30
rtt(config-wireguard)# mtu 1420
```

Укажем приватный ключ клиента и отключим Firewall:

```
rtt(config-wireguard)# private-key wg_client_private.key
rtt(config-wireguard)# ip firewall disable
```


Перейдем в настройки разрешённого пира, укажем связку публичного ключа сервера и разрешённого IP-адреса, а также укажем адрес и порт удаленного сервера:

```
rtt(config-wireguard)# peer 1
rtt(config-wireguard-peer)# public-key wg_server_public.key
rtt(config-wireguard-peer)# access-addresses object-group WG_SERVERS
rtt(config-wireguard-peer)# remote address 212.0.50.9
rtt(config-wireguard-peer)# remote port 43020
```

Для усиления криптостойкости установим заранее известный симметричный ключ:

```
rtt(config-wireguard-peer)# pre-shared-key base64
r4u48oYTouJ+j1GrAtVWRIZqlQ2YLjEZEvc+Yttc6R4=
```

Включим пир и WireGuard-туннель:

```
rtt(config-wireguard-peer)# enable
rtt(config-wireguard-peer)# exit
rtt(config-wireguard)# enable
```

Попробуем отправить ICMP с рабочего ПК сотрудника (10.40.0.10) на удаленный сервер (10.50.0.10):

```
rtt@client:~$ ping 10.50.0.10 -c 4
PING 10.50.0.10 (10.50.0.10) 56(84) bytes of data.
64 bytes from 10.50.0.10: icmp_seq=1 ttl=62 time=1.85 ms
64 bytes from 10.50.0.10: icmp_seq=2 ttl=62 time=1.47 ms
64 bytes from 10.50.0.10: icmp_seq=3 ttl=62 time=1.97 ms
64 bytes from 10.50.0.10: icmp_seq=4 ttl=62 time=1.72 ms

--- 10.50.0.10 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 1.465/1.753/1.974/0.188 ms
```

С помощью команды **monitor** можно убедиться, что ICMP-пакеты проходят через wireguard туннель:

```
rtt# monitor wireguard 1
06:54:36.536109 ip: (tos 0x0, ttl 63, id 22999, offset 0, flags [DF], proto ICMP (1),
length 84)
    10.40.0.10 > 10.50.0.10: ICMP echo request, id 1568, seq 12, length 64
06:54:36.537358 ip: (tos 0x0, ttl 63, id 32883, offset 0, flags [none], proto ICMP (1),
length 84)
    10.50.0.10 > 10.40.0.10: ICMP echo reply, id 1568, seq 12, length 64

rtt# monitor gigabitethernet 1/0/1
06:54:36.536109 50:1f:e6:04:51:00 > 50:dd:a1:04:50:00, ethertype IPv4 (0x0800), length
170: (tos 0x0, ttl 64, id 21376, offset 0, flags [none], proto UDP (17), length 156)
    212.0.50.10.40548 > 212.0.50.9.43020: UDP, length 128
06:54:36.537358 50:dd:a1:04:50:00 > 50:1f:e6:04:51:00, ethertype IPv4 (0x0800), length
170: (tos 0x0, ttl 64, id 41730, offset 0, flags [none], proto UDP (17), length 156)
    212.0.50.9.43020 > 212.0.50.10.40548: UDP, length 128
```

Счётчики сессий WireGuard-туннеля можно посмотреть командой:

```
rtt# show tunnels counters wireguard 1
```

Очистить счётчики сессий WireGuard-туннеля можно командой:

```
rtt# clear tunnels counters wireguard 1
```

Конфигурацию WireGuard-туннеля можно посмотреть командой:

```
rtt# show tunnels configuration wireguard 1
```

19.УПРАВЛЕНИЕ СЕРВИСАМИ

19.1. Настройка DHCP-сервера

Встроенный DHCP-сервер маршрутизатора может быть использован для настройки сетевых параметров устройств в локальной сети. DHCP-сервер маршрутизаторов способен передавать дополнительные опции на сетевые устройства, например:

- `default-router` – IP-адрес маршрутизатора, используемого в качестве шлюза по умолчанию;
- `domain-name` – доменное имя, которое должен будет использовать клиент при разрешении имен хостов через Систему Доменных Имен (DNS);
- `dns-server` – список адресов серверов доменных имен в данной сети, о которых должен знать клиент. Адреса серверов в списке располагаются в порядке убывания предпочтения.

19.1.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Включить IPv4/IPv6 DHCP-сервер.	<code>rtt(config)# ip dhcp-server [vrf <VRF>]</code>	<VRF> – имя экземпляра VRF, в рамках которого будет работать DHCP-сервер. Задается строкой до 31 символа.
		<code>rtt(config)# ipv6 dhcp-server [vrf <VRF>]</code>	
2	Задать значение кода DSCP для использования в IP-заголовке исходящих пакетов DHCP-сервера (не обязательно).	<code>rtt(config)# ip dhcp-server dscp <DSCP></code>	<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63]. Значение по умолчанию: 61.
3	Создать пул IPv4/IPv6-адресов DHCP-сервера и перейти в режим его конфигурирования.	<code>rtt(config)# ip dhcp-server pool <NAME> [vrf <VRF>]</code>	<NAME> – имя пула IPv4/IPv6-адресов DHCP-сервера, задается строка до 31 символа.
		<code>rtt(config)# ipv6 dhcp-server pool <NAME> [vrf <VRF>]</code>	<VRF> – имя экземпляра VRF, в рамках которого будет работать данный пул IP-адресов DHCP-сервера. Задается строкой до 31 символа
4	Задать IPv4/IPv6-адрес и маску для подсети, из которой будет выделен пул IPv4/IPv6-адресов.	<code>rtt(config-dhcp-server)# network <ADDR/LEN></code>	<ADDR/LEN> – IP-адрес и префикс подсети, задается в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].
		<code>rtt(config-ipv6-dhcp-server)# network <IPV6-ADDR/LEN></code>	<IPV6-ADDR/LEN> – IP-адрес и префикс подсети, задается в виде X:X:X:X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].

Шаг	Описание	Команда	Ключи
5	Добавить диапазон IPv4/IPv6-адресов к пулу адресов, конфигурируемого DHCP-сервера.	<code>rtt (config-dhcp-server) # address-range <FROM-ADDR>-<TO-ADDR></code>	<p><FROM-ADDR> – начальный IP-адрес диапазона;</p> <p><TO-ADDR> – конечный IP-адрес диапазона,</p> <p>Адреса задаются в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p> <p>Можно указать до 32 диапазонов IP-адресов, список задаётся через запятую.</p>
		<code>rtt (config-ipv6-dhcp-server) # address-range <FROM-ADDR>-<TO-ADDR></code>	<p><FROM-ADDR> – начальный IPv6-адрес диапазона;</p> <p><TO-ADDR> – конечный IP-адрес диапазона;</p> <p>Адреса задаются в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].</p>
6	Добавить IPv4/IPv6-адрес для определенного физического адреса к пулу адресов конфигурируемого DHCP-сервера (не обязательно).	<code>rtt (config-dhcp-server) # address <ADDR> {mac-address <MAC> client-identifier <CI>}</code>	<p><ADDR> – IP-адрес клиента, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><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].</p> <p><CI> – идентификатор клиента согласно DHCP Option 61. Может быть задан в одном из следующих видов:</p> <ul style="list-style-type: none"> • НН:НН:НН:НН:НН:НН: – идентификатор клиента в шестнадцатеричной форме и MAC-адрес клиента; • STRING – текстовая строка длиной от 1 до 64 символов.
		<code>rtt (config-ipv6-dhcp-server) # address <ADDR> mac-address <MAC></code>	<p><IPv6-ADDR> – IPv6-адрес клиента, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];</p> <p><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].</p>

Шаг	Описание	Команда	Ключи
7	Задать список IPv4-адресов шлюзов по умолчанию, которые DHCP-сервер будет сообщать клиентам, используя DHCP-опцию 3.	<code>rtt (config-dhcp-server) # default-router <ADDR></code>	<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.
8	Задать DNS-имя сетевого домена. Имя домена передаётся клиентам в составе DHCP-опции 15 (не обязательно).	<code>rtt (config-dhcp-server) # domain-name <NAME></code> <code>rtt (config-ipv6-dhcp-server) # domain-name <NAME></code>	<NAME> – DNS-имя домена клиента, задаётся строкой до 255 символов.
9	Задать список IPv4/IPv6-адресов DNS-серверов. Список передаётся клиентам в составе DHCP-опции 6 (не обязательно).	<code>rtt (config-dhcp-server) # dns-server <ADDR></code> <code>rtt (config-ipv6-dhcp-server) # dns-server <IPv6-ADDR></code>	<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. <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.
10	Задать максимальное время аренды IP-адресов (не обязательно). Если DHCP-клиент запрашивает время аренды, превосходящее максимальное значение, то будет установлено время, заданное этой командой.	<code>rtt (config-dhcp-server) # max-lease-time <TIME></code> <code>rtt (config-ipv6-dhcp-server) # max-lease-time <TIME></code>	<TIME> – максимальное время аренды IP-адреса, задаётся в формате DD:HH:MM, где: <ul style="list-style-type: none"> • 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.
11	Задать время аренды, на которое клиенту будет выдан IP-адрес (не обязательно). Данное время будет использоваться если клиент не запрашивал определенное время аренды.	<code>rtt (config-dhcp-server) # default-lease-time <TIME></code> <code>rtt (config-ipv6-dhcp-server) # default-lease-time <TIME></code>	<TIME> – максимальное время аренды IP-адреса, задаётся в формате DD:HH:MM, где: <ul style="list-style-type: none"> • 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.
12		<code>rtt (config) # ip dhcp-server vendor-class-id <NAME></code>	<NAME> – идентификатор класса поставщика, задаётся строкой до 31 символа.

Шаг	Описание	Команда	Ключи
	Создать идентификатор класса поставщика (DHCP Опция 60) (не обязательно).	<code>rtt(config)# ipv6 dhcp-server vendor-class-id <NAME></code>	
13	Задать специфическую информацию поставщика (DHCP Опция 43).	<code>rtt(config-dhcp-vendor-id)# vendor-specific-options <HEX></code> <code>rtt(config-ipv6-dhcp-vendor-id)# vendor-specific-options <HEX></code>	<HEX> – специфическая информация поставщика, задаётся в шестнадцатеричном формате до 128 символов.
14	Задать IP-адрес NetBIOS-сервера (DHCP опция 44) (не обязательно).	<code>rtt(config-dhcp-server)# netbios-nameserver <ADDR></code>	<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.
15	Задать IP-адрес tftp-сервера (DHCP Option 150) (не обязательно).	<code>rtt(config-dhcp-server)# tftp-server <ADDR></code>	<ADDR> – IP-адрес DNS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].



Для продления ресурса внутреннего flash-накопителя информация о выданных сервером IP-адресах хранится в энергозависимой памяти. Поэтому стоит учитывать, что при перезагрузке маршрутизатора сервер начинает выдавать адреса заново. Занятость адресов будет проверяться с помощью ICMP-запросов.

19.1.2. Пример настройки

Задача:

Настроить работу DHCP-сервера в локальной сети, относящейся к зоне безопасности «trusted». Задать пул IP-адресов из подсети 192.168.1.0/24 для раздачи клиентам. Задать время аренды адресов 1 день. Настроить передачу клиентам маршрута по умолчанию, доменного имени и адресов DNS-серверов с помощью DHCP-опций.

Решение:

Создадим зону безопасности «trusted» и установим принадлежность используемых сетевых интерфейсов к зонам:

```
rtt# configure
rtt(config)# security zone trusted
rtt(config-zone)# exit
```

Создадим пул адресов с именем «Simple» и добавим в данный пул адресов диапазон IP-адресов для выдачи в аренду клиентам сервера. Укажем параметры подсети, к которой принадлежит данный пул, и время аренды для выдаваемых адресов:

```
rtt# configure
rtt(config)# ip dhcp-server pool Simple
rtt(config-dhcp-server)# network 192.168.1.0/24
rtt(config-dhcp-server)# address-range 192.168.1.100-192.168.1.125
rtt(config-dhcp-server)# default-lease-time 1:00:00
```

Сконфигурируем передачу клиентам дополнительных сетевых параметров:

- маршрут по умолчанию: 192.168.1.1;
- имя домена: rusteletech.loc;
- список DNS-серверов: DNS1: 172.16.0.1, DNS2: 8.8.8.8.

```
rtt(config-dhcp-server)# domain-name "rusteletech.loc"
rtt(config-dhcp-server)# default-router 192.168.1.1
rtt(config-dhcp-server)# dns-server 172.16.0.1,8.8.8.8
rtt(config-dhcp-server)# exit
```

Для того чтобы DHCP-сервер мог раздавать IP-адреса из конфигурируемого пула, на маршрутизаторе должен быть создан IP-интерфейс, принадлежащий к той же подсети, что и адреса пула.

```
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# security-zone trusted
rtt(config-if-gi)# ip address 192.168.1.1/24
rtt(config-if-gi)# exit
```

Для разрешения прохождения сообщений протокола DHCP к серверу необходимо создать соответствующие профили портов, включающие порт источника 68 и порт назначения 67, используемые протоколом DHCP, и создать разрешающее правило в политике безопасности для прохождения пакетов протокола UDP:

```
rtt(config)# object-group service dhcp_server
rtt(config-object-group-service)# port-range 67
rtt(config-object-group-service)# exit
rtt(config)# object-group service dhcp_client
rtt(config-object-group-service)# port-range 68
rtt(config-object-group-service)# exit
rtt(config)# security zone-pair trusted self
rtt(config-zone-pair)# rule 30
rtt(config-zone-rule)# match protocol udp
rtt(config-zone-rule)# match source-port object-group dhcp_client
rtt(config-zone-rule)# match destination-port object-group dhcp_server
rtt(config-zone-rule)# action permit
rtt(config-zone-rule)# enable
rtt(config-zone-rule)# exit
rtt(config-zone-pair)# exit
```

Разрешим работу сервера:

```
rtt(config)# ip dhcp-server
rtt(config)# exit
```

Просмотреть список арендованных адресов можно с помощью команды:

```
rtt# show ip dhcp binding
```

Просмотреть сконфигурированные пулы адресов можно командами:

```
rtt# show ip dhcp server pool
rtt# show ip dhcp server pool Simple
```



Конфигурирование настроек для IPv6 производится по аналогии с IPv4.

19.2. Конфигурирование Destination NAT

Функция Destination NAT (DNAT) состоит в преобразовании IP-адреса назначения у пакетов, проходящих через сетевой шлюз.

DNAT используется для перенаправления трафика, идущего на некоторый «виртуальный» адрес в публичной сети, на «реальный» сервер в локальной сети, находящийся за сетевым шлюзом. Эту функцию можно использовать для организации публичного доступа к серверам, находящимся в частной сети и не имеющим публичного сетевого адреса.

19.2.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Перейти в режим настройки сервиса трансляции адресов получателя.	<code>rtt(config)# nat destination</code>	
2	Создать пул IP-адресов и/или TCP/UDP-портов с определённым именем (не обязательно).	<code>rtt(config-dnat)# pool <NAME></code>	<NAME> – имя пула NAT-адресов, задаётся строкой до 31 символа.
3	Установить внутренний IP-адрес, на который будет заменяться IP-адрес получателя.	<code>rtt(config-dnat-pool)# ip address <ADDR></code>	<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
4	Установить внутренний TCP/UDP-порт, на который будет заменяться TCP/UDP-порт получателя.	<code>rtt(config-dnat-pool)# ip port <PORT></code>	<PORT> – TCP/UDP-порт, принимает значения [1..65535].
5	Создать группу правил с определённым именем.	<code>rtt(config-dnat)# ruleset <NAME></code>	<NAME> – имя группы правил, задаётся строкой до 31 символа.

Шаг	Описание	Команда	Ключи
6	Указать экземпляр VRF, в котором будет работать данная группа правил (не обязательно).	<code>rtt(config-dnat-ruleset)# ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.
7	Задать область применения группы правил. Правила будут применяться только для трафика, идущего из определенной зоны или интерфейса.	<code>rtt(config-dnat-ruleset)# from { zone <NAME> interface <IF> tunnel <TUN> default }</code>	<p><NAME> – имя зоны изоляции;</p> <p><IF> – имя интерфейса устройства;</p> <p><TUN> – имя туннеля устройства.</p> <p>default – обозначает группу правил для всего трафика, источник которого не попал под критерии других групп правил.</p>
8	Задать правило с определённым номером. Правила обрабатываются в порядке возрастания.	<code>rtt(config-dnat-ruleset)# rule <ORDER></code>	<ORDER> – номер правила, принимает значения [1...10000].
9	Задать IP-адреса {отправителя получателя}, для которых должно срабатывать правило.	<code>rtt(config-dnat-rule)# match [not] {source destination}-address { address-range { <ADDR>[-<ADDR>] } prefix { <ADDR/LEN> } object-group { network <OBJ-GROUP-NETWORK-NAME> } }</code>	<p>address-range <ADDR>[-<ADDR>] – диапазон IP-адресов для правил NAT. Если не указывать IP-адрес конца диапазона, то в качестве IP-адреса для срабатывания правила используется только IP-адрес начала диапазона. Параметр задаётся в виде A.B.C.D, где каждая часть принимает значения [0..255];</p> <p>prefix <ADDR/LEN> – IP-подсеть, используемая для срабатывания правила NAT. Параметр задаётся в виде A.B.C.D/E, где каждая часть A – D принимает значения [0..255] и E принимает значения [1..32];</p> <p>object-group network <OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.</p>

Шаг	Описание	Команда	Ключи
10	Задать сервисы (TCP/UDP-портов) {отправителя получателя}, для которых должно срабатывать правило (не обязательно).	<pre>rtt(config-dnat-rule) # match [not] {source destination}- port <TYPE> {<PORT-SET- NAME> <FORM-PORT> - <TO-PORT>}</pre>	<p><TYPE> – тип аргумента, устанавливаемый в качестве адреса:</p> <ul style="list-style-type: none"> address-range – указать диапазон IPv4/IPv6 адресов; object-group – указать имя профиля; any – установить в качестве адреса любой адрес. <p><PORT-SET-NAME> – имя профиля порта, задаётся строкой до 31 символа;</p> <p><FROM-PORT> – начальный порт диапазона;</p> <p><TO-PORT> – конечный порт диапазона.</p>
11	Установить имя или номер IP-протокола, для которого должно срабатывать правило (не обязательно).	<pre>rtt(config-dnat-rule) # match [not] {protocol <TYPE> protocol-id <ID> }</pre>	<p><TYPE> – тип протокола, принимает значения: esp, icmp, icmp6, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rsvp, l2tp, gre. Значение «any» указывает на любой тип протокола.</p> <p><ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].</p>
12	Задать тип и код сообщений протокола ICMP, для которых должно срабатывать правило (если в качестве протокола выбран ICMP) (не обязательно).	<pre>rtt(config-dnat-rule) # match [not] icmp {<ICMP_TYPE><ICMP_CODE> <TYPE-NAME>}</pre>	<p><ICMP_TYPE> – тип сообщения протокола ICMP, принимает значения [0..255].</p> <p><ICMP_CODE> – код сообщения протокола ICMP, принимает значения [0..255]. Значение «any» указывает на любой код сообщения.</p> <p><TYPE-NAME> – имя типа ICMP-сообщения.</p>
13	Задать действие «трансляция адреса и порта получателя» для трафика, удовлетворяющего критериям, заданным командами «match».	<pre>rtt(config-dnat-rule) # action destination-nat { off pool <NAME> netmap <ADDR/LEN> }</pre>	<p>off – трансляция отключена;</p> <p>pool <NAME> – имя пула, содержащего набор IP-адресов и/или TCP/UDP-портов;</p> <p>netmap <ADDR/LEN> – IP-адрес и маска подсети, используемые при трансляции. Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].</p>

Шаг	Описание	Команда	Ключи
14	Активировать конфигурируемое правило.	<code>rtt(config-dnat-rule) # enable</code>	
15	Включить функцию отслеживания сессий уровня приложений для протоколов FTP, SIP, H323, netbios-ns, PPTP (не обязательно).	<code>rtt(config) # ip firewall sessions tracking</code> <code>{<PROTOCOL> sip [port <OBJECT-GROUP-SERVICE>] all}</code>	<p>all – включает функцию отслеживания сессий уровня приложений для всех доступных протоколов;</p> <p><PROTOCOL> – протокол уровня приложений, сессии которого должны отслеживаться, принимает значения [ftp, h323, pptp, netbios-ns];</p> <p><OBJECT-GROUP-SERVICE> – имя профиля TCP/UDP-портов sip-сессии, задаётся строкой до 31 символа. Если группа не указана, то отслеживание сессий sip будет осуществляться для порта 5060.</p>
16	Включить функцию трансляции IP-адресов в заголовках уровня приложений (не обязательно).	<code>rtt(config) # nat alg {<PROTOCOL> all}</code>	<p>all – включает трансляцию IP-адресов в заголовках всех доступных протоколов.</p> <p><PROTOCOL> – протокол уровня приложений, в заголовках которого должна работать трансляция адресов, принимает значения [ftp, h323, pptp, netbios-ns, gre, sip, tftp].</p>



При использовании ключа not правило будет срабатывать для значений, которые не входят в указанный профиль.

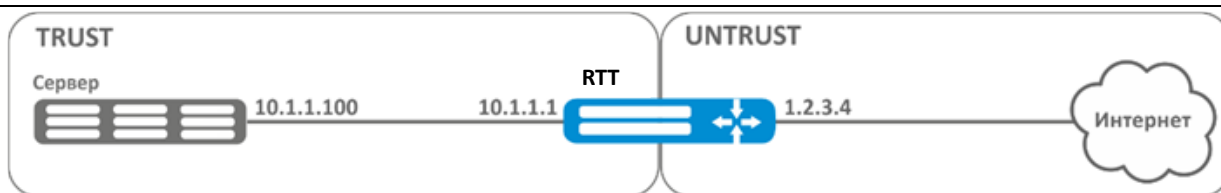
Каждая команда «match» может содержать ключ «not». При использовании данного ключа под правило будут подпадать пакеты, не удовлетворяющие заданному критерию.

Более подробная информация о командах для настройки маршрутизатора содержится в «Справочник команд CLI».

19.2.2. Пример настройки Destination NAT

Задача:

Организовать доступ из публичной сети, относящейся к зоне «UNTRUST», к серверу локальной сети в зоне «TRUST». Адрес сервера в локальной сети – 10.1.1.100. Сервер должен быть доступным извне по адресу 1.2.3.4, доступный порт 80.



Решение:

Создадим зоны безопасности «UNTRUST» и «TRUST». Установим принадлежность используемых сетевых интерфейсов к зонам. Одновременно назначим IP-адреса интерфейсам.

```

rtt# configure
rtt(config)# security zone UNTRUST
rtt(config-zone)# exit
rtt(config)# security zone TRUST
rtt(config-zone)# exit
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# security-zone TRUST
rtt(config-if-gi)# ip address 10.1.1.1/25
rtt(config-if-gi)# exit
rtt(config)# interface tengigabitethernet 1/0/1
rtt(config-if-te)# ip address 1.2.3.4/29
rtt(config-if-te)# security-zone UNTRUST
rtt(config-if-te)# exit

```

Создадим профили IP-адресов и портов, которые потребуются для настройки правил Firewall и правил DNAT.

- NET_UPLINK – профиль адресов публичной сети;
- SERVER_IP – профиль адресов локальной сети;
- SRV_HTTP – профиль портов.

```

rtt(config)# object-group network NET_UPLINK
rtt(config-object-group-network)# ip address 1.2.3.4
rtt(config-object-group-network)# exit
rtt(config)# object-group service SRV_HTTP
rtt(config-object-group-service)# port 80
rtt(config-object-group-service)# exit
rtt(config)# object-group network SERVER_IP
rtt(config-object-group-network)# ip address 10.1.1.100
rtt(config-object-group-network)# exit

```

Войдем в режим конфигурирования функции DNAT и создадим пул адресов и портов назначения, в которые будут транслироваться адреса пакетов, поступающие на адрес 1.2.3.4 из внешней сети.

```

rtt(config)# nat destination
rtt(config-dnat)# pool SERVER_POOL
rtt(config-dnat-pool)# ip address 10.1.1.100
rtt(config-dnat-pool)# ip port 80
rtt(config-dnat-pool)# exit

```

Создадим набор правил «DNAT», в соответствии с которыми будет производиться трансляция адресов. В атрибутах набора укажем, что правила применяются только для пакетов, пришедших из

зоны «UNTRUST». Набор правил включает в себя требования соответствия данных по адресу и порту назначения (`match destination-address`, `match destination-port`) и по протоколу. Кроме этого, в наборе задано действие, применяемое к данным, удовлетворяющим всем правилам (`action destination-nat`). Набор правил вводится в действие командой **enable**.

```
rtt(config-dnat)# ruleset DNAT
rtt(config-dnat-ruleset)# from zone UNTRUST
rtt(config-dnat-ruleset)# rule 1
rtt(config-dnat-rule)# match destination-address object-group NET_UPLINK
rtt(config-dnat-rule)# match protocol tcp
rtt(config-dnat-rule)# match destination-port object-group SRV_HTTP
rtt(config-dnat-rule)# action destination-nat pool SERVER_POOL
rtt(config-dnat-rule)# enable
rtt(config-dnat-rule)# exit
rtt(config-dnat-ruleset)# exit
rtt(config-dnat)# exit
```

Для пропуска трафика, идущего из зоны «UNTRUST» в «TRUST», создадим соответствующую пару зон. Пропускать следует только трафик с адресом назначения, соответствующим заданному в профиле «SERVER_IP» и прошедший преобразование DNAT.

```
rtt(config)# security zone-pair UNTRUST TRUST
rtt(config-zone-pair)# rule 1
rtt(config-zone-pair-rule)# match destination-address object-group network
SERVER_IP
rtt(config-zone-pair-rule)# match destination-nat
rtt(config-zone-pair-rule)# action permit
rtt(config-zone-pair-rule)# enable
rtt(config-zone-pair-rule)# exit
rtt(config-zone-pair)# exit
rtt(config)# exit
```

Произведенные настройки можно посмотреть с помощью команд:

```
rtt# show ip nat destination pools
rtt# show ip nat destination rulesets
rtt# show ip nat proxy-arp
rtt# show ip nat translations
```

19.3. Конфигурирование Source NAT

Функция Source NAT (SNAT) используется для подмены адреса источника у пакетов, проходящих через сетевой шлюз. При прохождении пакетов из локальной сети в публичную сеть адрес источника заменяется на один из публичных адресов шлюза. Дополнительно к адресу источника может применяться замена порта источника. При прохождении пакетов из публичной сети в локальную происходит обратная подмена адреса и порта.

Функция SNAT может быть использована для предоставления доступа в Интернет компьютерам, находящимся в локальной сети. При этом не требуется назначения публичных IP-адресов этим компьютерам.

19.3.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Перейти в режим настройки сервиса трансляции адресов отправителя.	<code>rtt(config)# nat source</code>	
2	Создать пул IP-адресов и/или TCP/UDP-портов с определённым именем (не обязательно).	<code>rtt(config-snat)# pool <NAME></code>	<NAME> – имя пула NAT-адресов, задаётся строкой до 31 символа.
3	Установить диапазон IP-адресов, для которых будет заменяться IP-адрес отправителя.	<code>rtt(config-snat-pool)# ip address-range <IP>[-<ENDIP>]</code>	<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.
4	Задать диапазон внешних TCP/UDP-портов, на которые будет заменяться TCP/UDP-порт отправителя.	<code>rtt(config-snat-pool)# ip port-range <PORT>[-<ENDPORT>]</code>	<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.
5	Установить внешний TCP/UDP-порт, на который будет заменяться TCP/UDP-порт отправителя.	<code>rtt(config-snat-pool)# ip port <PORT></code>	<PORT> – TCP/UDP-порт, принимает значения [1..65535].
6	Включить функции NAT persistent.	<code>rtt(config-snat-pool)# persistent</code>	
7	Создать группу правил с определённым именем.	<code>rtt(config-snat)# ruleset <NAME></code>	<NAME> – имя группы правил, задаётся строкой до 31 символа.
8	Указать экземпляр VRF, в котором будет работать данная группа правил (не обязательно).	<code>rtt(config-snat-ruleset)# ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.

Шаг	Описание	Команда	Ключи
9	Задать область применения группы правил. Правила будут применяться только для трафика, идущего в определенную зону или интерфейс.	<pre>rtt(config-snat-ruleset)# to { zone <NAME> interface <IF> tunnel <TUN> default }</pre>	<p><NAME> – имя зоны изоляции;</p> <p><IF> – имя интерфейса устройства;</p> <p><TUN> – имя туннеля устройства</p> <p>default – обозначает группу правил для всего трафика, источник которого не попал под критерии других групп правил.</p>
10	Задать правило с определённым номером. Правила обрабатываются в порядке возрастания.	<pre>rtt(config-snat-ruleset)# rule <ORDER></pre>	<p><ORDER> – номер правила, принимает значения [1..10000].</p>
11	Задать профиль IP-адресов {отправителя получателя}, для которых должно срабатывать правило.	<pre>rtt(config-dnat-rule)# match [not] {source destination}-address { address-range { <ADDR>[-<ADDR>] } prefix { <ADDR/LEN> } object-group { network <OBJ-GROUP-NETWORK-NAME> } }</pre>	<p>address-range <ADDR>[-<ADDR>] – диапазон IP-адресов для правил NAT. Если не указывать IP-адрес конца диапазона, то в качестве IP-адреса для срабатывания правила используется только IP-адрес начала диапазона.</p> <p>Параметр задаётся в виде A.B.C.D, где каждая часть принимает значения [0..255];</p> <p>prefix <ADDR/LEN> – IP-подсеть, используемая для срабатывания правила NAT. Параметр задаётся в виде A.B.C.D/E, где каждая часть A – D принимает значения [0..255] и E принимает значения [1..32];</p> <p>object-group network <OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.</p>
12	Задать профиль IP-адресов {отправителя получателя}, для которых должно срабатывать правило (не обязательно).	<pre>rtt(config-snat-rule)# match [not] {source destination}-port <TYPE> {<PORT-SET-NAME> <FROM-PORT> - <TO-PORT>}</pre>	<p><TYPE> – тип аргумента, устанавливаемый в качестве адреса:</p> <ul style="list-style-type: none"> address-range – указать диапазон IPv4/IPv6 адресов; object-group – указать имя профиля; any – установить в качестве адреса любой адрес. <p><PORT-SET-NAME> – имя профиля порта, задаётся строкой до 31 символа;</p> <p><FROM-PORT> – начальный порт диапазона;</p> <p><TO-PORT> – конечный порт диапазона.</p>

Шаг	Описание	Команда	Ключи
13	Установить имя или номер IP-протокола, для которого должно срабатывать правило (не обязательно).	<code>rtt(config-snat-rule)# match [not] {protocol protocol-id} <TYPE></code>	<TYPE> – тип протокола, принимает значения: esp, icmp, icmp6, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rsvp, l2tp, gre. Значение «any» указывает на любой тип протокола; <ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].
14	Задать тип и код сообщений протокола ICMP, для которых должно срабатывать правило (не обязательно).	<code>rtt(config-snat-rule)# match [not] icmp {<ICMP_TYPE><ICMP_CODE> <TYPE-NAME>}</code>	<ICMP_TYPE> – тип сообщения протокола ICMP, принимает значения [0..255]; <ICMP_CODE> – код сообщения протокола ICMP, принимает значения [0..255]. Значение «any» указывает на любой код сообщения; <TYPE-NAME> – имя типа ICMP сообщения
15	Задать действие «трансляция адреса и порта отправителя» для трафика, удовлетворяющего критериям, заданным командами «match».	<code>rtt(config-snat-rule)# action source-nat { off pool <NAME> netmap <ADDR/LEN> [static] interface [FIRST_PORT – LAST_PORT] }</code>	off – трансляция отключена; pool<NAME> – имя пула, содержащего набор IP-адресов и/или TCP/UDP-портов; netmap <ADDR/LEN> – IP-адрес и маска подсети, используемые при трансляции; static – опция для организации статического NAT. Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – BBB принимает значения [0..255] и EE принимает значения [1..32]. interface [FIRST_PORT – LAST_PORT] – задаёт трансляцию в IP-адрес интерфейса. Если дополнительно задан диапазон TCP/UDP-портов, то трансляция будет происходить только для TCP/UDP-портов отправителя, входящих в указанный диапазон.
16	Активировать конфигурируемое правило.	<code>rtt(config-snat-rule)# enable</code>	

Шаг	Описание	Команда	Ключи
17	Включить функцию отслеживания сессий уровня приложений для протоколов FTP, SIP, H323, netbios-ns, PPTP (не обязательно).	<pre>rtt(config)# ip firewall sessions tracking {<PROTOCOL> sip [port <OBJECT-GROUP- SERVICE>] all}</pre>	<p>all – включает функцию отслеживания сессий уровня приложений для всех доступных протоколов</p> <p><PROTOCOL> – протокол уровня приложений, сессии которого должны отслеживаться, принимает значения [ftp, h323, pptp, netbios-ns].</p> <p><OBJECT-GROUP-SERVICE> – имя профиля TCP/UDP-портов sip-сессии, задаётся строкой до 31 символа. Если группа не указана, то отслеживание сессий sip будет осуществляться для порта 5060.</p>
18	Включить функцию трансляции IP-адресов в заголовках уровня приложений (не обязательно).	<pre>rtt(config)# nat alg {<PROTOCOL> all}</pre>	<p>all – включает трансляцию IP-адресов в заголовках всех доступных протоколов.</p> <p><PROTOCOL> – протокол уровня приложений, в заголовках которого должна работать трансляция адресов, принимает значения [ftp, h323, pptp, netbios-ns, gre, sip, tftp].</p>



При использовании ключа **not** правило будет срабатывать для значений, которые не входят в указанный профиль.

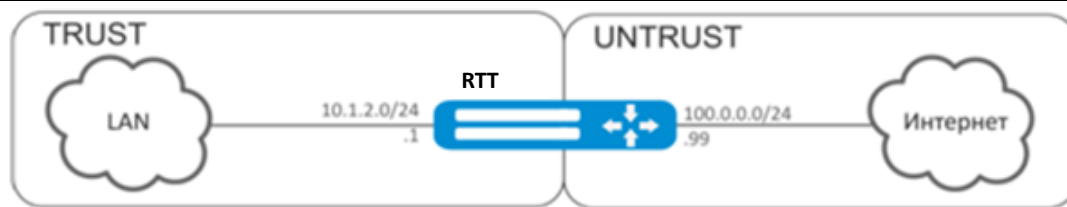
Каждая команда «**match**» может содержать ключ «**not**». При использовании данного ключа под правило будут подпадать пакеты, не удовлетворяющие заданному критерию.

Более подробная информация о командах для настройки маршрутизатора содержится в документе «Справочник команд CLI».

19.3.2. Пример настройки 1

Задача:

Настроить доступ пользователей локальной сети 10.1.2.0/24 к публичной сети с использованием функции Source NAT. Задать диапазон адресов публичной сети для использования SNAT 100.0.0.100-100.0.0.249.



Решение:

Конфигурирование начнем с создания зон безопасности, настройки сетевых интерфейсов и определения их принадлежности к зонам безопасности. Создадим доверенную зону «TRUST» для локальной сети и зону «UNTRUST» для публичной сети.

```

rtt# configure
rtt(config)# security zone UNTRUST
rtt(config-zone)# exit
rtt(config)# security zone TRUST
rtt(config-zone)# exit
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# ip address 10.1.2.1/24
rtt(config-if-gi)# security-zone TRUST
rtt(config-if-gi)# exit
rtt(config)# interface tengigabitethernet 1/0/1
rtt(config-if-te)# ip address 100.0.0.99/24
rtt(config-if-te)# security-zone UNTRUST
rtt(config-if-te)# exit
  
```

Для конфигурирования функции SNAT и настройки правил зон безопасности потребуется создать профиль адресов локальной сети «LOCAL_NET», включающий адреса, которым разрешен выход в публичную сеть, и профиль адресов публичной сети «PUBLIC_POOL».

```

rtt(config)# object-group network LOCAL_NET
rtt(config-object-group-network)# ip address-range 10.1.2.2-10.1.2.254
rtt(config-object-group-network)# exit
rtt(config)# object-group network PUBLIC_POOL
rtt(config-object-group-network)# ip address-range 100.0.0.100-100.0.0.249
rtt(config-object-group-network)# exit
  
```

Для пропуска трафика из зоны «TRUST» в зону «UNTRUST» создадим пару зон и добавим правила, разрешающие проходить трафику в этом направлении. Дополнительно включена проверка адреса источника данных на принадлежность к диапазону адресов «LOCAL_NET» для соблюдения ограничения на выход в публичную сеть. Действие правил разрешается командой **enable**.

```

rtt(config)# security zone-pair TRUST UNTRUST
rtt(config-zone-pair)# rule 1
rtt(config-zone-pair-rule)# match source-address object-group network LOCAL_NET
rtt(config-zone-pair-rule)# action permit
rtt(config-zone-pair-rule)# enable
rtt(config-zone-pair-rule)# exit
rtt(config-zone-pair)# exit
  
```

Конфигурируем сервис SNAT. Первым шагом создаётся пул адресов публичной сети, используемых для сервиса SNAT.

```

rtt(config)# nat source
rtt(config-snat)# pool TRANSLATE_ADDRESS
rtt(config-snat-pool)# ip address-range 100.0.0.100-100.0.0.249
rtt(config-snat-pool)# exit

```

Вторым шагом создаётся набор правил SNAT. В атрибутах набора укажем, что правила применяются только для пакетов, направляющихся в публичную сеть – в зону «UNTRUST». Правила включают проверку адреса источника данных на принадлежность к пулу «LOCAL_NET».

```

rtt(config-snat)# ruleset SNAT
rtt(config-snat-ruleset)# to zone UNTRUST
rtt(config-snat-ruleset)# rule 1
rtt(config-snat-rule)# match source-address object-group network LOCAL_NET
rtt(config-snat-rule)# action source-nat pool TRANSLATE_ADDRESS
rtt(config-snat-rule)# enable
rtt(config-snat-rule)# exit
rtt(config-snat-ruleset)# exit

```

Для того чтобы маршрутизатор отвечал на запросы протокола ARP для адресов, входящих в публичный пул, необходимо запустить сервис ARP Proxy. Сервис ARP Proxy настраивается на интерфейсе, которому принадлежит IP-адрес из подсети профиля адресов публичной сети «PUBLIC_POOL».

```

rtt(config)# interface tengigabitethernet 1/0/1
rtt(config-if-te)# ip nat proxy-arp PUBLIC_POOL

```

Для того чтобы устройства локальной сети могли получить доступ к публичной сети, на них должна быть настроена маршрутизация – адрес 10.1.2.1 должен быть назначен адресом шлюза.

На самом маршрутизаторе также должен быть создан маршрут для направления на публичную сеть. Этот маршрут может быть назначен маршрутом по умолчанию с помощью следующей команды.

```

rtt(config)# ip route 0.0.0.0/0 100.0.0.1
rtt(config)# exit

```

19.3.3. Пример настройки 2

Задача:

Настроить доступ пользователей локальной сети 21.12.2.0/24 к публичной сети с использованием функции Source NAT без использования межсетевого экрана (firewall). Диапазон адресов публичной сети для использования SNAT 200.10.0.100-200.10.0.249.



Решение:

```
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# ip address 21.12.2.1/24
rtt(config-if-gi)# ip firewall disable
rtt(config-if-gi)# exit
rtt(config)# interface tengigabitethernet 1/0/1
rtt(config-if-te)# ip address 200.10.0.1/24
rtt(config-if-te)# ip firewall disable
rtt(config-if-te)# exit
```

Для конфигурирования функции SNAT потребуется создать профиль адресов локальной сети «LOCAL_NET», включающий адреса, которым разрешен выход в публичную сеть, и профиль адресов публичной сети «PUBLIC_POOL»:

```
rtt(config)# object-group network LOCAL_NET
rtt(config-object-group-network)# ip address-range 21.12.2.2-21.12.2.254
rtt(config-object-group-network)# exit

rtt(config)# object-group network PUBLIC_POOL
rtt(config-object-group-network)# ip address-range 200.10.0.100-200.10.0.249
rtt(config-object-group-network)# exit
```

Конфигурируем сервис SNAT.

Первым шагом создаётся пул адресов публичной сети, используемых для сервиса SNAT:

```
rtt(config)# nat source
rtt(config-snat)# pool TRANSLATE_ADDRESS
rtt(config-snat-pool)# ip address-range 200.10.0.100-200.10.0.249
rtt(config-snat-pool)# exit
```

Вторым шагом создаётся набор правил SNAT. В атрибутах набора укажем, что правила применяются только для пакетов, направляющихся в публичную сеть через порт te1/0/1. Правила включают проверку адреса источника данных на принадлежность к пулу «LOCAL_NET»:

```
rtt(config-snat)# ruleset SNAT
rtt(config-snat-ruleset)# to interface te1/0/1
rtt(config-snat-ruleset)# rule 1
rtt(config-snat-rule)# match source-address object-group network LOCAL_NET
rtt(config-snat-rule)# action source-nat pool TRANSLATE_ADDRESS
rtt(config-snat-rule)# enable
rtt(config-snat-rule)# exit
rtt(config-snat-ruleset)# exit
```

Для того чтобы маршрутизатор отвечал на запросы протокола ARP для адресов, входящих в публичный пул, необходимо запустить сервис ARP Proxy. Сервис ARP Proxy настраивается на интерфейсе, которому принадлежит IP-адрес из подсети профиля адресов публичной сети «PUBLIC_POOL»:

```
rtt(config)# interface tengigabitethernet 1/0/1
rtt(config-if-te)# ip nat proxy-arp PUBLIC_POOL
```

Для того чтобы устройства локальной сети могли получить доступ к публичной сети, на них должна быть настроена маршрутизация – адрес 21.12.2.1 должен быть назначен адресом шлюза.

На самом маршрутизаторе также должен быть создан маршрут для направления на публичную сеть. Этот маршрут может быть назначен маршрутом по умолчанию с помощью следующей команды:

```
rtt(config)# ip route 0.0.0.0/0 200.10.0.254
rtt(config)# exit
```

19.4. Конфигурирование Static NAT

Static NAT — статический NAT задает однозначное соответствие одного адреса другому. Иными словами, при прохождении через маршрутизатор, адрес меняется на другой строго заданный адрес, один-к-одному. Запись о такой трансляции хранится неограниченно долго, пока не будет произведена перенастройка NAT на маршрутизаторе.

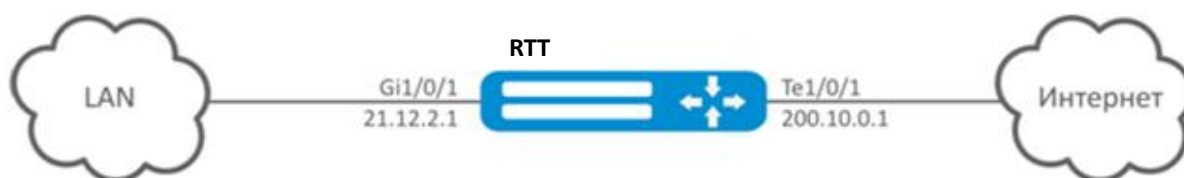
19.4.1. Алгоритм настройки

Настройка Static NAT осуществляется средствами Source NAT, алгоритм настройки которой описан в разделе **Конфигурирование Source NAT**, алгоритм настройки настоящего руководства.

19.4.2. Пример настройки Static NAT

Задача:

Настроить двухстороннюю и постоянную трансляцию из локальной сети для диапазона адресов 21.12.2.100-21.12.2.150 в публичную сеть 200.10.0.0/24. Диапазон адресов публичной сети для использования трансляции – 200.10.0.100-200.10.0.150.



Решение:

Начнем конфигурирование с настройки сетевых интерфейсов и отключения межсетевого экрана:

```
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# ip address 21.12.2.1/24
rtt(config-if-gi)# ip firewall disable
rtt(config-if-gi)# exit
rtt(config)# interface tengigabitethernet 1/0/1
rtt(config-if-te)# ip address 200.10.0.1/24
rtt(config-if-te)# ip firewall disable
rtt(config-if-te)# exit
```

Для конфигурирования Static NAT потребуется создать профиль адресов локальной сети «LOCAL_NET», включающий локальную подсеть:

```
rtt(config)# object-group network LOCAL_NET
rtt(config-object-group-network)# ip prefix 21.12.2.0/24
rtt(config-object-group-network)# exit
```

Диапазон адресов публичной сети для использования Static NAT задаем в профиле «PROXY»:

```
rtt(config)# object-group network PROXY
rtt(config-object-group-network)# ip address-range 200.10.0.100-200.10.0.150
rtt(config-object-group-network)# exit
```

Конфигурируем сервис Static NAT в режиме конфигурирования SNAT. В атрибутах набора укажем, что правила применяются только для пакетов, направляющихся в публичную сеть через порт te1/0/1. Правила включают проверку адреса источника данных на принадлежность к пулу «LOCAL_NET».

```
rtt(config)# nat source
rtt(config-snat)# ruleset SNAT
rtt(config-snat-ruleset)# to interface te1/0/1
rtt(config-snat-ruleset)# rule 1
rtt(config-snat-rule)# match source-address object-group network LOCAL_NET
rtt(config-snat-rule)# action source-nat netmap 200.10.0.0/24 static
rtt(config-snat-rule)# enable
rtt(config-snat-rule)# exit
rtt(config-snat-ruleset)# exit
```

Для того чтобы маршрутизатор отвечал на запросы протокола ARP для адресов, входящих в пул трансляции «PROXY», необходимо запустить сервис ARP Proxy. Сервис ARP Proxy настраивается на интерфейсе, которому принадлежит IP-адрес из подсети профиля адресов «PROXY».

```
rtt(config)# interface tengigabitethernet 1/0/1
rtt(config-if-te)# ip nat proxy-arp PROXY
```

Для того чтобы устройства локальной сети могли получить доступ к сети 200.10.0.0/24, на них должна быть настроена маршрутизация – адрес 21.12.2.1 должен быть назначен адресом шлюза.

Изменения конфигурации вступают в действие по команде применения.

```
rtt# commit
Configuration has been successfully committed
rtt# confirm
Configuration has been successfully confirmed
```

Посмотреть активные трансляции можно с помощью команды:

```
rtt# show ip nat translations
```

19.5. Настройка NTP

NTP (англ. Network Time Protocol — протокол сетевого времени) — сетевой протокол для синхронизации внутренних часов оборудования с использованием IP-сетей, использует для своей работы протокол UDP, учитывает время передачи и использует алгоритмы для достижения высокой точности синхронизации времени.

19.5.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Включить NTP.	<code>rtt(config)# ntp enable</code>	
2	Задать IP-адрес NTP-сервера, либо участника NTP-синхронизации.	<code>rtt(config)# ntp { pool server peer } { <IPv4> <NAME> IPv6 }</code>	<p><IPv4> – IP-адрес назначения (шлюз), задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p> <p><NAME> – DNS-имя сервера, задаётся строкой до 31 символа.</p> <p><IPv6> – IP-адрес назначения (шлюз), задаётся в виде X:X:X:X::X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF].</p>
3	Включить отправку нескольких пакетов вместо одного при установке соединения.	<code>rtt(config-ntp)# burst</code>	
4	Включить отправку нескольких пакетов вместо одного в случае разрыва соединения.	<code>rtt(config-ntp)# iburst</code>	
5	Задать ключ для аутентификации (не обязательно).	<code>rtt(config-ntp)# key <ID></code>	<ID> – идентификатор ключа, задается в диапазоне [1..255].
6	Установить максимальное значение интервала времени между отправкой сообщений NTP-серверу (не обязательно).	<code>rtt(config-ntp)# maxpoll <INTERVAL></code>	<p><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].</p> <p>Значение по умолчанию: 10 (2^{10} = 1024 секунды или 17 минут 4 секунды).</p>
7	Установить минимальное значение	<code>rtt(config-ntp)# minpoll <INTERVAL></code>	<INTERVAL> – минимальное значение интервала опроса в секундах вычисляется путем возведения

Шаг	Описание	Команда	Ключи
	интервала времени между отправкой сообщений NTP-серверу (не обязательно).		двойки в степень, заданную параметром команды, принимает значение [4..6]. Значение по умолчанию: 6 ($2^6 = 64$ секунды или 1 минута 4 секунды).
8	Отметить данный NTP-сервер как предпочтительный (не обязательно).	<code>rtt(config-ntp) # prefer</code>	
9	Определить список доверенных IP-адресов, с которыми может происходить обмен ntp-пакетами (не обязательно).	<code>rtt(config) # ntp access-addresses <NAME></code>	<NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.
10	Указать идентификатор ключа из профиля связки ключей (не обязательно).	<code>rtt(config) # ntp authentication trusted-key <ID></code>	<ID> – идентификатор ключа из профиля связки ключей.
11	Указать имя профиля связки ключей (не обязательно).	<code>rtt(config) # ntp authentication key-chain <WORD></code>	<WORD> – имя профиля связки ключей.
12	Активировать аутентификацию для NTP по ключу (не обязательно).	<code>rtt(config) # ntp authentication enable</code>	
13	Включить режим приёма широковещательных сообщений NTP-серверов для глобальной конфигурации и всех существующих VRF (не обязательно).	<code>rtt(config) # ntp broadcast-client enable</code>	
14	Задать значение кода DSCP для использования в IP-заголовке исходящих пакетов NTP-сервера (не обязательно).	<code>rtt(config) # ntp dscp <DSCP></code>	<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63]. Значение по умолчанию: 46.
15	Включить режим query-only, ограничивающий взаимодействие по NTP для определенного	<code>rtt(config) # ntp object-group query-only <NAME></code>	<NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

Шаг	Описание	Команда	Ключи
	профиля IP-адресов (не обязательно).		
16	Включить режим serve-only, ограничивающий взаимодействие по NTP для определенного профиля IP-адресов (не обязательно).	<code>rtt(config)# ntp object-group serve-only <NAME></code>	<NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.
17	Указать source-IP-адреса для NTP-пакетов для всех peer (не обязательно).	<code>rtt(config)# ntp source address <ADDR></code>	<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
18	Задать текущие время и дату в ручном режиме (не обязательно).	<code>rtt# set date <TIME> [<DAY> <MONTH> [<YEAR>]]</code>	<p><TIME> – устанавливаемое системное время, задаётся в виде HH:MM:SS, где:</p> <ul style="list-style-type: none"> • 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]; <p><MONTH> – месяц, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];</p> <p><YEAR> – год, принимает значения [2001..2037].</p>

19.5.2. Пример настройки

Задача:

Настроить синхронизацию времени от NTP-сервера.

IP-адрес маршрутизатора RTT – 192.168.52.8,

IP-адрес NTP-сервера – 192.168.52.41.



Решение:

Предварительно нужно выполнить следующие действия:

- указать зону безопасности для интерфейса gi1/0/1;
- настроить IP-адрес для интерфейса gi1/0/1, чтобы обеспечить IP-связность с NTP-сервером.

Пример:

```
security zone untrust
exit
object-group service NTP
  port-range 123
exit
interface gigabitethernet 1/0/1
  security-zone untrust
  ip address 192.168.52.8/24
exit
security zone-pair untrust self
  rule 10
    action permit
    match protocol udp
    match destination-port object-group NTP
    enable
  exit
exit
```

Основной этап конфигурирования:

Включение синхронизации системных часов с удаленными серверами:

```
rtt(config)# ntp enable
```

Настройка NTP-сервера:

```
rtt(config)# ntp server 192.168.52.41
```

Указать предпочтительность данного NTP-сервера (необязательно):

```
rtt(config-ntp)# prefer
```

Указать интервал времени между отправкой сообщений NTP-серверу:

```
rtt(config-ntp)# minpoll 4
rtt(config-ntp)# end
rtt# commit
rtt# confirm
```

Команда для просмотра текущей конфигурации протокола NTP:

```
rtt# show ntp configuration
```

Команда для просмотра текущего состояния NTP-серверов (пиров):

```
rtt# show ntp peers
```

20. МОНИТОРИНГ

20.1. Настройка Netflow

Netflow — сетевой протокол, предназначенный для учета и анализа трафика. Netflow позволяет передавать данные о трафике (адрес отправителя и получателя, порт, количество информации и др.) с сетевого оборудования (сенсора) на коллектор. В качестве коллектора может использоваться обычный сервер.



В текущей реализации трафик, отброшенный маршрутизатором по каким-либо причинам, не будет учитываться в статистике.

20.1.1. Алгоритм настройки

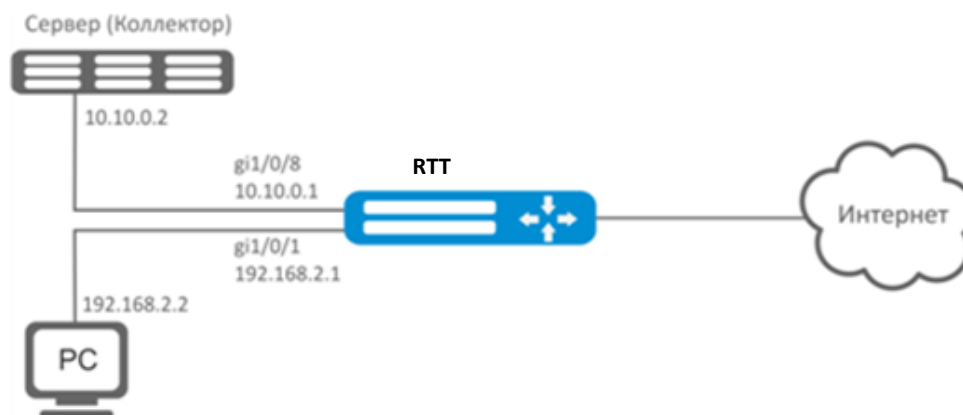
Шаг	Описание	Команда	Ключи
1	Задать версию Netflow-протокола.	<code>rtt(config)# netflow version <VERSION></code>	<VERSION> – версия Netflow-протокола: 5, 9 и 10.
2	Установить максимальное количество наблюдаемых сессий.	<code>rtt(config)# netflow max-flows <COUNT></code>	<COUNT> – количество наблюдаемых сессий, принимает значение [10000..2000000]. Значение по умолчанию: 512000.
3	Установить интервал, по истечении которого информация об активных сессиях экспортируются на коллектор.	<code>rtt(config)# netflow active-timeout <TIMEOUT></code>	<TIMEOUT> – интервал времени, по истечении которого информация об активных сессиях экспортируются на коллектор, задается в секундах, принимает значение [5..36000]. Значение по умолчанию: 1800 секунд.
4	Установить интервал, по истечении которого информация об устаревших сессиях экспортируются на коллектор.	<code>rtt(config)# netflow inactive-timeout <TIMEOUT></code>	<TIMEOUT> – задержка перед отправкой информации об устаревших сессиях, задается в секундах, принимает значение [0..240]. Значение по умолчанию: 15 секунд.
5	Установить частоту отправки статистики на Netflow-коллектор.	<code>rtt(config)# netflow refresh-rate <RATE></code>	<RATE> – частота отправки статистики, задается в пакетах на поток, принимает значение [1..10000]. Значение по умолчанию: 10.
6	Активировать Netflow на маршрутизаторе.	<code>rtt(config)# netflow enable</code>	

Шаг	Описание	Команда	Ключи
7	Создать коллектор Netflow и перейти в режим его конфигурирования.	rtt(config)# netflow collector <ADDR>	<ADDR> – IP-адрес коллектора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
8	Установить порт Netflow-сервиса на сервере сбора статистики.	rtt(config-netflow-host)# port <PORT>	<PORT> – номер UDP-порта, указывается в диапазоне [1..65535]. Значение по умолчанию: 2055.
9	Включить отправку статистики на Netflow-сервер в режим конфигурирования интерфейса/туннеля/сетевого моста.	rtt(config-if-gi)# ip netflow export	

20.1.2. Пример настройки

Задача:

Организовать учет трафика с интерфейса gi1/0/1 для передачи на сервер через интерфейс gi1/0/8 для обработки.



Решение:

Предварительно необходимо настроить адресацию на интерфейсах.

Основной этап конфигурирования:

Укажем IP-адрес коллектора:

```
rtt(config)# netflow collector 10.10.0.2
```

Включим сбор экспорта статистики Netflow на сетевом интерфейсе gi1/0/1:

```

rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# ip netflow export

```

Активируем Netflow на маршрутизаторе:

```

rtt(config)# netflow enable

```

Для просмотра статистики Netflow используется команда:

```

rtt# show netflow statistics

```

Настройка Netflow для учета трафика между зонами аналогична настройке sFlow, описание приведено в разделе Настройка sFlow.

20.2. Настройка sFlow

sFlow — стандарт для мониторинга компьютерных сетей, беспроводных сетей и сетевых устройств, предназначенный для учета и анализа трафика.

20.2.1. Алгоритм настройки

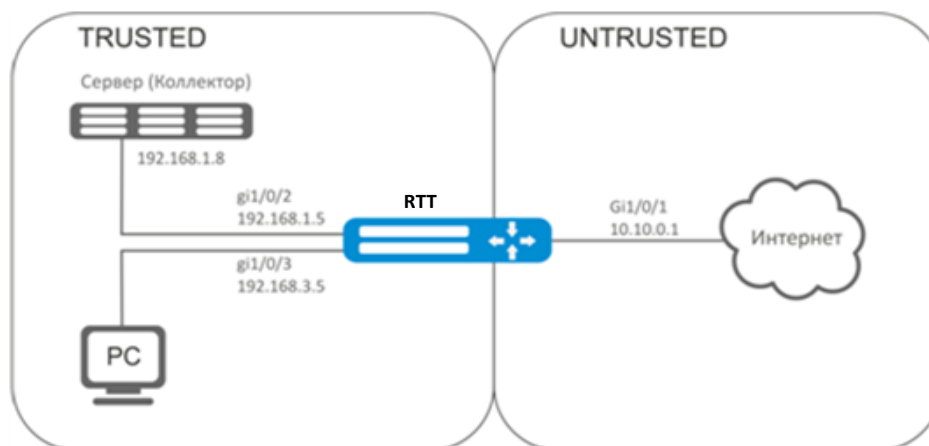
Шаг	Описание	Команда	Ключи
1	Установить частоту отправки пакетов пользовательского трафика в неизменном виде на sFlow-коллектор.	<code>rtt(config)# sflow sampling- rate <RATE></code>	<RATE> – частота отправки пакетов пользовательского трафика на коллектор, принимает значение [1..65535]. При значении частоты 10 на коллектор будет отправлен один пакет из десяти. Значение по умолчанию: 1000.
2	Установить интервал, по истечении которого происходит получение информации о счетчиках сетевого интерфейса.	<code>rtt(config)# sflow poll- interval <TIMEOUT></code>	<TIMEOUT> – интервал, по истечении которого происходит получение информации о счетчиках сетевого интерфейса, принимает значение [1..300] секунд. Значение по умолчанию: 10 секунд.
3	Создать коллектор sFlow и перейти в режим его конфигурирования.	<code>rtt(config)# sflow collector <ADDR> [vrf <VRF>]</code>	<ADDR> – IP-адрес коллектора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]; <VRF> – имя экземпляра VRF, задаётся строкой до 31 символа.
4	Указать порт sFlow-коллектора (необязательно).	<code>rtt(config- sflow-host)# port <PORT></code>	<PORT> – номер UDP-порта, указывается в диапазоне [1..65535].

Шаг	Описание	Команда	Ключи
5	Установить адрес sFlow-агента (необязательно).	<code>rtt(config)# sflow agent-ip <ADDR></code>	<ADDR> – IPv4/IPv6-адрес агента sFlow. Если команда не указана, то в качестве адреса агента будет использован случайный адрес из присутствующих в конфигурации.
6	Активировать сервис sFlow на маршрутизаторе.	<code>rtt(config)# sflow enable</code>	
7	В режиме конфигурирования интерфейса/туннеля/сетевого моста включить отправку статистики sFlow.	<code>rtt(config-if- gi)# ip sflow export</code>	

20.2.2. Пример настройки

Задача:

Организовать учет трафика между зонами trusted и untrusted.



Решение:

Для сетей RTT создадим две зоны безопасности:

```
rtt# configure
rtt(config)# security zone TRUSTED
rtt(config-zone)# exit
rtt(config)# security zone UNTRUSTED
rtt(config-zone)# exit
```

Настроим сетевые интерфейсы и определим их принадлежность к зонам безопасности:

```
rtt(config)# interface gi1/0/1
rtt(config-if-gi)# security-zone UNTRUSTED
rtt(config-if-gi)# ip address 10.10.0.1/24
```

```

rtt(config-if-gi)# exit
rtt(config)# interface gi1/0/2-3
rtt(config-if-gi)# security-zone TRUSTED
rtt(config-if-gi)# exit
rtt(config)# interface gi1/0/2
rtt(config-if-gi)# ip address 192.168.1.5/24
rtt(config-if-gi)# exit
rtt(config)# interface gi1/0/3
rtt(config-if-gi)# ip address 192.168.3.5/24
rtt(config-if-gi)# exit

```

Укажем IP-адрес коллектора:

```

rtt(config)# sflow collector 192.168.1.8

```

Включим экспорт статистики по протоколу sFlow для любого трафика в правиле «rule1» для направления TRUSTED-UNTRUSTED:

```

rtt(config)# security zone-pair TRUSTED UNTRUSTED
rtt(config-zone-pair)# rule 1
rtt(config-zone-pair-rule)# action sflow-sample
rtt(config-zone-pair-rule)# match protocol any
rtt(config-zone-pair-rule)# match source-address any
rtt(config-zone-pair-rule)# match destination-address any
rtt(config-zone-pair-rule)# enable

```

Активируем sFlow на маршрутизаторе:

```

rtt(config)# sflow enable

```

Настройка sFlow для учета трафика с интерфейса осуществляется аналогично настройке Netflow.

20.3. Настройка SNMP

SNMP (англ. Simple Network Management Protocol — простой протокол сетевого управления) — протокол, предназначенный для управления устройствами в IP-сетях на основе архитектур TCP/UDP. SNMP предоставляет данные для управления в виде переменных, описывающих конфигурацию управляемой системы.

20.3.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Включить SNMP-сервер.	<code>rtt(config)# snmp-server</code>	

Шаг	Описание	Команда	Ключи
2	Определить community для доступа по протоколу SNMPv2c.	<pre> rtt(config)# snmp- server community <COMMUNITY> [<TYPE>] [{ <IP-ADDR> <IPv6- ADDR> }] [client-list <OBJ- GROUP-NETWORK-NAME>] [<VERSION>] [view <VIEW-NAME>] [vrf <VRF>] </pre>	<p><COMMUNITY> – сообщество для доступа по протоколу SNMP;</p> <p><TYPE> – уровень доступа:</p> <ul style="list-style-type: none"> • ro – доступ только для чтения; • rw – доступ для чтения и записи. <p><IP-ADDR> – IP-адрес клиента, которому предоставлен доступ, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><IPv6-ADDR> – IPv6-адрес клиента, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];</p> <p><OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, от которых обрабатываются snmp-запросы, задаётся строкой до 31 символа;</p> <p><VERSION> – версия snmp, поддерживаемая данным community, принимает значения v1 или v2c;</p> <p><VIEW-NAME> – имя профиля SNMP view, задаётся строкой до 31 символа;</p> <p><VRF> – имя экземпляра VRF, из которого будет разрешен доступ, задается строкой до 31 символа.</p>
3	Устанавливает значение переменной SNMP, содержащей контактную информацию.	<pre> rtt(config)# snmp- server contact <CONTACT> </pre>	<p><CONTACT> – контактная информация, задается строкой до 255 символов.</p>
4	Установить значение кода DSCP для использования в IP-заголовке исходящих пакетов SNMP-сервера (не обязательно).	<pre> rtt(config)# snmp- server dscp <DSCP> </pre>	<p><DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].</p> <p>Значение по умолчанию: 63.</p>
5	Создать SNMPv3-пользователь.	<pre> rtt(config)# snmp- server user <NAME> </pre>	<p><NAME> – имя пользователя, задаётся строкой до 31 символа.</p>

Шаг	Описание	Команда	Ключи
6	Устанавливает значение переменной SNMP, содержащей информацию о расположении оборудования.	<code>rtt(config)# snmp-server location <LOCATION></code>	<LOCATION> – информация о расположении оборудования, задается строкой до 255 символов.
7	Определить уровень доступа пользователя по протоколу SNMPv3.	<code>rtt(config-snmp-user)# access <TYPE></code>	<TYPE> – уровень доступа: <ul style="list-style-type: none"> • ro – доступ только для чтения; • rw – доступ для чтения и записи.
8	Определить режим безопасности пользователя по протоколу SNMPv3.	<code>rtt(config-snmp-user)# authentication access <TYPE></code>	<TYPE> – режим безопасности: <ul style="list-style-type: none"> • auth – используется только аутентификация; • priv – используется аутентификация и шифрование данных.
9	Определить алгоритм аутентификации SNMPv3-запросов.	<code>rtt(config-snmp-user)# authentication algorithm <ALGORITHM></code>	<ALGORITHM> – алгоритм шифрования: <ul style="list-style-type: none"> • md 5 – пароль шифруется по алгоритму md5; • sha 1 – пароль шифруется по алгоритму sha1.
10	Установить пароль для аутентификации SNMPv3-запросов.	<code>rtt(config-snmp-user)# authentication key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<CLEAR-TEXT> – пароль, задается строкой от 8 до 16 символов; <ul style="list-style-type: none"> • 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...).
11	Активировать фильтрацию и установить профиль IP-адресов, с которых могут приниматься SNMPv3-пакеты с данным именем SNMPv3-пользователя.	<code>rtt(config-snmp-user)# client-list <NAME></code>	<NAME> – имя ранее созданной object-group, задается строкой до 31 символа.

Шаг	Описание	Команда	Ключи
12	Указать vrf для SNMPv3-пользователя (не обязательно).	<code>rtt(config-snmp-user) # ip vrf forwarding <VRF></code>	<VRF> – имя экземпляра VRF, из которого будет разрешен доступ, задается строкой до 31 символа.
13	Активировать фильтрацию и установить IPv4/IPv6-адрес, которому предоставлен доступ к маршрутизатору под данным SNMPv3-пользователем.	<code>rtt(config-snmp-user) # ip address <ADDR></code>	<ADDR> – IP-адрес клиента, которому предоставлен доступ, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		<code>rtt(config-snmp-user) # ipv6 address <ADDR></code>	<IPv6-ADDR> – IPv6-адрес клиента, задается в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].
14	Активировать SNMPv3-пользователя.	<code>rtt(config-snmp-user) # enable</code>	Значение по умолчанию: процесс выключен.
15	Определить алгоритм шифрования передаваемых данных.	<code>rtt(config-snmp-user) # privacy algorithm <ALGORITHM></code>	<p><ALGORITHM> – алгоритм шифрования:</p> <ul style="list-style-type: none"> • aes 128 – использовать алгоритм шифрования AES-128; • des – использовать алгоритм шифрования DES.
16	Установить пароль для шифрования передаваемых данных.	<code>rtt(config-snmp-user) # privacy key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED- TEXT> }</code>	<p><CLEAR-TEXT> – пароль, задается строкой от 8 до 16 символов;</p> <p><ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).</p>
17	Установить профиль snmp view, позволяющий разрешать или запрещать доступ к тем или иным OID для user.	<code>rtt(config-snmp-user) # view <VIEW-NAME></code>	<VIEW-NAME> – имя SNMP view профиля, на основании которого обеспечивается доступ к OID, задается строкой до 31 символа.

Шаг	Описание	Команда	Ключи
18	Включить передачу SNMP-уведомлений на указанный IP-адрес и перейти в режим настройки SNMP-уведомлений.	<pre>rtt(config)# snmp-server host { <IP-ADDR> <IPv6-ADDR> } [vrf <VRF>]</pre>	<p><IP-ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p> <p><IPv6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];</p> <p><VRF> – имя экземпляра VRF, в котором находится коллектор SNMP-уведомлений, задается строкой до 31 символа.</p>
19	Определить порт коллектора SNMP-уведомлений на удаленном сервере (не обязательно).	<pre>rtt(config-snmp-host)# port <PORT></pre>	<p><PORT> – номер UDP-порта, указывается в диапазоне [1..65535].</p> <p>Значение по умолчанию: 162.</p>
20	Установить IP-адрес для отправки уведомлений на удаленный сервер.	<pre>rtt(config-snmp-host)# source-address { <ADDR> <IPv6-ADDR> object-group <NETWORK_OBJ_GROUP_NAME> }</pre>	<p><ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><IPv6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];</p> <p><NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве source address.</p> <p>Значение по умолчанию: IPv4/IPv6 – адрес интерфейса, ближайшего к удаленному SNMP-серверу.</p>
21	Установить интерфейс или туннель маршрутизатора, IPv4/IPv6-адрес которого будет использоваться для отправки уведомлений на удаленный сервер.	<pre>rtt(config-snmp-host)# source-interface { <IF> <TUN> }</pre>	<p><IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;</p> <p><TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.</p>

Шаг	Описание	Команда	Ключи
22	Разрешить отправку SNMP-уведомлений различных типов.	<code>rtt(config)# snmp-server enable traps <TYPE></code>	<p><TYPE> – тип фильтруемых сообщений. Может принимать значения:</p> <p>config, entry, entry-sensor, environment, envmon, files-operations, flash, flash-operations, interfaces, links, ports, screens, snmp, syslog.</p> <p>Дополнительные параметры зависят от типа фильтра. См. справочник команд CLI.</p>
23	Создать профиль snmp view, позволяющий разрешать или запрещать доступ к тем или иным OID для community (SNMPv2) и user (SNMPv3).	<code>rtt(config)# snmp-server view <VIEW-NAME></code>	<p><VIEW-NAME> – имя профиля SNMP view, задаётся строкой до 31 символа.</p>

20.3.2. Пример настройки

Задача:

Настроить SNMPv3-сервер с аутентификацией и шифрованием данных для пользователя admin. IP-адрес маршрутизатора RTT – 192.168.52.8, IP-адрес сервера – 192.168.52.41.



Решение:

Предварительно нужно выполнить следующие действия:

- указать зону для интерфейса gi1/0/1;
- настроить IP-адрес для интерфейсов gi1/0/1.

Основной этап конфигурирования:

Включаем SNMP-сервер:

```
rtt(config)# snmp-server
```

Создаем пользователя SNMPv3:

```
rtt(config)# snmp-server user admin
```

Определим режим безопасности:

```
rtt(snmp-user) # authentication access priv
```

Определим алгоритм аутентификации для SNMPv3-запросов:

```
rtt(snmp-user) # authentication algorithm md5
```

Установим пароль для аутентификации SNMPv3-запросов:

```
rtt(snmp-user) # authentication key ascii-text 123456789
```

Определим алгоритм шифрования передаваемых данных:

```
rtt(snmp-user) # privacy algorithm aes128
```

Установим пароль для шифрования передаваемых данных:

```
rtt(snmp-user) # privacy key ascii-text 123456789
```

Активируем SNMPv3-пользователя:

```
rtt(snmp-user) # enable
```

Определяем сервер-приемник Trap-PDU-сообщений:

```
rtt(config) # snmp-server host 192.168.52.41
```

20.4. Настройка Zabbix-agent/proxy

Zabbix-agent — агент, предназначенный для выполнения удаленных команд с Zabbix-сервера. Агент может работать в двух режимах: пассивный и активный. Для работы в пассивном режиме, по умолчанию, необходимо разрешающее правило в firewall — протокол tcp, порт 10050. Для активного режима — протокол tcp, порт 10051.

Zabbix-прокси — это сервис, способный собирать данные мониторинга с одного или нескольких наблюдаемых устройств и отправлять эту информацию Zabbix-серверу.

Текущая версия установленного агента (прокси) — 6.0.39.

20.4.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Перейти в контекст настройки агента/проxy.	<code>rtt(config-zabbix-agent)# zabbix-agent</code> <code>rtt(config-zabbix-proxy)# zabbix-proxy</code>	
2	Указать имя узла сети (опционально). Для активного режима имя должно совпадать с именем узла сети на Zabbix-сервере.	<code>rtt(config-zabbix-agent)# hostname <WORD></code> <code>rtt(config-zabbix-proxy)# hostname <WORD></code>	<WORD> – имя узла сети, задается строкой до 255 символов.
3	Указать адрес Zabbix-сервера.	<code>rtt(config-zabbix-agent)# server <ADDR></code> <code>rtt(config-zabbix-proxy)# server <ADDR></code>	<ADDR> – IP-адрес сервера, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
4	Указать адрес сервера для активных проверок (при использовании активного режима).	<code>rtt(config-zabbix-agent)# active-server <ADDR> <PORT></code> <code>rtt(config-zabbix-proxy)# active-server <ADDR> <PORT></code>	<ADDR> – IP-адрес сервера, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. <PORT> – порт сервера, задается в диапазоне [1..65535]. Значение по умолчанию 10051.
5	Указать порт, который будет слушать агент/прокси (не обязательно).	<code>rtt(config-zabbix-agent)# port <PORT></code> <code>rtt(config-zabbix-proxy)# port <PORT></code>	<PORT> – порт, который слушает zabbix-агент/прокси, задается в диапазоне [1..65535]. Значение по умолчанию: 10050.
6	Разрешить выполнение удаленных команд zabbix-агентом (при использовании активного режима).	<code>rtt(config-zabbix-agent)# remote-commands</code>	
7	Указать адрес, с которого будет осуществляться взаимодействие с сервером (не обязательно).	<code>rtt(config-zabbix-agent)# source-address <ADDR></code> <code>rtt(config-zabbix-proxy)# source-address <ADDR></code>	<ADDR> – IP-адрес сервера, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Значение по умолчанию: ближайший адрес по маршрутизации.

Шаг	Описание	Команда	Ключи
8	Указать максимальное время на обработку удаленных команд (не обязательно).	<code>rtt(config-zabbix-agent)# timeout <TIME></code> <code>rtt(config-zabbix-proxy)# timeout <TIME></code>	<TIME> – время ожидания, определяется в секундах [1..30]. Значение по умолчанию 3. Рекомендуется устанавливать максимальное значение, т. к. некоторые команды могут выполняться дольше значения по умолчанию. Если за указанное время команда не будет выполнена, то обработка команды будет прекращена.
9	Указать место хранения базы данных для Zabbix-proxy (не обязательно).	<code>rtt(config-zabbix-proxy)# database <PATH></code>	<PATH> – место хранения базы данных Zabbix-proxy. По умолчанию база данных Zabbix хранится в энергонезависимой памяти маршрутизатора.
10	Указать интервал запроса конфигурации от Zabbix-сервера (не обязательно).	<code>rtt(config-zabbix-proxy)# config-retrieve <TIME></code>	<TIME> – время между опросами в секундах, принимает значения [1..604800]. Значение по умолчанию: 60.
11	Включить функционал агента/прокси.	<code>rtt(config-zabbix-agent)# enable</code> <code>rtt(config-zabbix-proxy)# enable</code>	
12	Разрешить из соответствующей зоны безопасности firewall обращение к маршрутизатору (в зону self) по TCP-портам 10050, 10051. См. раздел Конфигурирование Firewall .		

20.4.2. Пример настройки zabbix-agent



Задача:

Настроить взаимодействие между агентом и сервером для выполнения удаленных команд с сервера.

Решение:

В контексте настройки агента укажем адрес Zabbix-сервера и адрес, с которого будет осуществляться взаимодействие с сервером:

```
rtt(config-zabbix-agent)# server 192.168.32.101
```

```
rtt(config-zabbix-agent) # source-address 192.168.39.170
```

Для активации активного режима укажем hostname, active-server, а также включим выполнение удаленных команд:

```
rtt(config-zabbix-agent) # hostname RTT-agent
rtt(config-zabbix-agent) # active-server 192.168.32.101
rtt(config-zabbix-agent) # remote-commands
```

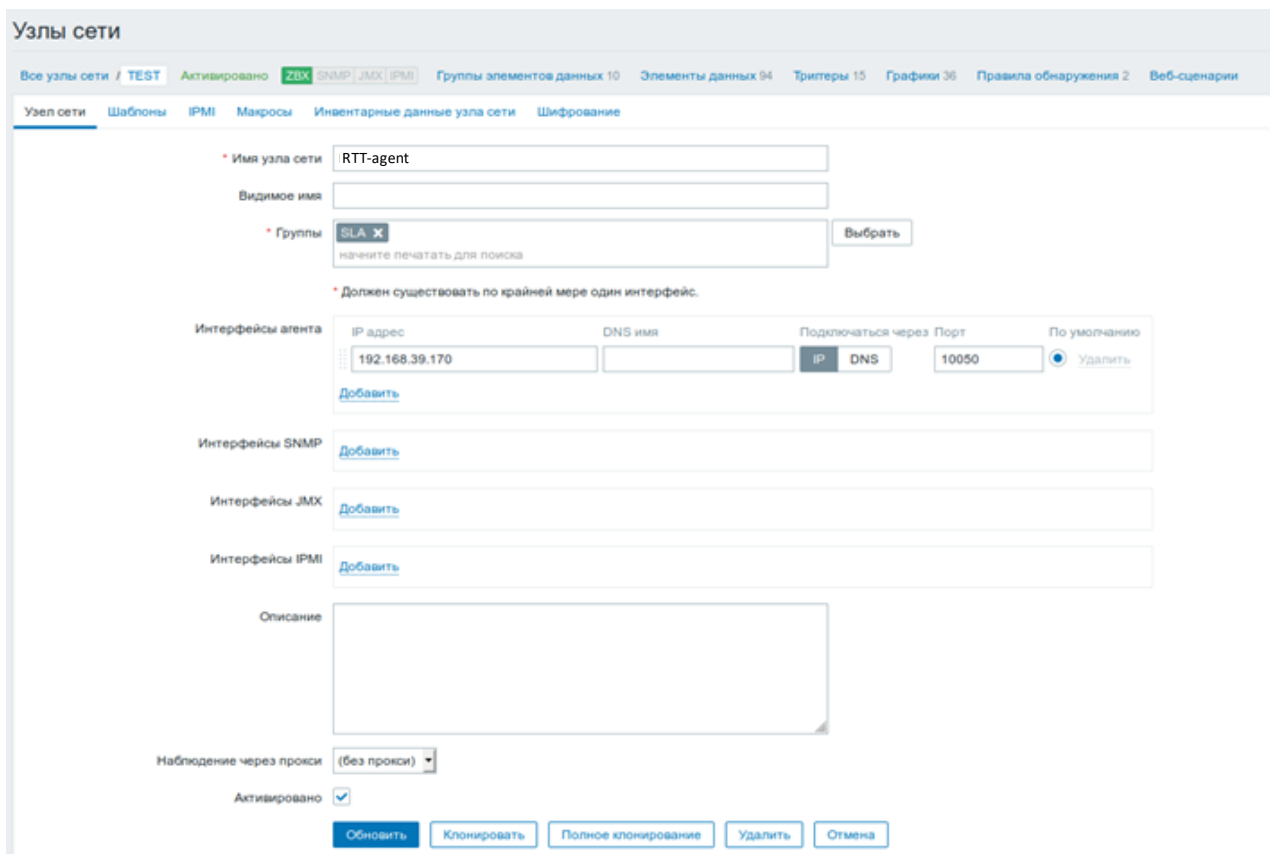
Зададим время выполнения удаленных команд и активируем функционал агента:

```
rtt(config-zabbix-agent) # timeout 30
rtt(config-zabbix-agent) # enable
```

20.4.3. Пример настройки zabbix-server

Перед настройкой необходимо убедиться, что сервер и агент используют синхронизированное время UTC с учетом локальных часовых поясов.

Создадим узел сети:



Узлы сети

Все узлы сети / **TEST** | Активировано | **ZBX** | SNMP | JMX | IPMI | Группы элементов данных 10 | Элементы данных 94 | Триггеры 15 | Графики 36 | Правила обнаружения 2 | Веб-сценарии

Узел сети | Шаблоны | IPMI | Макросы | Инвентарные данные узла сети | Шифрование

* Имя узла сети: RTT-agent

Видимое имя:

* Группы: SLA. ✕ | Выбрать

начните печатать для поиска

* Должен существовать по крайней мере один интерфейс.

Интерфейсы агента

IP адрес	DNS имя	Подключаться через	Порт	По умолчанию
192.168.39.170		IP	10050	<input checked="" type="radio"/> Удалить

[Добавить](#)

Интерфейсы SNMP: [Добавить](#)

Интерфейсы JMX: [Добавить](#)

Интерфейсы IPMI: [Добавить](#)

Описание:

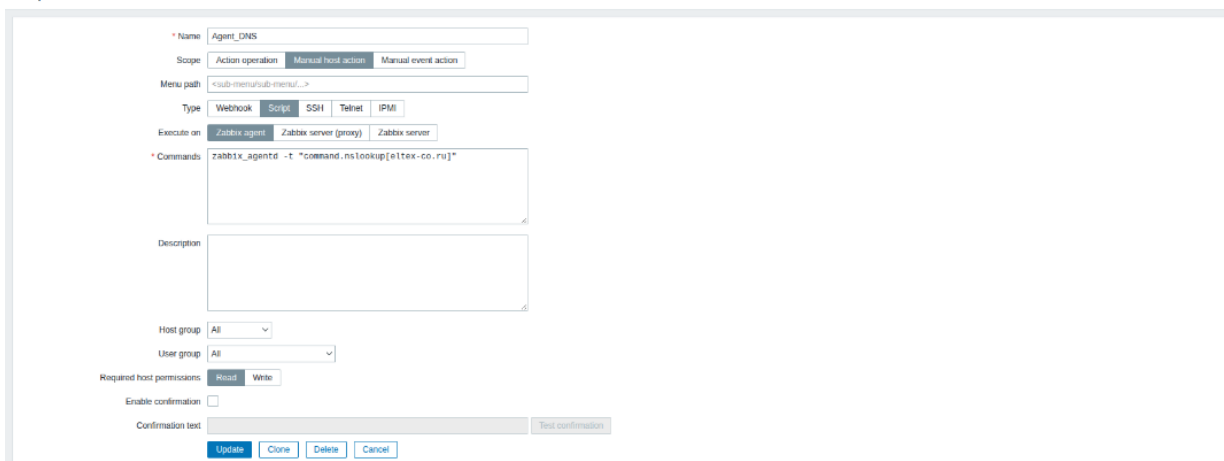
Наблюдение через прокси: (без прокси)

Активировано: ☒

[Обновить](#) [Клонировать](#) [Полное клонирование](#) [Удалить](#) [Отмена](#)

Создадим скрипт (Администрирование -> Скрипты -> Создать скрипт)

Scripts



Маршрутизаторы RTT поддерживают выполнение следующих привилегированных команд:

Ping

```
zabbix_agentd -t "command.ping[ domain.local -c 15]"
```

Клиент (RTT), получивший данную команду от сервера, выполнит ping до заданного узла и вернет результат серверу.



Использование ключа "-с" с указанием количества пакетов в тесте – обязательно. Без данного ключа команда ping не остановится самостоятельно и тест не будет считаться завершенным.

Ping в VRF

```
zabbix_agentd -t "command.ping_vrf[Backup, -c 15]"
```

Вышеупомянутая команда будет выполнена в заданном VRF с именем "Backup".

Fping

```
zabbix_agentd -t "command.fping[192.168.32.101]"
```

Клиент (RTT), получивший данную команду от сервера, выполнит **fping** до заданного узла (в нашем примере до 192.168.32.101) и вернет результат серверу.

Fping в VRF

```
zabbix_agentd -t "command.fping_vrf[Backup, domain.local]"
```

Команда будет выполнена в заданном VRF с именем "Backup".

Traceroute

```
zabbix_agentd -t "command.traceroute[192.168.32.101]"
```

Клиент (RTT), получивший данную команду от сервера, выполнит `traceroute` до заданного узла (в нашем примере до 192.168.32.101) и вернет результат серверу.

Traceroute в VRF

```
zabbix_agentd -t "command.traceroute_vrf[VRF, 192.168.32.101]"
```

Iperf

```
zabbix_agentd -t "command.iperf[-c 192.168.32.101 -u -t 5 -i 1]"
```

Клиент (RTT), получивший данную команду от сервера, выполнит `iperf` до заданного сервера (в нашем примере до 192.168.32.101) и вернет результат серверу.

Iperf3

```
zabbix_agentd -t "command.iperf3[-c 192.168.32.101 -t 5 -i 1]"
```

Iperf в VRF

```
zabbix_agentd -t "command.iperf_vrf[VRF, -c 192.168.32.101 -t 5 -i 1]"
```

Iperf3 в VRF

```
zabbix_agentd -t "command.iperf3_vrf[VRF,-c 192.168.32.101 -t 5 -i 1 ]"
```

Nslookup

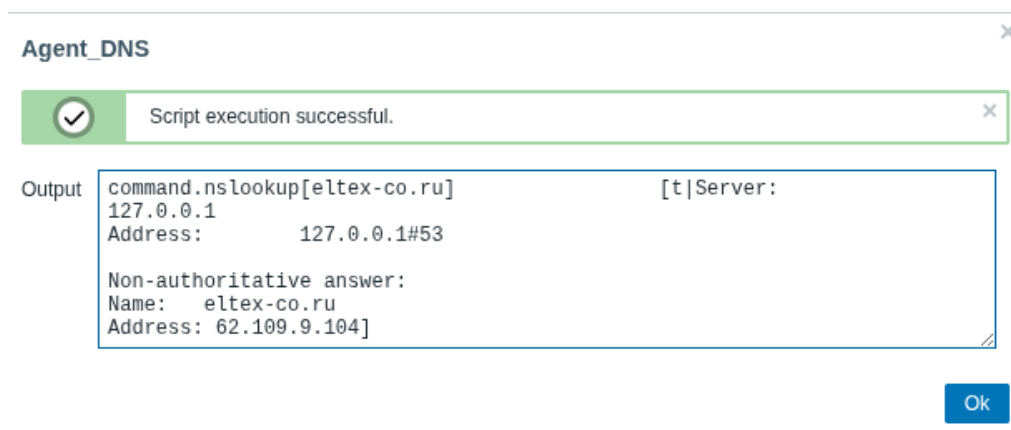
```
zabbix_agentd -t "command.nslookup[domain_name.local]"
```

Клиент (RTT), получивший данную команду от сервера, выполнит **nslookup** и вернет результат серверу.

Nslookup в VRF

```
zabbix_agentd -t "command.nslookup_vrf[VRF, domain_name.local]"
```

Пример выполнения команды **nslookup**:



20.5. Настройка Syslog

Syslog (англ. System Log — системный журнал) — стандарт отправки и регистрации сообщений о происходящих в системе событиях, используется в сетях, работающих по протоколу IP.

20.5.1. Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Включить от отправку syslog-сообщений на snmp-сервер в виде snmp-trap.	<code>rtt(config)# syslog snmp</code>	
2	Активировать или деактивировать отправку на snmp-сервер событий работы отдельных процессов маршрутизатора (не обязательно).	<code>rtt(config-syslog-snmp)# match [not] process-name <PROCESS-NAME></code>	<p><PROCESS-NAME> – см. в справочнике команд CLI.</p> <p>Если описаны разрешающие критерии (match process-name) – логируются только сообщения указанных процессов.</p> <p>Если указаны запрещающие критерии (match not process-name) – логируются сообщения всех не запрещенных процессов.</p> <p>По умолчанию разрешено логирование сообщений всех процессов.</p>

Шаг	Описание	Команда	Ключи
3	Указать уровень важности сообщений, которые будут отправляться на snmp-сервер.	<code>rtt(config-syslog-snmpp) # severity <SEVERITY></code>	<p><SEVERITY> – уровень важности сообщения, принимает значения (в порядке убывания важности):</p> <ul style="list-style-type: none"> • 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.
4	Включить отображение syslog-сообщений при удаленных подключениях (Telnet, SSH) (не обязательно).	<code>rtt(config) # syslog monitor</code>	
5	Активировать или деактивировать отображение при удаленных подключениях событий работы отдельных процессов маршрутизатора (не обязательно).	<code>rtt(config-syslog-monitor) # match [not] process-name <PROCESS-NAME></code>	<PROCESS-NAME> – описано во 2 пункте.
6	Указать уровень важности сообщений, которые будут отображаться при удаленных подключениях.	<code>rtt(config-syslog-monitor) # severity <SEVERITY></code>	<SEVERITY> – описано в 3 пункте.

Шаг	Описание	Команда	Ключи
7	Включить отображение syslog-сообщений при консольном подключении (не обязательно).	<code>rtt(config)# syslog console</code>	
8	Активировать или деактивировать отображение при консольном подключении событий работы отдельных процессов маршрутизатора (не обязательно).	<code>rtt(config-syslog-console)# match [not] process-name <PROCESS-NAME></code>	<PROCESS-NAME> – описано во 2 пункте.
9	Указать уровень важности сообщений, которые будут отображаться при консольном подключении.	<code>rtt(config-syslog-console)# severity <SEVERITY></code>	<SEVERITY> – описано в 3 пункте.
10	Указать категорию сообщений, которые будут сохраняться в локальный syslog-файл или отправляться на удаленный syslog-сервер.	<code>rtt(config)# syslog facility <FACILITY></code>	<FACILITY> – категория сообщений, принимает значения [local0..local7].
11	Включить сохранение сообщений syslog в указанный файл журнала (при необходимости ведения локального syslog-файла).	<code>rtt(config)# syslog file <NAME></code>	<NAME> – имя файла, в который будет производиться запись сообщений заданного уровня, задается строкой до 31 символа.
12	Активировать или деактивировать сохранение в локальный syslog-файл событий работы отдельных процессов маршрутизатора (не обязательно).	<code>rtt(config-syslog-file)# match [not] process-name <PROCESS-NAME></code>	<PROCESS-NAME> – описано во 2 пункте.

Шаг	Описание	Команда	Ключи
13	Указать уровень важности сообщений, которые будут сохраняться в локальный syslog-файл.	<code>rtt(config-syslog-file)# severity <SEVERITY></code>	<SEVERITY> – описано в 3 пункте.
14	Указать максимальный размер файла журнала (не обязательно).	<code>rtt(config)# syslog file-size <SIZE></code>	<SIZE> – размер файла, принимает значение [10..10000000] Кбайт.
15	Задать максимальное количество файлов, сохраняемых при ротации (не обязательно).	<code>rtt(config)# syslog max-files <NUM></code>	<NUM> – максимальное количество файлов, принимает значения [1.. 1000].
16	Включить передачу сообщений syslog на удаленный syslog-сервер (при необходимости отправки сообщений на удаленный syslog-сервер).	<code>rtt(config)#syslog host <HOSTNAME></code>	<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;
17	Указать IPv4/IPv6-адрес удаленного syslog-сервера.	<code>rtt(config-syslog-host)# remote-address { <ADDR> <IPv6-ADDR> }</code>	<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].
18	Указать IPv4/IPv6-адрес маршрутизатора, от которого будут отправляться пакеты на удаленный syslog-сервер (не обязательно).	<code>rtt(config-syslog-host)# source-address { <ADDR> <IPv6-ADDR> object-group <NETWORK_OBJ_GROUP_NAME> }</code>	<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]; <NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве source address; Значение по умолчанию: IPv4/IPv6-адрес интерфейса, с которого отправляются пакеты на удаленный syslog-сервер.

Шаг	Описание	Команда	Ключи
19	Указать транспортный протокол для передачи пакетов на удаленный syslog-сервер (не обязательно).	<code>rtt(config-syslog-host)# transport { tcp udp }</code>	<VRF> – имя экземпляра VRF, в котором доступен удаленный syslog-сервер, задается строкой до 31 символа; Значение по умолчанию: отсутствует (глобальная таблица маршрутизации).
20	Указать имя экземпляра VRF, в рамках которого будут отправляться пакеты на удаленный syslog-сервер (не обязательно).	<code>rtt(config-syslog-host)# vrf <VRF></code>	
21	Указать номер TCP/UDP-порта, на который будут отправляться пакеты с syslog-сообщениями (не обязательно).	<code>rtt(config-syslog-host)# port <PORT></code>	<PORT> – номер TCP/UDP-порта, на который будут отправляться пакеты с syslog-сообщениями. Значение по умолчанию: 514.
22	Активировать или деактивировать отправку на удаленный syslog-сервер событий работы отдельных процессов маршрутизатора (не обязательно).	<code>rtt(config-syslog-host)# match [not] process-name <PROCESS-NAME></code>	<PROCESS-NAME> – описано во 2 пункте.
23	Указать уровень важности сообщений, которые будут сохраняться в локальный syslog-файл.	<code>rtt(config-syslog-host)# severity <SEVERITY></code>	<SEVERITY> – описано в 3 пункте.
24	Включить вывод отладочных сообщений во время загрузки устройства (не обязательно).	<code>rtt(config)#syslog reload debugging</code>	
25	Включить процесс логирования введенных команд пользователя на локальный syslog-сервер (не обязательно).	<code>rtt(config)# syslog cli-commands</code>	
26	Включить нумерацию сообщений (не обязательно).	<code>rtt(config)#syslog sequence-numbers</code>	

Шаг	Описание	Команда	Ключи
27	Включить точность даты сообщений до миллисекунд (не обязательно).	<code>rtt(config)#syslog timestamp msec</code>	
28	Включить отображение имени процесса, который сформировал сообщение (не обязательно).	<code>rtt(config)#syslog program-name</code>	
29	Включить регистрацию неудачных аутентификаций (не обязательно).	<code>rtt(config)#logging login on-failure</code>	
30	Включить регистрацию изменений настроек системы аудита (не обязательно).	<code>rtt(config)#logging syslog configuration</code>	
31	Включить регистрацию изменений настроек пользователя (не обязательно).	<code>rtt(config)#logging userinfo</code>	

20.5.2. Пример настройки

Задача:

Настроить отправку сообщений для следующих системных событий:

- неудачная аутентификация пользователя;
- внесены изменения в конфигурацию логирования системных событий;
- старт/остановка системного процесса;
- внесены изменения в профиль пользователей.

IP-адрес маршрутизатора RTT – 192.168.52.8, IP-адрес Syslog-сервера – 192.168.52.41. Использовать параметры по умолчанию для отправки сообщений – протокол UDP порт 514.



Решение:

Предварительно нужно выполнить следующие действия:

- указать зону для интерфейса gi1/0/1;
- настроить IP-адрес для интерфейсов gi1/0/1.

Основной этап конфигурирования:

Создаем файл на маршрутизаторе для системного журнала, уровень сообщений для журналирования – info:

```
rtt(config)# syslog file tmpsys:syslog/RTT
rtt(config-syslog-file)# severity info
rtt(config-syslog-file)# exit
```

Указываем IP адрес и параметры удаленного syslog-сервера:

```
rtt(config)# syslog host SERVER
rtt(config-syslog-host)# remote-address 192.168.52.41
rtt(config-syslog-host)# severity info
rtt(config-syslog-host)# exit
```

Задаем логирование неудачных попыток аутентификации:

```
rtt(config)# logging login on-failure
```

Задаем логирование изменений конфигурации syslog:

```
rtt(config)# logging syslog configuration
```

Задаем логирование старта/остановки системных процессов:

```
rtt(config)# logging service start-stop
```

Задаем логирование внесений изменений в профиль пользователей:

```
rtt(config)# logging userinfo
```

Изменения конфигурации вступят в действие после применения:

```
rtt# commit
Configuration has been successfully committed
rtt# confirm
Configuration has been successfully confirmed
```

Посмотреть текущую конфигурацию системного журнала:

```
rtt# show syslog configuration
```

Посмотреть записи системного журнала:

```
rtt# show syslog RTT
```

20.6. Проверка целостности

Проверка целостности подразумевает проверку целостности хранимых исполняемых файлов.

20.6.1. Процесс настройки

Шаг	Описание	Команда	Ключи
1	Запустить проверку целостности системы	<code>rtt# verify filesystem</code> <code><detailed></code>	detailed – детальный вывод информации в консоль.

20.6.2. Пример конфигурации

Задача:

Проверить целостность файловой системы.

Решение:

Запускаем проверку целостности:

```
rtt# verify filesystem
Filesystem Successfully Verified
```

20.7. Настройка архивации конфигурации маршрутизатора

На маршрутизаторах RTT предусмотрена функция локального и/или удаленного копирования конфигурации по таймеру или при применении конфигурации.

20.7.1. Процесс настройки

Шаг	Описание	Команда	Ключи
1	Перейти в режим настройки параметров резервирования конфигурации.	<code>rtt(config)# archive</code>	
2	Установить тип сохранения резервных конфигураций маршрутизатора (не обязательно).	<code>rtt(config-ahchive)# type <TYPE></code>	<p><TYPE> – тип сохранения резервных конфигураций маршрутизатора. Принимает значения:</p> <ul style="list-style-type: none"> • local; • remote; • both. <p>Значение по умолчанию: remote.</p>
3	Включить режим резервирования конфигурации по таймеру (не обязательно).	<code>rtt(config-ahchive)# auto</code>	

Шаг	Описание	Команда	Ключи
4	Включить режим резервирования конфигурации после каждого успешного применения конфигурации (не обязательно).	rtt(config-ahchive)# by-commit	
5	Указать путь для удаленного копирования конфигураций маршрутизатора (обязательно для типов remote и both).	rtt(config-ahchive)# path <PATH>	<PATH> – определяет протокол, адрес сервера, расположение и префикс имени файла на сервере.
6	Задать период времени для автоматического резервирования конфигурации (не обязательно, актуально только для режима auto).	rtt(config-ahchive)# time-period <TIME>	<TIME> – периодичность автоматического резервирования конфигурации, принимает значение в минутах [1..525600]. Значение по умолчанию: 720 минут.
7	Задать максимальное количество локально сохраняемых резервных копий конфигураций (не обязательно, актуально при типах local и both).	rtt(config-ahchive)# count-backup <NUM>	<NUM> – максимальное количество локально сохраняемых резервных копий конфигураций. Принимает значения в диапазоне [1..100]. Значение по умолчанию: 1.

20.7.2. Пример конфигурации

Задача:

Настроить локальное и удаленное резервное копирование конфигурации маршрутизатора 1 раз в сутки и при успешном изменении конфигурации. Удаленные копии необходимо отправлять на tftp-сервер 172.16.252.77 в подпапку rtt-example. Максимальное количество локальных копий – 30.

Решение:

Для успешной работы удаленной архивации конфигураций, между маршрутизатором и сервером должна быть организована IP-связность, настроены разрешения на прохождение tftp-трафика по сети и сохранения файлов на сервере.

Основной этап конфигурирования:

Перейти в режим конфигурирования резервного копирования конфигураций:

```
rtt# configure
rtt(config)# archive
```

Задать режим локального и удаленного резервного копирования конфигурации:

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

Настроить путь для удаленного копирования конфигураций и максимальное количество локальных резервных копий:

```
rtt(config-archive)# path tftp://172.16.252.77:/rtt-example/rtt-example.cfg
rtt(config-archive)# count-backup 30
```

Задать интервал резервного копирования конфигурации в случае отсутствия изменений:

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

Включить режимы архивации конфигурации маршрутизатора по таймеру и при успешном изменении конфигурации:

```
rtt(config-archive)# auto
rtt(config-archive)# by-commit
```

После применения данной конфигурации 1 раз в сутки и при каждом успешном изменении конфигурации маршрутизатора на tftp-сервер будет отправляться конфигурационный файл с именем вида "rtt-exampleYYYYMMDD_HHMMSS.cfg". Также на самом маршрутизаторе в разделе flash:backup/ будет создаваться файл с именем вида "config_YYYYMMDD_HHMMSS". Когда в разделе flash:backup/ накопится 30 таких файлов, при создании нового будет удаляться наиболее старый. Посмотреть можно командой:

```
rtt(config)# show archive configuration
```

20.8. Настройка SLA

IP SLA (Internet Protocol Service Level Agreement) — технология измерения активных компьютерных сетей. На маршрутизаторах RTT, сервис IP SLA использует непрерывную генерацию трафика для тестирования качественных и количественных характеристик каналов связи в сети передачи данных на базе протокола IP. Два основных понятия при рассмотрении сервиса IP-SLA: SLA-agent (SLA-sender) – тестирующий маршрутизатор, отправляющий запросы; SLA-responder – удаленный/тестируемый маршрутизатор или произвольный хост, принимающий запросы от SLA-sender.

20.8.1. Алгоритм настройки SLA-теста

Шаг	Описание	Команда	Ключи
1	Создать в системе новый SLA-тест и перейти в режим его конфигурирования.	<code>rtt(config)# ip sla test <NUM></code>	<NUM> – номер SLA-теста, задается в диапазоне [1..10000].
2	Задать режим тестирования канала связи и параметры тестирования. Разные режимы подразумевают различный набор параметров, которые необходимо указать. Для одного SLA-теста возможно указать только один набор параметров тестирования.		

Шаг	Описание	Команда	Ключи
2.1	Конфигурирование ICMP-режима тестирования канала связи.	<pre> rtt(config-sla-test)# icmp-echo { <DST-ADDRESS> <IPV6-DST-ADDRESS> } { source-ip { <SRC-ADDRESS> <IPV6-SRC-ADDRESS> object-group <NETWORK_OBJ_GROUP_NAME> } source-interface { <IF> <TUN> } } [interval <INTERVAL>] [num-packets <NUM-PACKETS>] </pre>	<p><DST-ADDRESS> – IPv4-адрес, на который будут направляться тестовые пакеты. Задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><IPV6-DST-ADDRESS> – IPv6-адрес, на который будут направляться тестовые пакеты. Задаётся в виде X:X:X:X:X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];</p> <p><SRC-ADDRESS> – IPv4-адрес, с которого будут отправляться тестовые пакеты. Задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><IPV6-SRC-ADDRESS> – IPv6-адрес, с которого будут отправляться тестовые пакеты. Задаётся в виде X:X:X:X:X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];</p> <p><NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве source address;</p> <p><IF> – тип и идентификатор интерфейса, IP-адрес которого будет использоваться в качестве адреса источника пакетов. Задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;</p> <p><TUN> – тип и идентификатор туннеля, IP-адрес которого будет использоваться в качестве адреса источника пакетов. Задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;</p> <p><INTERVAL> – интервал между отправкой каждого последующего тестового пакета. Может принимать значение [1..255] миллисекунд;</p> <p><NUM-PACKETS> – количество тестовых пакетов, отправляемых в рамках одной сессии тестирования. Может принимать значение [1..100000].</p>

2.2	<p>Конфигурирование UDP-режима тестирования канала связи.</p> <p>Для корректной работы UDP-режим предполагает сконфигурированный Eltex SLA-responder на удаленной стороне.</p>	<pre> rtt(config-sla-test)# udp- jitter { <DST-ADDRESS> <IPV6-DST-ADDRESS> } <DST- PORT> { source-ip { <SRC- ADDRESS> <IPV6-SRC- ADDRESS> object-group <NETWORK_OBJ_GROUP_NAME> } source-interface { <IF> <TUN> } } [source-port <SRC-PORT>] [interval <INTERVAL>] [num-packets <NUM-PACKETS>] </pre>	<p><DST-ADDRESS> – IPv4-адрес, на который будут направляться тестовые пакеты. Задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><IPV6-DST-ADDRESS> – IPv6-адрес, на который будут направляться тестовые пакеты. Задаётся в виде X:X:X:X::X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];</p> <p><DST-PORT> – номер UDP-порта назначения тестовых пакетов, принимает значения [1..65535];</p> <p><SRC-ADDRESS> – IPv4-адрес, с которого будут отправляться тестовые пакеты. Задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><IPV6-SRC-ADDRESS> – IPv6-адрес, с которого будут отправляться тестовые пакеты. Задаётся в виде X:X:X:X::X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];</p> <p><NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве source address;</p> <p><IF> – тип и идентификатор интерфейса, с IP-адреса которого будут отправляться тестовые пакеты. Задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;</p> <p><TUN> – тип и идентификатор туннеля, с IP-адреса которого будут отправляться тестовые пакеты. Задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;</p> <p><SRC-PORT> – номер UDP-порта источника тестовых пакетов, принимает значения [1..65535];</p> <p><INTERVAL> – интервал между отправкой каждого последующего тестового пакета. Может принимать значение [1..255] миллисекунд;</p>
-----	--	---	--

Шаг	Описание	Команда	Ключи
			<p><NUM-PACKETS> – количество тестовых пакетов, отправляемых в рамках одной сессии тестирования. Может принимать значение [1..100000].</p>
2.3	Конфигурирование TCP-режима тестирования канала связи.	<pre> rtt(config-sla-test)# tcp- connect { <DST-ADDRESS> <IPV6-DST-ADDRESS> } { source-ip { <SRC-ADDRESS> <IPV6-SRC-ADDRESS> object-group <NETWORK_OBJ_GROUP_NAME> } source-interface { <IF> <TUN> } } [source-port <SRC-PORT>] </pre>	<p><DST-ADDRESS> – IPv4-адрес, на который будут направляться тестовые пакеты. Задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><IPV6-DST-ADDRESS> – IPv6-адрес, на который будут направляться тестовые пакеты. Задаётся в виде X:X:X:X::X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];</p> <p><DST-PORT> – номер TCP-порта назначения тестовых пакетов, принимает значения [1..65535];</p> <p><SRC-ADDRESS> – IPv4-адрес, с которого будут отправляться тестовые пакеты. Задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><IPV6-SRC-ADDRESS> – IPv6-адрес, с которого будут отправляться тестовые пакеты. Задаётся в виде X:X:X:X::X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];</p> <p><NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве source address;</p> <p><IF> – тип и идентификатор интерфейса, с IP-адреса которого будут отправляться тестовые пакеты. Задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;</p> <p><TUN> – тип и идентификатор туннеля, с IP-адреса которого будут отправляться тестовые пакеты. Задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;</p> <p><SRC-PORT> – номер TCP-порта источника тестовых пакетов, принимает значения [1..65535].</p>

Шаг	Описание	Команда	Ключи
3	Установить значение dscp, которым будут помечаться тестовые пакеты (необязательно).	<code>rtt(config-sla-test) # dscp <DSCP></code>	<DSCP> – значение кода DSCP, может принимать значение [0..63].
4	Установить частоту перезапуска SLA-теста (необязательно).	<code>rtt(config-sla-test) # frequency <TIME></code>	<TIME> – частота перезапуска SLA-теста, может принимать значение [1..604800] секунд.
5	Установить размер исходящего тестового пакета (необязательно).	<code>rtt(config-sla-test) # packet-size <SIZE></code>	<SIZE> – размер тестового пакета SLA-теста, может принимать значение [70..10000] байт.
6	Установить максимальное время ожидания ответа от удаленной стороны на тестовый пакет (необязательно).	<code>rtt(config-sla-test) # timeout <TIME></code>	<TIME> – время ожидания ответного пакета от удаленной стороны, может принимать значение [1..4294967295] миллисекунд.
7	Установить значение TTL для исходящих пакетов SLA-теста (необязательно).	<code>rtt(config-sla-test) # ttl <TTL></code>	<TTL> – значение TTL, может принимать значение [1..255].
8	Указать экземпляр VRF, в адресном пространстве которого должен работать SLA-тест (если подразумевается конфигурацией).	<code>rtt(config-sla-test) # vrf <VRF></code>	<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Шаг	Описание	Команда	Ключи
9	Установить пороговые значения характеристик канала (необязательно).	<pre> rtt(config-sla-test)# thresholds <TYPE> { high <VALUE_H> low <VALUE_L> forward [{ high <VALUE_H> low <VALUE_L> }] reverse [{ high <VALUE_H> low <VALUE_L> }] } [from-last <NUM- CHECK> { all over <NUM- LIMIT> }] </pre>	<p><TYPE> – тип отслеживаемой величины, может принимать значения:</p> <ul style="list-style-type: none"> • delay – допустимые значения задержек в канале; • jitter – допустимые значения джиттера в канале; • losses – допустимые значения потерь пакетов в канале. <p><VALUE_H> – верхнее пороговое значение, при пересечении которого сессия тестирования будет считаться проваленной;</p> <p><VALUE_L> – нижнее пороговое значение, при пересечении которого сессия тестирования будет вновь считаться успешной;</p> <p><NUM-CHECKS> – количество итераций теста, в течение которых проверяется соблюдение пороговых значений характеристик канала;</p> <p><NUM-LIMIT> – количество итераций теста с превышением установленного порога, после превышения которого тест считается проваленным.</p>
10	Запретить фрагментацию тестовых пакетов (необязательно).	<pre> rtt(config-sla-test)# disallow-fragmentation </pre>	
11	Установить количество записей о результатах тестирования, отображающихся в истории измерений (необязательно).	<pre> rtt(config-sla-test)# history <SIZE> </pre>	<p><SIZE> – число сохраняемых результатов, может принимать значение [1..1000].</p>
12	Задать описание SLA-теста (необязательно).	<pre> rtt(config-sla-test)# description <DESCRIPTION> </pre>	<p><DESCRIPTION> – описание SLA-теста, задается строкой до 255 символов.</p>
13	Настроить параметры аутентификации, согласно алгоритму настройки параметров аутентификации (только для UDP-тестов, необязательно).		
14	Активировать SLA-тест.	<pre> rtt(config-sla-test)# enable </pre>	

Шаг	Описание	Команда	Ключи
15	Выйти из параметров конфигурирования SLA-теста.	<code>rtt(config-sla-test)# exit</code>	
16	Задать расписание запуска активированных SLA-тестов.	<pre> rtt(config)# ip sla schedule { <TEST-NUMBER> all } [life { <LIFE-TIME> forever }] [start-time { <MONTH> <DAY> <TIME> now }] </pre>	<p><TEST-NUMBER> – номер SLA-теста, может принимать значение [1..10000]. При использовании ключа "all" вместо номера, устанавливается расписание работы для всех активированных SLA-тестов;</p> <p><LIFE-TIME> – время жизни теста, может принимать значение [1..2147483647] секунд;</p> <p>forever – время жизни теста не ограничено;</p> <p><TIME> – время начала теста, задаётся в виде HH:MM:SS, где:</p> <p>HH – часы, может принимать значение [0..23];</p> <p>MM – минуты, может принимать значение [0..59];</p> <p>SS – секунды, может принимать значение [0..59];</p> <p><MONTH> – месяц начала теста, принимает значения [January / February / March / April / May / June / July / August / September / October / November / December];</p> <p><DAY> – день месяца начала теста, может принимать значение [1..31];</p> <p>now – начать тест немедленно.</p>

Шаг	Описание	Команда	Ключи
17	Активировать вывод информационных сообщений групп событий (необязательно). Каждая из групп активируется отдельно.	<code>rtt(config)# ip sla logging <TYPE></code>	<p><TYPE> – название группы информационных сообщений, может принимать значения:</p> <ul style="list-style-type: none"> • error – отображение сообщений об ошибках в работе SLA-тестов, причинах их провала, а также ошибок в работе SLA-responder (если таковой сконфигурирован в системе); • delay – отображение сообщений о превышении/нормализации значений, установленных в thresholds delay; • jitter – отображение сообщений о превышении/нормализации значений, установленных в thresholds jitter; • losses – отображение сообщений о превышении/нормализации значений, установленных в thresholds losses; • status – отображение сообщений о смене статуса SLA-теста.
18	Активировать сервис SLA-agent.	<code>rtt(config)# ip sla</code>	

20.8.2. Настройка SLA-responder

Шаг	Описание	Команда	Ключи
1	В режиме конфигурирования интерфейса на маршрутизаторе, который является удаленной стороной SLA-теста, активировать SLA-responder.	<div> <code>rtt(config-if-gi)# ip sla responder <TYPE></code> </div> <div> <code>rtt(config-if-gi)# ipv6 sla responder eltex</code> </div>	<p><TYPE> – название целевой платформы SLA-agent, может принимать значения:</p> <ul style="list-style-type: none"> • eltex – функционал Eltex SLA-responder для Eltex SLA-agent; • cisco – функционал Cisco SLA-responder для Cisco SLA-agent.
2	Установить UDP-порт, на котором будет идти прослушивание запросов аутентификации от SLA-agent (если при конфигурировании SLA-теста был указан порт прохождения контрольной фазы, отличный от порта по умолчанию).		
2.1	Конфигурирование UDP-порта для прослушивания запросов аутентификации от Eltex SLA-agent (необязательно).	<div> <code>rtt(config-if-gi)# ip sla responder eltex port <PORT></code> </div> <div> <code>rtt(config-if-gi)# ipv6 sla responder eltex port <PORT></code> </div>	<p><PORT> – номер UDP-порта, может принимать значение [1..65535].</p>

Шаг	Описание	Команда	Ключи
2.2	Конфигурирование UDP-порта для прослушивания запросов аутентификации от Cisco SLA-agent (необязательно).	<code>rtt(config-if-gi) # ip sla responder cisco port <PORT></code>	<PORT> – номер UDP-порта, может принимать значение [1..65535].

20.8.3. Пример настройки ICMP-режима тестирования

Задача:

Настроить постоянную проверку доступности публичного DNS-сервера с IP-адресом 8.8.8.8. Интерфейс, имеющий доступ в сеть Интернет gi1/0/1, имеет адрес 192.168.44.15.

Решение:

Для выяснения сетевой доступности достаточной является проверка с помощью ICMP-запросов. Для этого настроим SLA-тест с типом icmp-echo и всеми параметрами по умолчанию:

```
rtt# configure
rtt(config)# ip sla test 1
rtt(config-sla-test)# icmp-echo 8.8.8.8 source-ip 192.168.44.15
rtt(config-sla-test)# enable
rtt(config-sla-test)# exit
rtt(config)# ip sla schedule 1 life forever start-time now
```

Также включим логирование событий смены статуса теста и сообщений о причинах неудачи (на случай, если адрес перестанет быть доступен).

```
rtt(config)# ip sla logging status
rtt(config)# ip sla logging error
rtt(config)# ip sla
rtt(config)# exit
rtt# commit
```

После применения конфигурации тест стартует, и выводится сообщение о его текущем состоянии:

```
rtt# 2023-12-13T14:01:55+00:00 %IP_SLA-I-STATUS: (test 1) State changed to success
```

Сводную информацию о результате и конфигурации теста можно вывести командой:

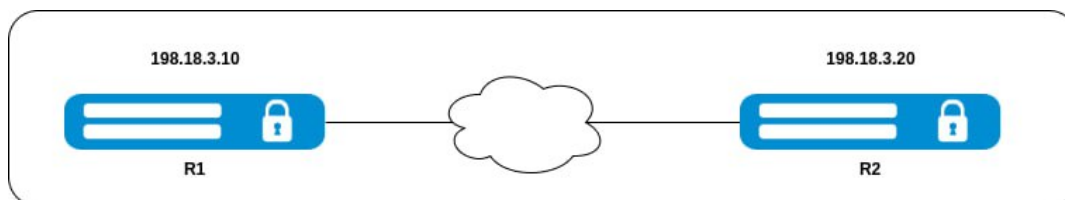
```
rtt# show ip sla test status
```

Test	Type	Source	Destination	Status	Last Run
1	icmp-echo	192.168.44.15	8.8.8.8	Successful	1 second(s) ago

20.8.4. Пример настройки UDP-режима тестирования

Задача:

Настроить тестирование качества канала связи между двумя маршрутизаторами Rusteletech. Маршрутизаторы находятся в одной подсети 198.18.3.0/24.



Решение:

Измерение качества канала связи (задержки, потери, дубликаты при передаче трафика и др.) возможно с использованием UDP-тестирования. Сконфигурируем SLA-тест с типом `udp-jitter`, который будет измерять количественные характеристики канала связи, а также сигнализировать о превышении установленных порогов.

Сконфигурируем на R1 SLA-тест `udp-jitter` с адресом назначения R2 (198.18.3.20). Поскольку ограничений на выбор портов не обозначено, воспользуемся портами 20002 на отправку и на получение. Также укажем интервал между пакетами, равный 10 мс, чтобы ускорить общий поток тестового трафика.

```
R1# configure
R1(config)# ip sla test 2
R1(config-sla-test)# udp-jitter 198.18.3.20 20002 source-ip 198.18.3.10 source-
port 20002 interval 10
```

Далее установим пороговые значения для информирования об ухудшении качества канала: максимальные значения двусторонней задержки – 15 мс, джиттера – 5мс и потерь – 5 пакетов (из 100 в настройках по умолчанию):

```
R1(config-sla-test)# thresholds delay high 15
R1(config-sla-test)# thresholds jitter forward high 5
R1(config-sla-test)# thresholds jitter reverse high 5
R1(config-sla-test)# thresholds losses high 5
```

Активируем тест и зададим расписание, согласно которому тест запустится немедленно и не будет иметь ограничений по следующим перезапускам:

```
R1(config-sla-test)# enable
R1(config)# ip sla schedule 2 start-time now life forever
```

Включим отображение всех групп сообщений, активируем сервис SLA-agent и применим конфигурацию:

```
R1(config)# ip sla logging status
```

```
R1(config)# ip sla logging error
R1(config)# ip sla logging delay
R1(config)# ip sla logging jitter
R1(config)# ip sla logging losses
R1(config)# ip sla
R1(config)# exit
R1# commit
```

Тест будет завершаться ошибкой до тех пор, пока не будет активирован Eltex SLA-responder на второй стороне – маршрутизаторе R2:

```
R1# 2023-12-13T14:01:55+00:00 %IP_SLA-I-STATUS: (test 2) State changed to fail
R1# 2023-12-13T14:01:55+00:00 %IP_SLA-E-ERROR: (test 2) Control phase failed:
destination host is not responding
```

Для этого перейдем в режим конфигурирования интерфейса, адрес которого ранее был указан как адрес назначения SLA-теста, и включим на нем Eltex SLA-responder:

```
R2(config)# interface gigabitethernet 1/0/1
R2(config-if-gi)# ip sla responder eltex
R2(config-if-gi)# exit
R2(config)# exit
R2# commit
```

Порт назначения пакетов аутентификации по умолчанию – 1800 и должен быть открыт на R2. Если прохождение трафика по данному порту запрещено, необходимо изменить настройку портов, воспользовавшись **Алгоритм настройки параметров аутентификации**, а также командами из раздела **Настройка SLA-responder**.

После активации SLA-responder тест перейдет в состояние 'Успешно'.

```
2023-12-13T15:35:32+00:00 %IP_SLA-I-STATUS: (test 2) State changed to success
```

При ухудшении характеристик канала и, вследствие, превышения обозначенных пороговых значений, на R1 будут выводиться сообщения вида:

```
2023-12-13T15:59:22+00:00 %IP_SLA-I-DELAY: (test 2) Two-way delay is high: 50.71ms > 15ms
2023-12-13T16:00:40+00:00 %IP_SLA-I-LOSSES: (test 2) Total losses are high: 43 > 5
2023-12-13T16:04:04+00:00 %IP_SLA-I-JITTER: (test 2) One-way jitter in forward direction
is high: 9.41ms > 5ms
2023-12-13T16:04:04+00:00 %IP_SLA-I-JITTER: (test 2) One-way jitter in reverse direction
is high: 9.41ms > 5ms
```

Сам SLA-тест при этом перейдет в состояние Fail и будет оставаться в нем до тех пор, пока характеристики канала не вернуться в допустимые пределы.

Просмотреть результаты измерений теста можно командой:

```
R1# show ip sla test statistics 2
Test number:                2
Test status:                 Successful
Transmitted packets:         100
Lost packets:                39 (39.00%)
Lost packets in forward direction: 0 (0.00%)
```

```

Lost packets in reverse direction:          39 (39.00%)
One-way delay forward min/avg/max:         0.08/94.10/130.86 milliseconds
One-way delay reverse min/avg/max:         0.08/94.10/130.86 milliseconds
One-way jitter forward:                   35.94 milliseconds
One-way jitter reverse:                   35.94 milliseconds
Two-way delay min/avg/max:                 0.15/188.19/261.73 milliseconds
Duplicate packets:                         5
Out of sequence packets in forward direction: 0
Out of sequence packets in reverse direction: 40

```

20.8.5. Алгоритм настройки параметров аутентификации

Шаг	Описание	Команда	Ключи
1	В режиме конфигурирования SLA-теста, установить тип алгоритма, который будет использоваться при хешировании ключей аутентификации.	<code>rtt(config-sla-test)# control-phase authentication algorithm <ALGORITHM></code>	<ALGORITHM> – алгоритм хеширования, принимает значения [sha-256, hmac-sha-256].
2	Задать ключ, который будет использоваться в процессе прохождения контрольной фазы для аутентификации. Может быть использован один из двух видов ключей аутентификации: ключ-строка, указываемая непосредственно в режиме конфигурирования SLA-теста, и ключ, содержащийся в предварительно сконфигурированной связке ключей (key-chain).		
2.1	При использовании ключ-строки установить ее непосредственно в режиме конфигурирования SLA-теста.	<code>rtt(config-sla-test)# control-phase authentication key-string ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<CLEAR-TEXT> – строка длиной 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...).
2.2.1	При использовании ключа из связки ключей вернуться в общий режим конфигурирования, а затем создать новую связку ключей.	<code>rtt(config)# key-chain <KEYCHAIN></code>	<KEYCHAIN> – идентификатор связки ключей, задается строкой длиной до 16 символов.
2.2.2	В режиме конфигурирования связки ключей создать новый ключ.	<code>rtt(config-key-chain)# key <NUM></code>	<NUM> – номер-идентификатор ключа в связке ключей, может принимать значение [1..255].
2.2.3	Привязать к созданному ключу ключ-строку.	<code>rtt(config-key-chain-key)# key-string ascii-text { <CLEAR-TEXT> </code>	<CLEAR-TEXT> – строка длиной от 8 до 16 символов;

Шаг	Описание	Команда	Ключи
		<code>encrypted</code> <code><ENCRYPTED-TEXT></code> <code>}</code>	<code><ENCRYPTED-TEXT></code> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).
2.2.4	Установить период времени, в течение которого данный ключ может использоваться для аутентификации исходящих пакетов (необязательно).	<code>rtt(config-</code> <code>keychain-key) #</code> <code>send-lifetime</code> <code><TIME_B> <DAY_B></code> <code><MONTH_B></code> <code><YEAR_B></code> <code><TIME_E> <DAY_E></code> <code><MONTH_E></code> <code><YEAR_E></code>	<p><code><TIME_B></code> – устанавливаемое время начала действия ключа, задаётся в виде HH:MM:SS, где:</p> <p>HH – часы, принимает значение [0..23];</p> <p>MM – минуты, принимает значение [0 .. 59];</p> <p>SS – секунды, принимает значение [0 .. 59].</p> <p><code><DAY_B></code> – день месяца начала действия ключа, принимает значения [1..31];</p> <p><code><MONTH_B></code> – месяц начала использования ключа, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];</p> <p><code><YEAR_B></code> – год начала использования ключа, принимает значения [2001..2037];</p> <p><code><TIME_E></code> – устанавливаемое время окончания действия ключа, задаётся в виде HH:MM:SS, где:</p> <p>HH – часы, принимает значение [0..23];</p> <p>MM – минуты, принимает значение [0 .. 59];</p> <p>SS – секунды, принимает значение [0 .. 59].</p> <p><code><DAY_E></code> – день месяца окончания действия ключа, принимает значения [1..31];</p> <p><code><MONTH_E></code> – месяц окончания действия ключа, принимает значения [January/February/March/April/May/June/July/August/September/October /November/December];</p> <p><code><YEAR_E></code> – год окончания действия ключа, принимает значения [2001..2037].</p>
2.2.5	Установить период времени, в течение которого данный ключ может использоваться для аутентификации входящих пакетов (необязательно).	<code>rtt(config-</code> <code>keychain-key) #</code> <code>accept-lifetime</code> <code><TIME_B> <DAY_B></code> <code><MONTH_B></code> <code><YEAR_B></code> <code><TIME_E> <DAY_E></code>	<p><code><TIME_B></code> – устанавливаемое время начала действия ключа, задаётся в виде HH:MM:SS, где:</p> <p>HH – часы, принимает значение [0..23];</p>

Шаг	Описание	Команда	Ключи
		<code><MONTH_E></code> <code><YEAR_E></code>	<p>MM – минуты, принимает значение [0 .. 59];</p> <p>SS – секунды, принимает значение [0 .. 59].</p> <p><DAY_B> – день месяца начала действия ключа, принимает значения [1..31];</p> <p><MONTH_B> – месяц начала использования ключа, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];</p> <p><YEAR_B> – год начала использования ключа, принимает значения [2001..2037];</p> <p><TIME_E> – устанавливаемое время окончания действия ключа, задаётся в виде HH:MM:SS, где:</p> <p>HH – часы, принимает значение [0..23];</p> <p>MM – минуты, принимает значение [0 .. 59];</p> <p>SS – секунды, принимает значение [0 .. 59].</p> <p><DAY_E> – день месяца окончания действия ключа, принимает значения [1..31];</p> <p><MONTH_E> – месяц окончания действия ключа, принимает значения [January/February/March/April/May/June/July/August/September/October /November/December];</p> <p><YEAR_E> – год окончания действия ключа, принимает значения [2001..2037].</p>
2.2.6	Вернуться в общий режим конфигурирования и привязать ранее созданную связку ключей к сервису SLA.	<code>rtt(config)# ip</code> <code>sla key-chain</code> <code><KEYCHAIN></code>	<KEYCHAIN> – идентификатор связки ключей, задается строкой длиной до 16 символов.
2.2.7	Перейти в режим конфигурирования ранее созданного SLA-теста и установить необходимый ключ из связки ключей, указав его номер.	<code>rtt(config-sla-</code> <code>test)# control-</code> <code>phase</code> <code>authentication</code> <code>key-id <NUM></code>	<NUM> – номер-идентификатор ключа в связке ключей, может принимать значение [1..255].

Шаг	Описание	Команда	Ключи
3	Указать порт, на который будут направляться пакеты для аутентификации в ходе контрольной фазы (необязательно).	<code>rtt(config-sla-test)# control-phase destination-port <PORT></code>	<PORT> – номер UDP-порта, может принимать значение [1..65535].
4	Указать порт, с которого будут отправляться пакеты для аутентификации в ходе контрольной фазы (необязательно).	<code>rtt(config-sla-test)# control-phase source-port <PORT></code>	<PORT> – номер UDP-порта, может принимать значение [1..65535].
5	Установить периодичность попыток повторного прохождения контрольной фазы в случае её неудачи.	<code>rtt(config-sla-test)# control-phase retry <TIME></code>	<TIME> – интервал между попытками, может принимать значение [1..86400] секунд.
6	Установить максимальное время ожидания ответного пакета аутентификации от удаленной стороны в ходе контрольной фазы.	<code>rtt(config-sla-test)# control-phase timeout <TIME></code>	<TIME> – время ожидания, может принимать значение [1..86400] секунд.
Далее необходимо симметрично настроить параметры аутентификации на удаленном маршрутизаторе (принимающая сторона).			
7	Перейти в режим конфигурирования сервиса SLA-responder.	<code>rtt(config)# ip sla responder [vrf <VRF>]</code>	<VRF> – имя экземпляра VRF, задаётся строкой длиной до 31 символа. При указании данного параметра, SLA-responder включается в указанном VRF.
8	Установить максимальное время ожидания следующего тестового пакета (необязательно).	<code>rtt(config-sla-responder)# timeout <TIME></code>	<TIME> – время ожидания следующего пакета, может принимать значение [1..4294967295] миллисекунд.
9	Установить тип алгоритма, который будет использоваться при хешировании ключей аутентификации.	<code>rtt(config-sla-responder)# authentication algorithm <ALGORITHM></code>	<ALGORITHM> – алгоритм хеширования, принимает значения [sha-256, hmac-sha-256].
10	Задать ключ, который будет использоваться для аутентификации приходящих запросов от SLA-agent. Может быть использован один из двух видов ключей аутентификации: ключ-строка, указываемая непосредственно в режиме конфигурирования SLA-responder, и предварительно сконфигурированная связка ключей (key-chain).		

Шаг	Описание	Команда	Ключи
10.1	При использовании ключ-строки установить ее непосредственно в режиме конфигурирования SLA-responder.	<code>rtt(config-sla-responder) # authentication key-string ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<CLEAR-TEXT> – строка длиной 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...).
10.2.1	При использовании связки ключей необходимо вернуться в общий режим конфигурирования, а затем создать новую связку ключей. Процесс создания повторяет создание связки на тестирующем маршрутизаторе и описан в рамках шагов 2.2.1 - 2.2.5 данного алгоритма.		
10.2.2	Привязать ранее созданную связку ключей к SLA-responder.	<code>rtt(config-sla-responder) # authentication key-chain <KEYCHAIN></code>	<KEYCHAIN> – идентификатор связки ключей, задается строкой длиной до 16 символов.

20.8.6. Пример конфигурации UDP-теста с аутентификацией по ключ-строке

Задача:

Установить нестандартные порты отправки и получения запросов аутентификации, а для аутентификации использовать ключ-строку. Базовый UDP-тест уже настроен.

Решение:

Конфигурация активного UDP-теста:

```
R-sender# show running-config sla
ip sla
ip sla logging error
ip sla logging status
ip sla test 1
  udp-jitter 10.0.0.1 20001 source-ip 10.0.0.2 source-port 20002
  enable
exit
ip sla schedule 1 life forever start-time now
```

```
R-responder# show running-config sla
interface gigabitethernet 1/0/3
    ip sla responder eltex
exit
```

Изменим порты отправки и получения запросов аутентификации (пакетов контрольной фазы). Для аутентификации будем использовать порт отправки – 50000 и порт получения – 49500. Для этого укажем их в параметрах SLA-теста:

```
R-sender# configure
R-sender(config)# ip sla test 1
R-sender(config)# ip sla test 1
R-sender(config-sla-test)# control-phase destination-port 49500
R-sender(config-sla-test)# control-phase source-port 50000
```

Таким образом, при каждом новом запуске SLA-теста первая пара запрос-ответ будет происходить по адресам 10.0.0.2:50000 ↔ 10.0.0.1:49500, а последующий тестовый трафик – 10.0.0.2:20002 ↔ 10.0.0.1:20001.

Здесь же укажем алгоритм для хеширования ключа и сам ключ:

```
R-sender(config-sla-test)# control-phase authentication algorithm sha-256
R-sender(config-sla-test)# control-phase authentication key-string ascii-text
sla_password
R-sender(config-sla-test)# end
R-sender# commit
```

Далее необходимо продублировать эти параметры на ответной стороне. Для этого перейдем в режим конфигурирования интерфейса, который выступает SLA-Responder, и укажем порт прослушивания запросов аутентификации:

```
R-responder# configure
R-responder(config)# interface gigabitethernet 1/0/3
R-responder(config-if-gi)# ip sla responder eltex port 49500
R-responder(config-if-gi)# exit
R-responder(config)#
```

После этого необходимо перейти в параметры SLA-Responder и указать там тот же алгоритм хеширования и ключ-пароль:

```
R-responder(config)# ip sla responder
R-responder(config-sla-responder)# authentication algorithm sha-256
R-responder(config-sla-responder)# authentication key-string ascii-text
sla_password
R-responder(config-sla-responder)# end
R-responder# commit
```

Таким образом конфигурации R-sender и R-responder:

```
R-sender# show running-config sla
ip sla
ip sla logging error
ip sla logging status
ip sla test 1
```

```
control-phase destination-port 49500
control-phase source-port 50000
control-phase authentication algorithm sha-256
control-phase authentication key-string ascii-text encrypted
8CB5107EA7005AFF2D
  udp-jitter 10.0.0.1 20001 source-ip 10.0.0.2 source-port 20002
  enable
exit
ip sla schedule 1 life forever start-time now
R-responder# show running-config sla
interface gigabitethernet 1/0/3
  ip sla responder eltex port 49500
  ip sla responder eltex
exit

ip sla responder
  authentication algorithm sha-256
  authentication key-string ascii-text encrypted 8CB5107EA7005AFF2D
exit
```

20.8.7. Пример конфигурации UDP-теста с аутентификацией по связке ключей

Задача:

Изменить конфигурацию, приведенную в примере выше, используя при этом связки ключей.

Решение:

После указания портов аутентификации, создадим связку ключей и новый ключ:

```
R-sender(config)# key-chain SLA_CHAIN
R-sender(config-key-chain)# key 1
R-sender(config-key-chain-key)# key-string ascii-text sla_password
R-sender(config-key-chain-key)# exit
R-sender(config-key-chain)# exit
R-sender(config)#
```

Привяжем созданную связку к SLA-agent, а ключ из связки привяжем к SLA-тесту:

```
R-sender(config)# ip sla key-chain SLA_CHAIN
R-sender(config)# ip sla test 1
R-sender(config-sla-test)# control-phase authentication key-id 1
R-sender(config-sla-test)# end
R-sender# commit
```

Аналогичные действия произведем на R-responder. Создадим связку ключей с необходимым ключом и привяжем связку к SLA-Responder:

```
R-responder(config)# key-chain SLA
R-responder(config-key-chain)# key 1
R-responder(config-key-chain-key)# key-string ascii-text sla_password
R-responder(config-key-chain-key)# exit
R-responder(config-key-chain)# exit
```

```
R-responder(config)# ip sla responder
R-responder(config-sla-responder)# authentication key-chain SLA
R-responder(config-sla-responder)# end
R-responder# commit
```

Таким образом конфигурации R-sender и R-responder:

```
R-sender# show running-config
key-chain SLA_CHAIN
  key 1
    key-string ascii-text encrypted 8FB80252A00E5BE802FA0217
  exit
exit

ip sla key-chain SLA_CHAIN
ip sla
ip sla logging error
ip sla logging status
ip sla test 1
  control-phase destination-port 49500
  control-phase source-port 50000
  control-phase authentication algorithm sha-256
  control-phase authentication key-id 1
  udp-jitter 10.0.0.1 20001 source-ip 10.0.0.2 source-port 20002
  enable
exit
ip sla schedule 1 life forever start-time now
R-responder# show running-config
key-chain SLA
  key 1
    key-string ascii-text encrypted 8FB80252A00E5BE802FA0217
  exit
exit

interface gigabitethernet 1/0/3
  ***
  ip sla responder eltex port 49500
  ip sla responder eltex
exit

ip sla responder
  authentication algorithm sha-256
  authentication key-chain SLA
exit
```

Использование связок ключей позволяет комбинировать различные уникальные пароли для аутентификации между SLA-agent и SLA-Responder.

20.8.8. Настройка пороговых значений

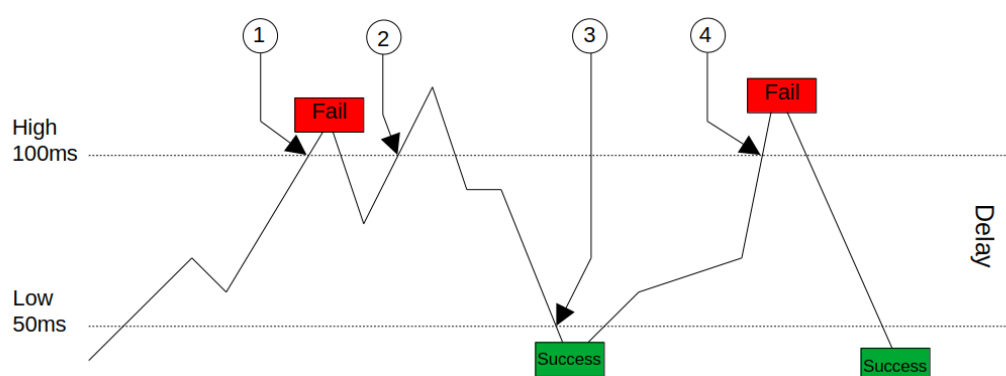
Для настройки реакции SLA-теста на определенные характеристики канала связи предусмотрен механизм активного реагирования на параметры thresholds.

Реагирование на качество канала связи может осуществляться по трём основным параметрам:

- Delay – измерение задержки между эталонным показателем интервала передачи очередного тестового пакета;
- Losses – измерение количества утерянных тестовых пакетов среди общего количества переданных пакетов;
- Jitter – измерение «дрожания» в канале, т. е. разброса во времени прохождения тестовых пакетов.

Для конфигурирования доступна настройка верхнего (high) и нижнего (low) пороговых значений для каждого из представленных параметров. Каждый из параметров может быть установлен как для одностороннего направления (forward/reverse), так и для двухстороннего (two-way).

Механизм активного изменения состояния SLA-теста при установленных пороговых значениях представлен на примере отслеживания показателя Delay с графиком значений величины во времени:



Событие	Описание
1	Первичное превышение верхнего порогового значения. Иницируется смена состояния SLA-теста на «Fail», отображается информационное сообщение о превышении установленного порога.
2	Повторное превышение верхнего порогового значения. Состояние SLA-теста неизменно (Fail), информационных сообщений не отображается. Для восстановления состояния теста ожидается снижение величины до значений меньших, чем установленный порог Low.
3	Понижение отслеживаемой величины до значений ниже установленного порога Low. Иницируется смена состояния SLA-теста на «Success», отображается информационное сообщение о нормализации показателя.
4	Очередная смена состояния SLA-теста, вызванная пересечением значения верхнего порога и предварительно нормализованным пересечением нижнего порога.

Пример конфигурации icmp-jitter SLA-теста с thresholds по показателю delay:

```
R-sender# show running-config sla
ip sla
ip sla logging delay
```

```
ip sla logging error
ip sla logging status
ip sla test 1
  icmp-jitter 198.18.0.100 source-ip 198.18.0.1 num-packets 150
  thresholds delay high 15
  thresholds delay low 8
  thresholds delay forward high 10
  thresholds delay forward low 5
  thresholds delay reverse high 10
  thresholds delay reverse low 5
  enable
exit
ip sla schedule all life forever start-time now
```



Нижний порог (low) допускается не устанавливать. В таком случае он принимается равным верхнему, и нормализация характеристики будет засчитываться при снижении значений до показателей ниже установленного уровня.

20.8.9. Измерение характеристик канала связи

Механизм IP-SLA позволяет производить замеры как двухсторонних (two-way/round), так и односторонних (forward/reverse) характеристик канала связи.

После первичной настройки IP-SLA-тестов и запроса статистики замеров может возникнуть ситуация, в которой односторонние характеристики в прямом и обратном направлениях равны друг другу, а также являются половиной от двухсторонних характеристик. Причиной такого поведения чаще всего является несоответствие вычисленной односторонней характеристики критериям проверки механизма IP-SLA.

Для корректного измерения односторонних параметров существует несколько обязательных критериев:

- **Оба узла (SLA-пробер и ответчик) должны быть синхронизованы по NTP;**
- Суммарное отклонение от NTP на обоих узлах не должно превышать значение двухсторонней характеристики;
- Значение односторонней характеристики не должно быть меньше 0, а также превышать значение двухсторонней характеристики.

Только при соблюдении данных критериев можно ожидать корректное вычисление всех характеристик канала передачи данных.

```
R-sender# show ntp peers
Clock is synchronized, stratum 5, reference is 192.168.32.1
  remote              vrf      refid      st  t  when  poll  reach  delay  offset  jitter
-----
* 192.168.32.1        -----  172.16.5.63  4   s   10    16    255    0.183  -2.314  0.711

R-sender# show ip sla test statistics 2
Test number:                2
Description:                --
Test status:                Successful
Transmitted packets:        100
Lost packets:               1 (1.00%)
Lost packets in forward direction: 1 (1.00%)
```

Lost packets in reverse direction:	0 (0.00%)
One-way delay forward min/avg/max:	23.19/24.61/30.57 milliseconds
One-way delay reverse min/avg/max:	46.26/47.25/55.28 milliseconds
One-way jitter forward:	2.77 milliseconds
One-way jitter reverse:	0.92 milliseconds
Two-way delay min/avg/max:	70.26/71.85/80.03 milliseconds
Two-way jitter min/avg/max:	2.01/3.02/3.92 milliseconds
Duplicate packets:	0
Out of sequence packets in forward direction:	0
Out of sequence packets in reverse direction:	0
Number of successes:	3 (100.00%)
Number of failures:	0 (0.00%)

21. ЧАСТО ЗАДАВАЕМЫЕ ВОПРОСЫ

Не удалось получить маршруты по BGP и/или OSPF, сконфигурированных в VRF. Соседство успешно устанавливается, но в записи маршрутов в RIB отказано
%ROUTING-W-KERNEL: Can not install route. Reached the maximum number of BGP routes in the RIB

Необходимо выделить ресурс RIB для VRF, по умолчанию он равен нулю. Делаем это в режиме конфигурирования VRF:

```
rtt(config)# ip vrf <NAME>
rtt(config-vrf)# ip protocols ospf max-routes 12000
rtt(config-vrf)# ip protocols bgp max-routes 1200000
rtt(config-vrf)# end
```

Закрываются сессии SSH/Telnet, проходящие через маршрутизатор RTT

Для поддержания сессии активной необходимо настроить передачу keepalive-пакетов. Опция отправки keepalive настраивается в клиенте SSH, например, для клиента PuTTY раздел “Соединение”.

В свою очередь, на маршрутизаторе можно выставить время ожидания до закрытия неактивных сессий TCP (в примере выставлен 1 час):

```
rtt(config)# ip firewall sessions tcp-established-timeout 3600
```

На интерфейсе был отключен firewall (ip firewall disable). После внесения этого интерфейса в security zone, удаления из конфигурации ip firewall disable и применения изменений – доступ для активных сессий с данного порта не закрылся согласно правилам security zone-pair

Изменения в конфигурации Firewall будут действовать только для новых сессий, сброса активных сессий в Firewall не происходит. Отчистить активные сессии в firewall можно командой:

```
rtt# clear ip firewall session
```

Не поднимается LACP на портах XG R-800

По умолчанию на port-channel режим speed 1000M, необходимо выставить speed 10G.

```
rtt(config)# interface port-channel 1
rtt(config-if-port-channel)# speed 10G
```

Как полностью очистить конфигурацию RTT и как сбросить на заводскую конфигурацию?

Очистка конфигурации происходит путем копирования пустой конфигурации в candidate-config и применения его в running-config.

```
rtt# copy system:default-config system:candidate-config
```

Процесс сброса на заводскую конфигурацию аналогичен.

```
rtt# copy system:factory-config system:candidate-config
```

В случае невозможности аутентификации на маршрутизаторе (неизвестен логин/пароль) конфигурацию можно сбросить к заводской следующим образом:

1. дождаться полной загрузки маршрутизатора
2. зажать функциональную кнопку "F" на 15 секунд
3. отпустить функциональную кнопку "F"
4. дождаться полной загрузки маршрутизатора с заводской конфигурацией

Как привязать subinterface к созданным VLAN?

При создании саб-интерфейса VLAN создается и привязывается автоматически (прямая зависимость индекс sub – VID).

```
rtt(config)# interface gigabitethernet 1/0/1.100
```

После применения можно наблюдать информационные сообщения:

```
2016-07-14T012:46:24+00:00 %VLAN: creating VLAN 100
```

Есть ли функционал в маршрутизаторах серии RTT для анализа трафика?

В маршрутизаторах серии RTT реализована возможность анализировать трафик на интерфейсах из CLI. Сниффер запускается командой monitor.

```
rtt# monitor gigabitethernet 1/0/1
```

Как настроить ip prefix-list 0.0.0.0/0?

Ниже приведен пример конфигурации префикс-листа, разрешающего прием маршрута по умолчанию.

```
rtt(config)# ip prefix-list rtt
rtt(config-pl)# permit 0.0.0.0/0
```

Проблема прохождения асинхронного трафика

В случае организации сети с асинхронной маршрутизацией, Firewall будет запрещать "неправильный (ошибочный)" входящий трафик (не открывающий новое соединение и не принадлежащий никакому установленному соединению) из соображений безопасности.

Разрешающее правило в Firewall не решит поставленную задачу для подобных схем.

Решить задачу можно, отключив Firewall на входном интерфейсе:

```
rtt(config-if-gi)# ip firewall disable
```

Как можно сохранить локальную копию конфигурации маршрутизатора?

Если необходимо скопировать текущую running или candidate – конфигурацию на самом маршрутизаторе – можно воспользоваться командой copy с указанием в качестве источника копирования "system:running-config" или "system:candidate-config", а в качестве назначения – файл в разделе "flash:data/".

```
rtt# copy system:candidate-config flash:data/temp.txt
```

Также существует возможность копирования ранее сохраненных конфигурационных файлов (автоматически из раздела flash:backup/ или вручную из раздела flash:data/) в candidate-конфигурацию:

```
rtt# copy flash:data/temp.txt system:candidate-config  
rtt# copy flash:backup/config_20190918_164455 system:candidate-config
```

22.ПРИЛОЖЕНИЕ А. PACKET FLOW

22.1. Порядок обработки входящего/исходящего трафика сетевыми службами маршрутизаторов RTT

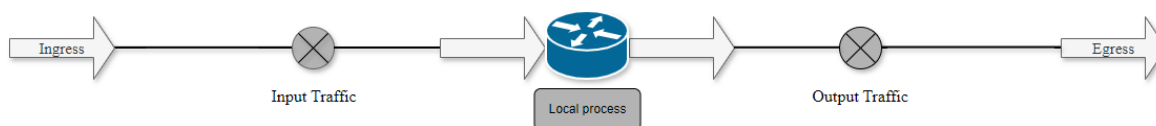


Таблица 1 – Порядок обработки входящего трафика

Шаг	Описание
1	Выполнение функций ACL на входящем трафике
2	Выполнение функций QoS (ограничение полосы пропускания, классификация и т. д.)
3	Выполнение функций DOS defense. На данном этапе выполняются функции защиты от DDOS из разделов firewall screen dos-defense, firewall screen spy-blocking, firewall screen suspicious-packets
4	Инспектирование пакета сервисом IDS/IPS в режиме service-ips monitor
5	Отключение функций Firewall командой ip firewall disable. Разрешение трафика исключает проверки на этапах 6, 13, 15
6	Выполнение правил между зонами any/self
7	Выполнение дефрагментации пакета
8	Выполнение начальных функций BRAS (инициализация соединений, сессий)
9	Выполнение HTTP/HTTPS прокси
10	Функции Destination NAT
11	Routing Decision (FIB)
12	Выполнение функций DOS defense. На этапе данном этапе выполняются специфичные функции защиты от DDOS из разделов firewall screen dos-defense, firewall screen spy-blocking, firewall screen suspicious-packets: ip firewall screen suspicious-packets large-icmp ip firewall screen dos-defense winnuke ip firewall screen spy-blocking port-scan
13	Выполнение правил между специальными зонами /self
14	Разрешение служебного трафика кластера
15	Передача пакета в DPI

Шаг	Описание
16	Передача пакета в Netflow/sFlow (Ingress)
17	IPsec (decode). После выполнения этого шага происходит переход к п.1

Таблица 2 – Порядок обработки исходящего трафика

Шаг	Описание
1	Local Policy Based Routing
2	Route Decision
3	Передача пакета в DPI
4	tcp adjust-mss
5	Netflow/sFlow (Egress)
6	BRAS (для исходящих пакетов)
7	Выполнение функций Source NAT
8	IPsec (encode)
9	Выполнение фрагментации пакетов
10	Выполнение функций QoS (ограничение полосы пропускания, классификация и т. д.)

22.2. Порядок обработки транзитного трафика сетевыми службами маршрутизаторов RTT



Таблица 3 – Порядок обработки транзитного трафика

Шаг	Описание
1	Выполнение функций ACL на входящем трафике
2	Выполнение функций QoS (ограничение полосы пропускания, классификация и т. д.)
3	Выполнение функций DOS defense. На данном этапе выполняются функции защиты от DDOS из раздела firewall screen dos-defense, firewall screen spy-blocking, firewall screen suspicious-packets
4	Отключение функций Firewall командой ip firewall disable. Разрешение трафика исключает проверки на этапах 5, 15, 16

Шаг	Описание
5	Выполнение правил между специальными зонами /any
6	Выполнение дефрагментации пакета
7	Выполнение начальных функций BRAS (инициализация соединений, сессий)
8	Разрешение трафика, исходящий порт которого voice-port. Разрешение трафика включается только в случае наличия настроенного voice-port.
9	Выполнение HTTP/HTTPS прокси
10	Функции Destination NAT
11	Policy Based Routing
12	Routing Decision (FIB)
Если пакет перед передачей необходимо обработать протоколом более высокого уровня, выполняются следующие действия:	
12.1	<p>Выполнение функций DOS defense.</p> <p>На данном этапе выполняются специфичные функции защиты от DDOS из разделов firewall screen dos-defense, firewall screen spy-blocking, firewall screen suspicious-packets:</p> <p>ip firewall screen suspicious-packets large-icmp</p> <p>ip firewall screen dos-defense winnuke</p> <p>ip firewall screen spy-blocking port-scan</p>
12.2	Передача пакета в DPI
12.3	Передача пакета в Netflow/sFlow (Ingress)
12.4	Передача пакета в Antispam
12.5	<p>IPsec (decode).</p> <p>После выполнения этого шага происходит переход к п.3</p>
13	tcp adjust-mss
14	<p>Выполнение функций DOS defense.</p> <p>На данном этапе выполняются специфичные функции защиты от DDOS из разделов firewall screen dos-defense, firewall screen spy-blocking, firewall screen suspicious-packets:</p> <p>ip firewall screen suspicious-packets large-icmp</p> <p>ip firewall screen dos-defense winnuke</p> <p>ip firewall screen spy-blocking port-scan</p>
15	Выполнение правил между специальными зонами, any/any

Шаг	Описание
16	Передача пакета в DPI
17	Разрешение трафика, исходящий порт которого voice-port. Разрешение трафика включается только в случае наличия настроенного voice-port.
18	Netflow/sFlow (Egress)
19	Инспектирование пакета сервисом IPS/IDS в режиме service-ips inline
20	BRAS (для исходящих пакетов)
21	Выполнение функций Source NAT
22	IPsec (encode)
Если необходимо шифрование, то после этого процесса, выполняются следующие операции:	
22.1	Передача пакета в DPI
22.2	tcp adjust-mss
22.3	Netflow/sFlow (Egress)
22.4	BRAS (для исходящих пакетов)
22.5	Выполнение функций Source NAT
23	Выполнение фрагментации пакетов
24	Выполнение функций QoS (ограничение полосы пропускания, классификация и т. д.)