

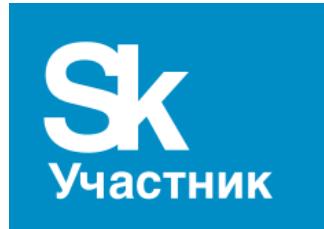


RDP.RU

EcoRouter User Guide

Руководство по установке и конфигурированию

Редакция: сентябрь 2018 г.



EcoRouter. User Guide

Руководство по установке и конфигурированию

Редакция: сентябрь 2018 г.

© РДП.РУ

Телефон: +7 (495) 204-9-204

<http://rdp.ru/>

Оглавление

Введение	13
Условные обозначения	14
Список терминов и сокращений.....	15
1 Оборудование.....	17
1.1 Нумерация интерфейсов (портов)	18
1.2 Просмотр информации о сетевых модулях	19
1.3 Поддерживаемые SFP модули	20
1.4 Мониторинг блоков питания.....	20
2 Общие сведения о работе с CLI	23
2.1 Подключение к EcoRouter	23
2.1.1 Консольный порт.....	23
2.1.2 Порт MGMT	23
2.2 Режимы работы консоли.....	23
2.3 Доступ к интерфейсу командной строки	25
2.4 Пароль на вход в режим администрирования	26
2.5 Сохранение конфигурации.....	27
2.6 Подсказки и горячие клавиши	27
2.7 Команды просмотра	28
2.8 Использование команды ping.....	29
2.9 Команда трассировки.....	31
2.10 Приветствие (banner motd)	33
3 Авторизация в системе.....	35
3.1 Вход в систему.....	35
3.2 Уровни доступа	35
3.3 Создание учетных записей пользователей.....	37
3.4 Команды просмотра	38
3.5 Аккаунтинг (Syslog)	38
3.6 Служебные пользователи	39
3.7 Настройки AAA	39
3.7.1 Приоритет авторизации	39
3.7.2 RADIUS	39
3.7.3 TACACS+	44
3.8 Профили безопасности	45
3.9 Инфраструктура открытых ключей	49

4	Виды интерфейсов.....	53
4.1	Порт	53
4.2	Агрегирование каналов.....	56
4.3	Интерфейс	56
4.4	Интерфейс loopback	57
4.5	Интерфейс demux	57
4.6	Bridge domain	58
4.7	Интерфейс bridge domain.....	58
4.8	Интерфейс PPPoE	58
4.9	Service Instance.....	59
4.10	Команды просмотра состояний интерфейсов	59
4.11	Команды просмотра SFP модулей	61
5	Сервисные интерфейсы	64
5.1	Виды инкапсуляции	64
5.1.1	Виды инкапсуляции	64
5.1.2	Команды настройки инкапсуляции	64
5.2	Операции над метками	65
5.2.1	Команды операций над метками.....	65
5.2.2	Направление движения трафика через сервисный интерфейс.....	65
5.2.3	Операции над метками в сервисных интерфейсах.....	66
5.3	Просмотр настроек сервисных интерфейсов.....	71
5.3.1	Просмотр всех сервисных интерфейсов на всех портах.....	71
5.3.2	Просмотр сервисных интерфейсов на отдельном порту	72
5.3.3	Просмотр сервисных интерфейсов по номеру	72
6	Агрегирование каналов.....	73
6.1	Вычисление хэш-функции	73
6.2	LACP.....	74
6.2.1	Настройка параметров	74
6.2.2	Команды просмотра	75
6.3	ECMP	77
6.4	Настройка Link aggregation	77
6.4.1	Именование агрегированных портов.....	77
6.4.2	Команды настройки агрегированного порта	77
6.4.3	Базовая настройка агрегированного порта. Способ 1	77
6.4.4	Базовая настройка агрегированного порта. Способ 2	78

6.4.5 Команды просмотра состояния агрегированного порта	78
7 Списки доступа	80
7.1 Policy-filter-list	80
7.1.1 Базовая конфигурация списка фильтров	81
7.1.2 Настройка фильтрации маршрутной информации в BGP	81
7.1.3 Настройка фильтрации маршрутной информации в IS-IS	82
7.1.4 Настройка фильтрации маршрутной информации в OSPF	85
7.2 Префиксные списки (prefix-list)	88
7.2.1 Команды просмотра списков префиксов.....	88
7.2.2 Настройка префиксных списков	89
7.3 Filter-map	90
7.3.1 Настройка L2 filter-map.....	91
7.3.2 Настройка L3 filter-map.....	93
7.3.3 Команды просмотра L2 filter-map	98
7.3.4 Команды просмотра L3 filter-map	100
7.3.5 Настройка политики для абонентской сессии	101
8 Виртуальные маршрутизаторы	106
8.1 Команды настройки виртуальных маршрутизаторов	106
8.2 Пример настройки виртуального маршрутизатора.....	107
8.3 Команды просмотра	108
9 Маршрутизация	110
9.1 Введение в маршрутизацию	110
9.2 Настройка статических маршрутов	110
9.2.1 Базовая настройка статических маршрутов	110
9.2.2 Административная дистанция статических маршрутов	111
9.3 Настройка RIP	111
9.3.1 Метрика RIP	111
9.3.2 Таймеры RIP	112
9.3.3 Split horizon	112
9.3.4 Функция ручной суммаризации маршрутов	112
9.3.5 Команды настройки.....	112
9.3.6 Пример базовой настройки.....	113
9.3.7 Включение протокола в виртуальном маршрутизаторе	114
9.3.8 Команды просмотра	114
9.4 Настройка OSPF	114

9.4.1	Пример настройки	115
9.4.2	Аутентификация	118
9.4.3	Фильтрация и суммаризация маршрутов OSPF	119
9.4.4	Маршрут по умолчанию	120
9.4.5	Зоны OSPF	120
9.4.6	Редистрибуция OSPF.....	121
9.4.7	Виртуальные линки и Multi-Area соседства	121
9.4.8	Команды просмотра OSPF	121
9.4.9	Дополнительные команды конфигурирования OSPF	122
9.4.10	Команды перезапуска процесса маршрутизации	122
9.4.11	Loop-Free Alternate (LFA) в OSPF	123
9.5	Настройка IS-IS	124
9.5.1	Пример настройки	125
9.5.2	Редистрибуция, фильтрация и суммаризация маршрутов.....	127
9.5.3	Маршруты по умолчанию и mesh-группы	127
9.5.4	Дополнительные команды конфигурирования.....	128
9.5.5	Команды просмотра	128
9.6	Настройка BGP	129
9.6.1	Базовая настройка BGP	129
9.6.2	BGP атрибуты	131
9.6.3	Команды конфигурирования атрибутов через route-map	132
9.6.4	Пример настройки BGP	134
9.6.5	Фильтрация и соседские отношения в BGP	136
9.6.6	Обновление партнерских BGP отношений	137
9.6.7	Регулярные выражения	139
9.6.8	Рефлекторы маршрутов и конфедерации.....	139
9.6.9	Команды конфигурирования BGP	140
9.6.10	BGP. Команды просмотра.....	143
9.6.11	Dampening	143
9.6.12	Background BGP scanners	144
9.6.13	Команды clear.....	145
9.6.14	BGP Blackhole	146
9.7	Карты маршрутов	148
9.7.1	Настройка карт маршрутов.....	148
9.7.2	Обработка записей в картах маршрутов	149

10	Настройка туннелирования	151
10.1	GRE.....	151
10.1.1	MTU в протоколах туннелирования	151
10.1.2	Флаги в GRE.....	151
10.1.3	Пример базовой настройки туннеля GRE	152
10.1.4	Команды просмотра	153
10.2	IP in IP	153
10.2.1	MTU в IP in IP	154
10.2.2	Флаги в IP in IP	154
10.2.3	Пример базовой настройки туннеля IP in IP	155
11	Бриджинг с поддержкой L3	157
11.1	Настройка.....	158
11.2	Создание BDI.....	159
11.3	Команды просмотра	159
12	Dynamic Host Configuration Protocol	161
12.1	Список команд.....	162
12.2	Базовая настройка.....	162
12.3	Команды просмотра состояния DHCP	164
13	ARP	165
14	Настройка IP Demux	167
14.1	Пример настройки IP Demux.....	168
14.2	Команды просмотра IP Demux	169
15	Мультикаст.....	170
15.1	IGMP	170
15.2	IGMP SSM Mapping.....	172
15.3	Proxy-IGMP	174
15.3.1	Настройка	174
15.4	PIM-SM/SSM.....	175
15.4.1	Команды просмотра	179
15.4.2	Командыброса данных	179
15.5	PIM-DM and mixed Sparse-Dense mode	180
16	Multiprotocol Label Switching.....	181
16.1	Настройка статического MPLS	181
16.2	LDP	182
16.3	Pseudowire	183

16.3.1	Настройка L2-circuit	183
16.3.2	Backup Pseudowire	186
16.4	Совместная работа BGP и MPLS	187
16.4.1	Топология.....	187
16.4.2	Конфигурация маршрутизаторов.....	187
16.4.3	MPLS карта	191
17	MPLS L3 VPN	193
17.1	Требования.....	193
17.2	MPLS VPN терминология	193
17.3	Процесс маршрутизации сетей VPN	194
17.4	Конфигурирование MPLS Layer-3 VPN.....	195
17.4.1	Топология.....	195
17.4.2	Включение коммутации по меткам	195
17.4.3	Включение IGP	196
17.4.4	Включение протокола коммутации меток	197
17.4.5	Настройка BGP-соседства между PE-маршрутизаторами.....	198
17.4.6	Создание VRF	198
17.4.7	Подключение интерфейсов к VRF.....	198
17.4.8	Настройка VRF-RD и целевых маршрутов	199
17.4.9	Конфигурация CE-соседей для VPN (с использованием BGP / OSPF / RIP)	
	200	
17.4.10	Проверка настройки MPLS-VPN	201
17.5	MPLS Layer-3 eBGP VPN Configuration.....	203
17.5.1	Настройка eBGP между PE и ASBR	204
17.5.2	Настройка eBGP между PE и RR и между ASBR.....	206
17.5.3	Соединение PE-маршрутизаторов с использованием eBGP Multi-hop ...	207
17.5.4	Соединение PE-маршрутизаторов с RR через RR, используя eBGP multi-hop	
	209	
18	Virtual Private LAN Service	212
18.1	Общие требования для работы VPLS (Martini)	213
18.2	Схема с одним PE, терминирующим L2-circuit.....	213
18.3	Схема с тремя PE, L2-circuit и Service-instance	214
18.4	Команды просмотра VPLS	215
18.5	Дополнительные настройки VPLS	216
19	VRRP.....	217
19.1	Базовая настройка	217

19.2	Дополнительные функции.....	217
19.2.1	Функция preempt-mode.....	218
19.2.2	Функция switch-back-delay.....	218
19.2.3	Функция circuit-failover	218
19.2.4	Функция accept-mode	218
19.2.5	Функция advertisement-interval	218
19.2.6	Функция vrrp vmac.....	218
19.3	Поддерживаемые версии протокола.....	219
19.4	Пример конфигурации	219
19.5	Известные особенности взаимодействия EcoRouter с оборудованием других производителей.....	221
20	BFD	223
20.1	Протокол BFD.....	223
20.2	Пример настройки single-hop BFD-OSPF	227
21	BRAS.....	229
21.1	Настройки IPoE.....	229
21.1.1	Динамический IPoE	233
21.1.2	Параметры IPoE	236
21.1.3	Логирование IPoE	238
21.1.4	Команды просмотра карт абонентов и сервисов абонентов.....	240
21.1.5	Функционал ARP Proxy	244
21.2	Настройки PPPoE	244
21.2.1	Протокол Point-to-Point (PPP).....	247
21.2.2	Пул IP-адресов	248
21.2.3	Аутентификация PPPoE	249
21.2.4	Особенность подключения PPPoE-абонента	249
21.2.5	Команда просмотра состояния PPPoE сессии.....	250
21.2.6	Параметры PPPoE при аутентификации через RADIUS-сервер	251
21.3	Общие настройки для BRAS	253
21.3.1	Настройка скорости доступа для абонентов BRAS	253
22	SNMP	256
22.1	Запуск и остановка сервиса SNMP	256
22.2	Настройка SNMP community.....	257
22.3	Настройка представлений (SNMP views).....	258
22.4	Настройка отправки асинхронных сообщений	258

22.5	SNMPv3	259
	22.5.1 Операции с пользователем	259
	22.5.2 Операции с группой	260
	22.5.3 Команды просмотра	261
23	QoS	262
23.1	Архитектура QoS.....	262
23.2	Классификация трафика	262
23.3	RED.....	265
23.4	Планировщик/Scheduler.....	267
	23.4.1 Настройка планировщика и очередей	269
23.5	Счетчики	271
23.6	Ограничение скорости	272
23.7	Маркировка трафика.....	273
23.8	Перемаркировка трафика	275
23.9	Сервисные политики.....	276
23.10	Профиль трафика	278
23.11	Карты классов.....	279
23.12	Ограничение входящего трафика по классам	280
24	Настройки зеркалирования.....	282
24.1	Mirror-session	282
24.2	Пример настройки зеркалирования.....	283
	24.2.1 Пример правила 1	284
	24.2.2 Пример правила 2	285
	24.2.3 Пример правила 3	286
24.3	Приостановка зеркалирования.....	287
24.4	Просмотр правил зеркалирования.....	287
25	Встроенный NAT	289
25.1	Пример конфигурации static source NAT	291
25.2	Пример конфигурации static source PAT	292
26	Операции с прошивкой	293
26.1	Скачивание образа прошивки	293
26.2	Установка скачанного образа прошивки	295
26.3	Действия после установки образа прошивки	296
26.4	Удаление образа прошивки	297
26.5	Выгрузка образа прошивки	297

26.6	Проверка целостности системных файлов	297
26.7	Сброс до factory	298
26.8	"Мягкий" сброс	298
27	Экспорт и импорт конфигурации.....	300
27.1	Подключение к серверу	300
27.2	Путь копирования	300
27.3	Архив конфигурации	300
27.4	Выбор интерфейса.....	301
27.5	Экспорт конфигурации	301
27.6	Импорт конфигурации	302
28	NTP.....	304
28.1	Базовая настройка.....	304
28.2	Команды просмотра NTP	305
29	Precision Time Protocol	307
	29.1.1 Команды просмотра	308
30	Flow export.....	310
30.1	Пример настройки	311
30.2	Команды просмотра	312
31	CoPP	314
31.1	Команды просмотра	315
32	Поток E1	317
32.1	Порты и каналы E1	317
	32.1.1 Настройка контроллера.....	318
	32.1.2 Настройка порта E1	319
	32.1.3 Настройка аутентификации	320
32.2	Настройка Multilink PPP	321
33	Виртуальные машины и контейнеры.....	323
33.1	Виртуальные машины и контейнеры. Общие сведения	323
33.2	Конфигурирование подключения интерфейсов виртуальной машины к EcoRouter	324
33.3	Конфигурирование доступа внешних средств управления контейнерами	325
33.4	Копирование виртуальных дисков	325
33.5	Распределение ядер между виртуальными машинами и data-plane	326
33.6	Подключение к виртуальной машине	326
	33.6.1 Подготовка клиентской машины	326

33.6.2 Конфигурирование доступа внешних средств управления виртуальной машиной	327
33.6.3 Управление гипервизором.....	327
33.7 Быстрая настройка виртуальных машин.....	328
34 Логирование и отладка	336
34.1 Логирование.....	336
34.2 Включение/выключение отладки	338
34.3 Архив логов	342
34.3.1 Просмотр архива логов	342
34.3.2 Удаление архива логов.....	342
34.3.3 Копирование архива логов на внешний сервер	342
ПРИЛОЖЕНИЕ А.....	344

Введение

В настоящем руководстве описан порядок установки и первичной настройки маршрутизатора EcoRouter (далее – EcoRouter).

Настоящее руководство действительно для встроенного программного обеспечения версии 3.2. Некоторые команды и значения параметров могут отличаться для более поздних или более ранних версий программного обеспечения. Для получения информации об актуальной версии программного обеспечения и документации обратитесь на сайт производителя <http://rdp.ru/> или в службу технической поддержки.

Рекомендации по настройке, сопровождающиеся словами «ВНИМАНИЕ», «ВАЖНО» и обведенные в двойную рамку, обязательны к исполнению для корректной работы оборудования и встроенного программного обеспечения. При невыполнении этих рекомендаций, EcoRouter может работать некорректно.

Условные обозначения

Для наглядности в тексте документации используются различные стили оформления. Области применения стилей указаны в Таблица 1.

Таблица 1 – Стили оформления в документе

Стиль оформления	Область применения	Пример
Полужирный шрифт	Названия элементов пользовательского интерфейса (команды, кнопки клавиатуры, символы консоли, рекомендуемые значения вводимых параметров)	Для создания правила зеркалирования используется команда: mirror-session <название> .
Шрифт Courier New	Примеры кода. Примеры вывода консоли	Устанавливаем связку порта и интерфейса L3. ecorouter(config-service-instance) #connect ip interface e1
Рамка, голубой цвет фона	Примеры вывода консоли	<p>В текущей конфигурации виртуального маршрутизатора находится только помещенный туда интерфейс.</p> <pre>ecorouter#show run ! no service password- encryption</pre>

В Таблица 2 приведены условные обозначения, используемые при описании консоли.

Таблица 2 – Условные обозначения при описании консоли

Условное обозначение	Расшифровка	Пример
Описание консоли		
< >	Пользовательские значения параметров	<часть команды>?
[]	Кнопки клавиатуры	<часть команды>[TAB]
Примеры		
Шрифт Courier New	Вывод консоли	<pre>ecorouter>en ecorouter#conf t Enter configuration commands, one per line. End with CNTL/Z.</pre>

Список терминов и сокращений

Сокращение	Расшифровка
AAA	Authentication, Authorization, Accounting
ACL	Access control list – списки контроля доступа
AS	Автономная система
ASN	Номер автономной системы
BA	Behavior Aggregation
BDI	Interface bridge domain – интерфейс bridge domain
BGP	Border Gateway Protocol
CIR	Committed Information Rate
CLI	Command Line Interface – интерфейс командной строки
DHCP	Dynamic Host Configuration Protocol
DSCP	DSCP поле заголовка IP пакета
ECMP	Equal-cost multi-path routing
EGP	Exterior Gateway Protocol
EXP	EXP поле заголовка MPLS пакета
FTP	File Transfer Protocol
GRE	Generic Routing Encapsulation
ICMP	Internet Control Message Protocol
IGP	Internal Gateway Protocol
IP	Internet Protocol
LACP	Link Aggregation Control Protocol
MED	Multi-Exit Discriminator
MP-BGP	Multiprotocol BGP
MPLS	Multiprotocol Label Switching
NTP	Network Time Protocol
OSPF	Open Shortest Path First
PDU	Protocol Data Unit
PIM	Protocol Independent Multicast
PIR	Peak Information Rate
RED	Random early detection
RID	Router ID
RIP	Routing Information Protocol

Сокращение	Расшифровка
RSVP	Resource ReSerVation Protocol
SI	Service Instance – сервисный интерфейс
SPAN	Switched Port Analyzer
TCP	Transmission Control Protocol
TFTP	Trivial File Transfer Protocol
TTL	Time to Live
UDP	User Datagram Protocol
UTC	Coordinated Universal Time
VLAN	Virtual Local Area Network
VRF	Virtual Routing and Forwarding
VRRP	Virtual Router Redundancy Protocol
OC	Операционная система

1 Оборудование

На рисунках ниже представлен вид передней панели оборудования серии EcoRouter. Модели представлены в следующем порядке:

- ER-116 (ER-110),
- ER-216,
- ER-1004,
- ER-2008.

У всех устройств серии на передней панели расположены:

- консольный порт RJ-45 с маркировкой COM,
- управляющий (management, менеджмент-) порт с маркировкой MNG,
- фиксированные сетевые интерфейсы,
- сетевые модули (интерфейсные карты),
- два USB-разъема,
- светодиоды индикации.

На передней панели "младших" моделей серии (ER-110, ER-116, ER-216) также расположен разъем кабеля питания. В случае если питание производится от сети переменного тока, там же расположена кнопка включения питания.

В модели ER-116 интерфейсы GE8-GE11 - оптические.

Сетевые интерфейсы "младших" моделей серии промаркованы (GE0-GE15, E1[1]-E1[4]).

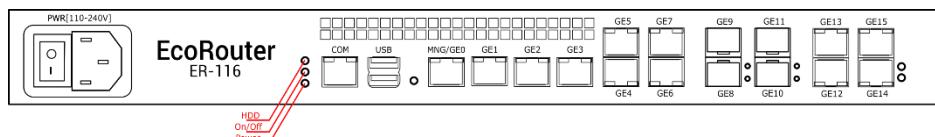


Рисунок 1

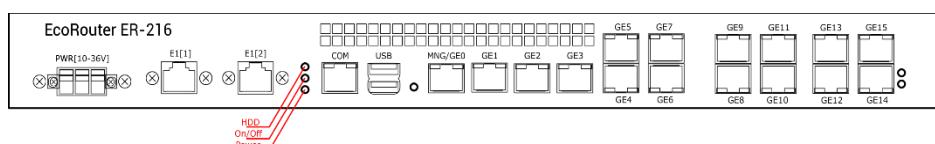


Рисунок 2

У "старших" моделей серии (ER-1004, ER-2008) разъемы кабеля питания и кнопка включения расположены на задней панели.

Нумерация сетевых модулей показана на рисунках ниже. В зависимости от установленных сетевых модулей, вид передней панели может отличаться.

Передняя панель ER-1004.

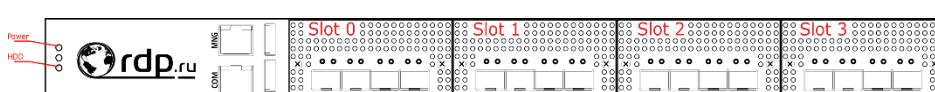


Рисунок 3

Передняя панель ER-2008.

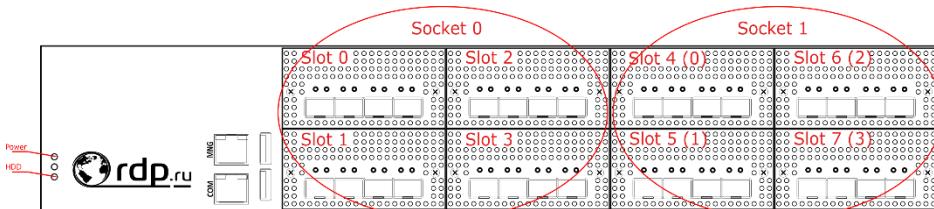


Рисунок 4

Модель ER-2008 работает на двух процессорах. Сетевые модули распределяются между процессорами (сокетами процессоров) группами по 4 модуля, как показано на рисунке выше.

Соответственно, сетевые модули в ER-2008 имеют двойную нумерацию:

- сквозную нумерацию от 0 до 7,
- нумерацию в пределах одного сокета от 0 до 3.

1.1 Нумерация интерфейсов (портов)

Поддерживаются сетевые интерфейсы с пропускной способностью 100Mbit, 1Gbit, 10Gbit, 40Gbit и 100Gbit.

В логике EcoRouter сетевые интерфейсы (L2) представлены объектами типа **port**.

Имена интерфейсов начинаются с префикса, зависящего от типа передатчика:

- feN – Fast Ethernet,
- geN – Gigabit Ethernet,
- teN – Ten Gigabit Ethernet,
- qeN – Quad Gigabit Ethernet,
- heN – Hundred Gigabit Ethernet,

где N – порядковый номер устройства (например: te0, ge3, fe1). Названия портов чувствительны к регистру и указываются только с маленькой буквы.

Для "младших" моделей название сетевых интерфейсов строится по принципу **<префикс><номер>**, например, ge2. Нумерация портов соответствует маркировке на передней панели устройства.

В "старших" моделях серии из-за их модульности применяются составные имена портов.

В ER-1004 название сетевых интерфейсов строится по принципу **<префикс><номер модуля>/<номер порта в модуле>**, например, te1/2. Где номер модуля изменяется в пределах от 0 до 3.

В ER-2008 название сетевых интерфейсов строится по принципу **<префикс><номер сокета>/<номер модуля в сокете>/<номер порта в модуле>**, например, te0/2/1, где номер сокета - 0 или 1. Номер модуля изменяется в пределах от 0 до 3.

На рисунке ниже показана нумерация портов в разных типах модулей.

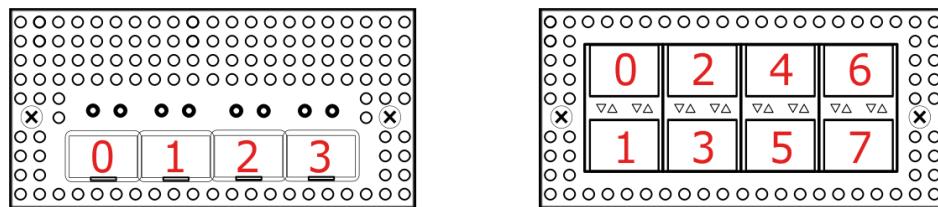


Рисунок 5

1.2 Просмотр информации о сетевых модулях

Для просмотра информации об установленных сетевых модулях (интерфейсных картах) используется команда административного режима **show platform inventory**.

Пример вывод команды для модели ER-1004.

```
ecorouter#show platform inventory
Item      Part number          Serial number      Description
-----
chassis   ER-1004-LBD          3.2.1.0.8859-develop-cee4202
slot0     NIC-8GE-TX
slot1     NIC-4XGE-SFPP
    te1/0:ML-SFP+DAC-V2-3      05G201511115480      Unspecified
    te1/1:ML-SFP+DAC-V2-3      X201601201111      Unspecified
    te1/2                      -----      SFF non-compatible
    te1/3                      -----      SFF non-compatible
slot2     empty
slot3     empty
```

Пример вывода команды для модели ER-2008.

```
ecorouter#show platform inventory
Item      Part number          Serial number      Description
-----
chassis   ER-2008              3.2.1.1.9218-merge-request-sfp-
fix-d9416e5
slot0     NIC-4XGE-SFPP
    te0/0/0                     -----      SFF non-compatible
    te0/0/1                     -----      SFF non-compatible
    te0/0/2                     -----      SFF non-compatible
    te0/0/3                     -----      SFF non-compatible
slot1     NIC-4XGE-SFPP
    te0/1/0                     -----      SFF non-compatible
    te0/1/1                     -----      SFF non-compatible
    te0/1/2                     -----      SFF non-compatible
    te0/1/3                     -----      SFF non-compatible
slot2     NIC-4XGE-SFPP
    te0/2/0                     -----      SFF non-compatible
    te0/2/1                     -----      SFF non-compatible
    te0/2/2                     -----      SFF non-compatible
    te0/2/3                     -----      SFF non-compatible
slot3     NIC-4XGE-SFPP
    te0/3/0                     -----      SFF non-compatible
```

te0/3/1	-----	SFF non-compatible
te0/3/2	-----	SFF non-compatible
te0/3/3	-----	SFF non-compatible
slot4 NIC-4XGE-SFPP		
te1/0/0	-----	SFF non-compatible
te1/0/1	-----	SFF non-compatible
te1/0/2:ML-SFP+DAC-V2-1	03G201605307001	Unspecified
te1/0/3	-----	SFF non-compatible
slot5 NIC-4XGE-SFPP		
te1/1/0	-----	SFF non-compatible
te1/1/1	-----	SFF non-compatible
te1/1/2	-----	SFF non-compatible
te1/1/3	-----	SFF non-compatible
slot6 empty		
slot7 NIC-4XGE-SFPP		
te1/3/0	-----	SFF non-compatible
te1/3/1	-----	SFF non-compatible
te1/3/2	-----	SFF non-compatible
te1/3/3	-----	SFF non-compatible

1.3 Поддерживаемые SFP модули

Поддерживаемое оборудование:

OptiCin SFP-LX.LC.20 (ER-116)

OptiCin SFP-PLUS-LX.LC.10 (ER-1004)

Gateray GR-SP10-M8503L-D (ER-116)

FiberXon FTM-3012C-SLG (ER-116)

D-link DEM-310GT (ER-116)

FiberTrade FT-100BASEFX-SGMII-M (ER-116)

FiberTrade FT-SFP-Cooper-10-1000 (ER-116)

FiberTrade SFP-Cooper-10-1000 (ER-116)

OEM PPH-1302-02C (ER-116)

OEM PPH-1302-02CD (ER-116)

Модели EcoRouter могут быть снабжены разным набором оптических портов (10, 100 MbE, 1, 10, 100 GbE). Поддерживается горячая замена оптических модулей, модули могут быть подключены или отключены после старта системы. При этом если в порт вставить модуль другой производительности (например, 10 GbE в порт 1 GbE), то работать он не будет, хотя может определяться системой.

1.4 Мониторинг блоков питания

Для отображения состояния работы блоков питания на устройстве используется команда административного режима **show platform power**. Корректная работа блока питания

обозначается статусом **ok**. Нерабочее состояние блока питания (если блок питания отключен от сети или вышел из строя) обозначается статусом **failed**.

Вывод команды для устройств с одним блоком питания:

```
ecorouter#show platform power
PSU is ok
```

Для платформ ER-116 и ER-216 "PSU is failed" выводится в случае, если один из сенсоров питания находится в состоянии ALARM.

Вывод команды для устройств с двумя блоками питания:

```
ecorouter#show platform power
PSU1 is ok
PSU2 is failed
```

Для просмотра информации о состоянии оборудования (напряжении, температуре, скорости вращения вентиляторов) используется команда административного режима **show platform sensors**. Для безвентиляторных платформ данная команда не будет отображать скорость вращения вентилятора.

Пример вывода команды:

```
ecorouter#show platform sensors
id | value | units | min | max | ALARM | description
1 | 1.79 | V | -inf | inf | NO | CPU VCORE
2 | 4.99 | V | -inf | inf | NO | +5V
3 | 11.88 | V | -inf | inf | NO | +12V
4 | 3.31 | V | -inf | inf | NO | +3.3V
5 | 3.26 | V | -inf | inf | NO | VBAT
6 | 3.31 | V | -inf | inf | NO | 3VSB
7 | 54 | C | -inf | inf | NO | CPU0
8 | 1 | C | -inf | inf | NO | CPU1
9 | 30 | C | -inf | inf | NO | MB
10 | 4232 | RPM | 1000.00 | inf | NO | FAN1
11 | 5294 | RPM | 1000.00 | inf | NO | FAN2
12 | 485 | RPM | 1000.00 | inf | YES | FAN3
13 | 5294 | RPM | 1000.00 | inf | NO | FAN4
14 | 4232 | RPM | 1000.00 | inf | NO | FAN5
15 | 5294 | RPM | 1000.00 | inf | NO | FAN6
16 | 4232 | RPM | 1000.00 | inf | NO | FAN7
17 | 5294 | RPM | 1000.00 | inf | NO | FAN8
```

Если значение параметра какого-либо из датчиков вышло за границы диапазона между минимальным и максимальным значениями (min и max соответственно), то в столбце ALARM в соответствующей строке будет выведено значение YES. В случае штатной работы в столбце ALARM отображается NO.

В таблице ниже описаны значения, выводимые данной командой **show platform sensors**.

Таблица 3

Параметр	Описание
CPU VCORE	Напряжение на процессоре. Предупреждение (ALARM) не выдается, потому что значение может сильно варьироваться от процессора, значение может завышать сама плата. Выводится для информации

Параметр	Описание
+12V	Напряжение 12 В на выходе блока питания. Предупреждение (ALARM) выдается, если значение отклоняется от допустимой нормы больше, чем на 10%
+5V	Напряжение 5 В на выходе блока питания. Предупреждение (ALARM) выдается, если значение отклоняется от допустимой нормы больше, чем на 10%
+3.3V	Напряжение 3,3 В на выходе блока питания. Предупреждение (ALARM) выдается, если значение отклоняется от допустимой нормы больше, чем на 5%
VBAT	Напряжение на батарее
3VSB	Дежурное напряжение
CPUn	Температура процессора. Предупреждение (ALARM) выдается, если температура превышает 90°C
MB	Температура материнской платы. Предупреждение (ALARM) выдается, если температура превышает 70°C
FANn	Скорость вращения вентилятора (обороты в минуту). Количество вентиляторов в выводе зависит от самой платформы (от 0 до 8-ми). Предупреждение (ALARM) выдается, если скорость вращения упала ниже 1000 RPM

Для принудительного сброса всех значений в столбце ALARM к NO используется команда **clear platform sensors**. Для сброса значения в столбце ALARM к NO для определенного сенсора используется команда **clear platform sensors <ID>**, где <ID> - порядковый номер сенсора (первый столбец в выводе команды **show platform sensors**).

ВНИМАНИЕ: сброс значения не влияет на работу самого оборудования. Если значение какого-либо параметра постоянно выходит за границы допустимого диапазона, необходимо провести диагностику оборудования.

Для отключения опроса определенного сенсора на предмет выхода параметров за допустимые значения (ALARM) используется команда **platform sensors alarm <ID> disable** или **no platform sensors alarm <ID> enable**, где <ID> - порядковый номер сенсора (первый столбец в выводе команды **show platform sensors**). Для включения опроса определенного сенсора используется команда **platform sensors alarm <ID> enable**.

2 Общие сведения о работе с CLI

Интерфейс командной строки (Command Line Interface, CLI) – основной интерфейс управления и мониторинга EcoRouter.

В этом разделе представлено общее описание интерфейса командной строки EcoRouter, основных команд, горячих клавиш и доступа к помощи.

2.1 Подключение к EcoRouter

Подключиться к маршрутизатору можно следующими способами:

- через консольный порт;
- через Ethernet-порт управления mgmt;
- через линейные Ethernet-порты.

Логин и пароль могут быть получены по запросу.

2.1.1 Консольный порт

Консольный порт (обычно самый левый порт 8P8C aka RJ45) имеет стандартное расположение контактов и совместим с консольными кабелями Cisco и других вендоров. Настройка порта: 115200 8N1 No flow control.

2.1.2 Порт MGMT

Порт управления mgmt (обычно левый порт в группе встроенных гигабитных ethernet-портов с маркировкой MNG/GE0) имеет IP-адрес по умолчанию 192.168.255.1/24. На управляющей машине предварительно необходимо настроить адрес из подсети 192.168.255.0/24 и использовать для доступа протокол ssh или telnet. Адрес порта mgmt можно впоследствии изменить командой **hw mgmt ip <адрес>**. Для настройки шлюза по умолчанию для mgmt-сети используйте команду **hw mgmt gw <адрес>**.

2.2 Режимы работы консоли

Интерфейс командной строки (CLI) – основной интерфейс управления и мониторинга EcoRouter.

EcoRouter даёт доступ к нескольким уровням командной строки. Каждый уровень характеризуется разными группами возможных команд.

Для удобства управления в EcoRouter разделены режимы пользовательского просмотра и режимы администрирования и конфигурации.

В таблице ниже описаны основные режимы, способы их включения и приглашения командной строки в этих режимах.

Таблица 4

Режим	Описание	Как попасть в режим	Приглашение командной строки
Пользовательский режим (user-exec)	Этот режим позволяет просматривать текущее состояние устройства, соединений, использовать сетевые утилиты	Подключиться к устройству	ecorouter>
Режим администрирования (enable-exec)	В этом режиме доступны те же команды, что и в пользовательском режиме, доступ в режим конфигурирования ОС и команды отладки	Ввести команду enable в приглашении командной строки, и пароль, если он установлен	ecorouter#
Режим конфигурации (config)	В режиме конфигурирования можно изменять и задавать настройки, которые влияют на работу устройства в целом	Ввести команду configure terminal , находясь в режиме администрирования	ecorouter(config)#
Контекстный режим (context-config)	В режиме конфигурирования многие структуры имеют несколько уровней конфигурации, при создании или входе в такую структуру (например, при создании интерфейса) пользователь попадает в контекстный режим конфигурирования. В этом режиме можно изменять настройки устройства	При вводе определенных команд в режиме конфигурирования	ecorouter(config-КОНТЕКСТ)#

При входе на устройство пользователь оказывается в режиме просмотра и видит приглашение командной строки в таком виде **ecorouter>**.

Чтобы переключиться в режим администрирования, нужно ввести команду **enable**, после чего приглашение командной строки изменит вид на **ecorouter#**. Чтобы отменить действие команды, нужно ввести команду **disable**.

Для переключения в режим конфигурирования нужно ввести команду **configure terminal**. После этого приглашение командной строки изменится на **ecorouter(config)#**. Для выхода из этого режима или с любого подуровня конфигурации используется команда **exit**.

```
EcoRouterOS version 3.0.0 EcoRouter 04/01/16 17:28:12
ecorouter>enable
ecorouter#configuration terminal
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#interface e3
ecorouter(config-if)#exit
ecorouter(config)#exit
ecorouter#
```

Чтобы закрыть активную сессию с устройством, дайте команду **logout** из режима просмотра.

```
ecorouter>logout
```

Разрыв сеанса или закрытие соединения автоматически приводит к потере всех несохраненных изменений в редактируемой конфигурации.

Большинство команд конфигурации можно отменить с помощью приставки **no**. Чтобы включить команду снова, нужно ввести её повторно без приставки **no**. Например, чтобы удалить созданный интерфейс, нужно дать команду **no interface e1**; чтобы создать его заново, нужно ввести команду **interface e1**.

2.3 Доступ к интерфейсу командной строки

Стандартно доступ к командной строке устройства осуществляется только через консольный и менеджмент порт, а для доступа к линейным портам по протоколам Telnet или SSH необходимо сконфигурировать профили безопасности (см. раздел "Профили безопасности").

В CLI маршрутизатора EcoRouter консольный порт обозначается как специализированная линия «con 0». Для ее настройки используется команда конфигурационного режима **line console 0**.

Устройство поддерживает до 872 одновременных сеансов по протоколам Telnet и SSH через менеджмент- и линейные порты, называемых виртуальными линиями (**vty**) и нумеруемых с 0 по 871.

Для настройки доступа по линейным портам используется команда конфигурационного режима **line vty <NUM | RANGE>**, где **NUM** - это номер конкретной линии, **RANGE** - диапазон номеров линий (значения указываются через пробел), к которым будут применены дальнейшие настройки. Команда переводит пользователя в режим конфигурирования виртуальных линий. Дальнейшие настройки будут использоваться как для Telnet, так и для SSH-сеансов.

Таким образом, команда **line vty 0 871** указывает маршрутизатору, что следующие за ней настройки будут применены к всем 872 виртуальным линиям, а команда **line vty 7** может настроить только 7 линию.

В режиме конфигурирования консоли и виртуальных линий доступны команды, приведенные в таблице ниже.

Таблица 5

Команда	Описание
exec-timeout <0-35791> <0-2147483>	Время ожидания. Если за указанный интервал времени в данной сессии на данной виртуальной линии (консоли) не будет произведено никаких действий, система автоматически завершит сеанс с сообщением типа "User is logged out by timeout" или "Vty connection is timed out". Для возобновления сеанса пользователю необходимо будет снова ввести свой логин и пароль. Сначала указывается количество минут, потом, через пробел, количество секунд при необходимости. При значении 0 маршрутизатор не будет отключать пользователей от соответствующей линии никогда. Значение по умолчанию - 10 минут

Команда	Описание
history max <0-2147483647>	Количество команд, которое будет сохраняться в буфере команд. Буфер доступен по нажатию клавиши стрелка вверх «↑». По умолчанию значение равно максимально возможному

Для просмотра информации о подключенных пользователях используется команда административного режима **show users** (данная команда доступна только для пользователей, которым назначена роль **admin**).

Пример вывода информации о подключенных пользователях:

```
ecorouter#show users
Line      User          Logged      Location      PID
 0  con 0    admin        00:00:03  ttyS0        1701
130 vty 0    admin        00:14:08  pts/0        1506
131 vty 1    admin        00:00:18  pts/1        1685
```

В выводе команды присутствуют следующие столбцы:

Line – названия линий,

User – имя пользователя, осуществившего вход в систему,

Logged – сколько времени прошло с момента подключения,

Location – внутренние обозначения линий,

PID – номер процесса.

2.4 Пароль на вход в режим администрирования

В EcoRouter существует возможность задать пароль на доступ к режиму администрирования (команда **enable**). Пароль задается командой конфигурационного режима **enable password**. Пароль может быть задан в явном виде или в виде хэша.

Для задания пароля в явном виде используется команда **enable password <PASS>**, где <PASS> - пароль. Пароль должен состоять из латинских букв и цифр. Максимальная длина пароля – 8 символов. Пароль должен начинаться с буквы. По умолчанию этот пароль будет записан в конфигурации маршрутизатора в открытом виде.

Пароль на доступ к режиму администрирования можно создать сразу в виде хэша при помощи команды конфигурационного режима **enable password 8 <hash>**, где **hash** – это уже зашифрованная алгоритмом DES (в формате Base64) строка пароля.

Для того чтобы снять пароль, достаточно ввести в конфигурационном режиме команду **no enable password** (без указания пароля).

В EcoRouter предусмотрена возможность хранения пароля в зашифрованном виде. Для этого используется алгоритм шифрования DES, и пароль записывается в конфигурационный файл маршрутизатора в виде DES-хэша.

Автоматическое шифрование пароля включается командой конфигурационного режима **service password-encryption**. После ввода данной команды записанный в конфигурации пароль шифруется, и так же будут шифроваться вновь создаваемые пароли.

При этом команда **no service password-encryption** выключает режим автоматического шифрования, но не расшифровывает уже созданный пароль.

```
ecorouter>enable  
Password:  
ecorouter#
```

2.5 Сохранение конфигурации

Команды, которые были даны в режиме конфигурации, вносят изменения в текущую конфигурацию. Изменения конфигурации вступают в силу после каждого нажатия клавиши **[Enter]** после ввода правильной команды. Эти изменения не сохраняются в файле конфигурации запуска до тех пор, пока не будет введена команда **write**. Если команда **write** дана не была, после перезагрузки устройства текущие изменения будут сброшены и не будут применяться.

У команды **write** есть несколько аргументов:

- **write file** или **write memory** – сохранение текущей конфигурации в файл;
- **write terminal** – вывод текущей конфигурации на экран, аналог команды **show running-config**.

```
ecorouter#write ?  
file      Write to file  
memory   Write to NV memory  
terminal  Write to terminal
```

2.6 Подсказки и горячие клавиши

В любом режиме доступна помощь по синтаксису команд. Чтобы просмотреть список всех доступных команд, введите знак вопроса в приглашении командной строки. Команды располагаются в алфавитном порядке.

```
ecorouter#?  
Exec commands:  
arp        IP ARP table  
clear     Reset functions  
configure  Enter configuration mode  
copy      Copy from one file to another  
debug     Debugging functions (see also 'undebbug')  
develop    Debug command  
disable    Turn off privileged mode command  
enable     Turn on privileged mode command
```

Чтобы посмотреть список всех доступных команд, начинающихся с определенных букв нужно ввести начало слова и знак вопроса.

```
ecorouter#co?  
configure  Enter configuration mode  
copy      Copy from one file to another
```

Чтобы просмотреть список существующих аргументов для команды, введите знак вопроса после команды.

```
ecorouter#configure?
terminal Configure from the terminal
```

Также можно задавать команды по начальным буквам. Количество начальных букв команды должно быть достаточным, чтобы можно было отличить одну команду от другой. Например, короткой записью для команды **show** будет **sh**. При такой записи также можно дополнить команду с начальных букв до конца слова с помощью клавиши **[Tab]** на клавиатуре.

Признаком успешно выполненной команды является приглашение командной строки. В случае если команда принята не была, появится сообщение об ошибке.

В любой момент можно использовать подсказки и горячие клавиши, представленные в таблице ниже.

Таблица 6

Команда/сочетание клавиш	Действие
?	Показывает перечень команд и/или аргументов, доступных в текущем контексте, а также подсказки по их назначению
<часть команды>?	Показывает перечень команд с таким началом
<часть команды>[ТАВ]	Пытается выполнить автозаполнение
стрелка вверх [↑]	Возврат к ранее введенной команде (история)
стрелка вниз [↓]	Возврат к команде, введенной позднее (история)

2.7 Команды просмотра

Для просмотра используются различные вариации команды **show** вида:

show <объект просмотра> <название объекта>

Такое представление команды **show** действует в административном режиме. Для того чтобы команда просмотра была принята в режиме конфигурации, перед командой должна быть приставка **do**:

do show <объект просмотра> <название объекта>

Пример:

```
ecorouter(config)#do show interface e1
Interface e1[15] is up, line protocol is up
Type: KNI
HW address 0000.abe1.b507
```

Для просмотра конфигурации в целом используется команда **show running-config** в административном или конфигурационном режиме.

Команды просмотра формируют вывод на экран блоками. Чтобы просмотреть следующий блок, необходимо нажать клавишу **[ПРОБЕЛ]**. Для выхода из режима просмотра используется клавиша **[Q]**.

Для удобства отображения вывода в консоль в EcoRouterOS поддерживаются фильтры, реализованные при помощи так называемых «модификаторов». Модификаторы вводятся после команды через символ ‘|’ (называемый «ріре»):

<команда просмотра> | <модификатор> <признак фильтрации>

Поддерживаемые модификаторы описаны в таблице ниже.

Таблица 7

Команда	Описание
include	Выводит строки, включающие заданный символ или группу символов
exclude	Выводит строки, исключающие заданный символ или группу символов
begin	Выводит строки, начинающиеся с заданного символа или группы символов
redirect	Отправляет вывод команды для сохранения в указанный файл

Рассмотрим пример работы модификаторов.

Вывод команды со статусами всех существующих интерфейсов:

```
ecorouter#show interface brief
Interface      Status      Protocol      Description
-----
qq1           up          up
89            up          up
t34           up          up
6              up          up
e3             up          up
```

Вывод команды только с интерфейсами, в названии которых содержится цифра 3:

```
ecorouter#show interface brief | include 3
t34           up          up
e3             up          up
```

Вывод команды с интерфейсами, в названии которых не содержится цифра 3:

```
ecorouter#show interface brief | exclude 3
Interface      Status      Protocol      Description
-----
qq1           up          up
89            up          up
6              up          up
```

Вывод команды с интерфейсами, название которых начинается на цифру 8:

```
ecorouter#show interface brief | begin 8
Interface      Status      Protocol      Description
-----
89            up          up
```

Чтобы отправить вывод команды в указанный файл, необходимо ввести:

```
ecorouter#show interface brief | redirect Text1.log
```

или (краткая форма выражения **redirect**)

```
ecorouter#show interface brief > Text1.log
```

2.8 Использование команды ping

Команда **ping** является общим способом поиска неисправностей в сетях. Команда использует протокол ICMP для отправки серии эхо-пакетов для определения, является ли удаленное оборудование активным, для определения времени задержек при передаче и для определения наличия потери пакетов. Данная утилита работает только из режима администрирования.

Стандартный вариант работы утилиты:

Общий вид команды:

```
ecorouter#ping xx.xx.xx.xx
ecorouter#ping ip xx.xx.xx.xx
ecorouter#ping mgmt xx.xx.xx.xx
```

Вариант команды **ping mgmt** используется для пинга сети через менеджмент-интерфейс.

Пример вывода:

```
ecorouter#ping ip 10.10.10.2
PING 10.10.10.2 (10.10.10.2) 56(84) bytes of data.
64 bytes from 10.10.10.2: icmp_seq=1 ttl=64 time=0.017 ms
64 bytes from 10.10.10.2: icmp_seq=2 ttl=64 time=0.016 ms
...
64 bytes from 10.10.10.2: icmp_seq=9 ttl=64 time=0.015 ms

--- 10.10.10.2 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8004ms
rtt min/avg/max/mdev = 0.015/0.018/0.023/0.005 ms
```

После запуска утилиты в таком виде запускается бесконечный **ping**. Он будет продолжаться до тех пор, пока не будет остановлен администратором. Прервать выполнение команды можно сочетанием клавиш **[Ctrl+z]** или **[Ctrl+c]**.

Расширенная версия утилиты **ping** даёт дополнительные возможности для диагностики. Например, позволяет изменить размер отправляемого пакета или указать альтернативный выходной интерфейс.

Для запуска расширенной версии нужно в приглашении командной строки ввести команду **ping** и нажать **[Enter]** на клавиатуре. В командной строке появится предложение ввести следующий аргумент, после которого нужно нажать **[Enter]**. Таким образом будет предложено заполнить все поля аргументов утилиты. В таблице ниже есть описание обязательных и необязательных для заполнения аргументов.

Таблица 8

Поле	Описание
Protocol [ip]:	Запрос поддерживаемого протокола. По умолчанию используется IP
Target IP address:	Запрос IP-адреса назначения. Если в качестве поддерживаемого протокола указан не протокол IP, введите здесь соответствующий адрес для указанного протокола. По умолчанию не используется
Name of the VRF :	Запрос указать имя VRF от которого будет осуществляться ping. По умолчанию не используется
Repeat count [5]:	Количество ping-пакетов до адреса назначения. Значение по умолчанию - 5
Datagram size [100]:	Размер ping-пакета (в байтах). По умолчанию: 100 байт
Timeout in seconds [2]:	Интервал времени ожидания. По умолчанию: 2 секунды. Запрос "ICMP-эхо" считается успешным, только если пакет ЭХО-ОТВЕТА получен до этого временного промежутка
Extended commands [n]:	Указывает на появление или отсутствие дополнительных команд. По умолчанию не используется
Broadcast [n]:	Указывает на то, что целевой ip-адрес является широковещательным. По умолчанию не используется

Общий вид исполнения **ping** с расширенными опциями.

```
ecorouter#ping  
Protocol [ip]: ip
```

Адрес, который требуется проверить.

```
Target IP address: 192.168.2.2  
Name of the VRF :  
Repeat count [5]:  
Datagram size [100]:  
Timeout in seconds [2]:  
Extended commands [n]:  
Broadcast [n]:  
PING 192.168.2.2 (192.168.2.2) 100(128) bytes of data.  
108 bytes from 192.168.2.2: icmp_seq=1 ttl=254 time=26.9 ms  
108 bytes from 192.168.2.2: icmp_seq=2 ttl=254 time=30.9 ms  
108 bytes from 192.168.2.2: icmp_seq=3 ttl=254 time=26.0 ms  
108 bytes from 192.168.2.2: icmp_seq=4 ttl=254 time=29.9 ms  
108 bytes from 192.168.2.2: icmp_seq=5 ttl=254 time=24.0 ms  
  
--- 192.168.2.2 ping statistics ---  
5 packets transmitted, 5 received, 0% packet loss, time 4003ms  
rtt min/avg/max/mdev = 24.001/27.606/30.998/2.571 ms
```

Команда выполнена успешно.

2.9 Команда трассировки

Команда **traceroute** используется для обнаружения путей следования пакета до адресов удаленных устройств, а также точек нарушения маршрутизации. Данная утилита работает только из режима администрирования.

Утилита отправляет по три пробных пакета UDP (User Datagram Protocol) на каждый из промежуточных узлов сети, через который проходит маршрут к удаленному хосту. Утилита ограничивает время прохождения пробного пакета по маршруту, используя параметр Time to live (TTL). С помощью TTL определяется количество переходов, которые нужно совершить пакету, чтобы достичь сети назначения. Параметр TTL увеличивается на 1 до тех пор, пока пакет не сможет достичь удаленный хост, или параметр TTL не достигнет максимального значения, равного 30.

Общий вид команды **traceroute**:

```
ecorouter#traceroute xx.xx.xx.xx
```

Стандартный вид вывода команды **traceroute**:

```
ecorouter#traceroute 192.168.2.2  
traceroute to 192.168.2.2 (192.168.2.2), 30 hops max, 60 byte packets  
1 192.168.1.1 (192.168.1.1) 11.955 ms 11.945 ms 11.941 ms  
2 192.168.2.2 (192.168.2.2) 22.933 ms 22.929 ms 22.927 ms  
ecorouter#
```

В этом выводе мы видим, что от устройства, откуда осуществляется команда, до адреса назначения существует только два маршрутизатора.

Расширенные возможности утилиты **traceroute**.

Для запуска расширенной версии нужно в приглашении командной строки ввести команду **traceroute** и нажать **[Enter]** на клавиатуре. В командной строке появится предложение ввести следующий аргумент, после которого нужно нажать **[Enter]**. Таким образом будет предложено заполнить все поля аргументов утилиты. В списке ниже есть описание обязательных и необязательных для заполнения аргументов.

Таблица 9

Поле	Описание
Protocol [ip]:	Запрос поддерживаемого протокола. По умолчанию используется IP
Target IP address:	Необходимо указать имя хоста или IP-адрес. Нет значения по умолчанию
Source address:	IP-адрес маршрутизатора, который будет использован в качестве адреса отправителя для тестирования. По умолчанию не используется
Name of the VRF :	Запрос указать имя VRF от которого будет осуществляться трассировка. По умолчанию не используется
Numeric display [n]:	По умолчанию имеется как символьическое, так и цифровое отображение; тем не менее можно отменить символьическое отображение
Timeout in seconds [2]:	Количество секунд ожидания ответа на тестовый пакет. Значение по умолчанию равно 2 секундам
Probe count [3]:	Число пробных пакетов, которые требуется отправить на каждом уровне TTL. Значение по умолчанию равно 3
Maximum time to live [30]:	Максимальное значение TTL, которое может использоваться. Значение по умолчанию – 30. Выполнение команды traceroute завершается при достижении точки назначения или данного значения
Port Number [33434]:	Порт назначения, используемый пробными сообщениями UDP. Значение по умолчанию – 33434

Пример:

```
ecorouter>enable
ecorouter#traceroute
Protocol [ip]: ip
```

Адрес, к которому выполняется трассировка.

```
Target IP address: 192.168.2.2
Source address: 10.10.10.1
Name of the VRF :
Numeric display [n]:
Timeout in seconds [2]:
Probe count [3]:
Maximum time to live [30]:
Port Number [33434]:
traceroute to 192.168.2.2 (192.168.2.2), 30 hops max, 60 byte packets
1 192.168.1.1 (192.168.1.1) 4.919 ms 4.908 ms 4.904 ms
2 192.168.2.2 (192.168.2.2) 25.902 ms 25.899 ms 25.896 ms
```

Трассировка успешно выполнена.

```
ecorouter#
```

2.10 Приветствие (banner motd)

При входе пользователя в CLI EcoRouter может отображаться текстовое сообщение – приветствие, называемое banner или message of the day (motd). Приветствие представляет собой текстовую строку и может быть изменено пользователем. Для этого необходимо ввести команду конфигурационного режима **banner motd {<text> | default}**, где **default** – это сообщение, установленное по умолчанию. Сообщение по умолчанию представляет собой строку с указанием установленной версии программного обеспечения EcoRouterOS.

Для просмотра установленного приветствия используется команда пользовательского режима **show banner motd**.

Для удаления приветствия используется команда конфигурационного режима **no banner motd**.

Для изменения сообщения следует ввести команду **banner motd** с новым текстом.

Пример настройки приветствия "Hello, World!!!".

```
ecorouter login: test
Password: example
User Access Verification
ecorouter>enable
Password: test
ecorouter#conf terminal
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#banner motd Hello, World!!!
ecorouter(config)#exit
ecorouter#exit
```

При следующем подключении и успешной аутентификации на экран будет выведено установленное сообщение. Ниже приведен пример удаления сообщения и возвращения к приветствию, установленному по умолчанию.

```
ecorouter login: test
Password: example
User Access Verification
Hello, World!!!
ecorouter>enable
Password: test
ecorouter#conf terminal
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#no banner motd
ecorouter(config)#exit
ecorouter#exit
ecorouter login: test
Password: example
User Access Verification

ecorouter>enable
Password: test
ecorouter#conf terminal
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#banner motd default
ecorouter(config)#exit
```

```
ecorouter#exit
ecorouter login: test
Password: example
User Access Verification
EcoRouterOS version 3.2.0 EcoRouter 06/21/16 09:20:13
ecorouter>
```

3 Авторизация в системе

AAA (от англ. *Authentication, Authorization, Accounting*) – используется для описания процесса предоставления доступа и контроля над ним.

- *Authentication* (автентификация) – сопоставление персоны (запроса) существующей учётной записи в системе безопасности. Осуществляется по логину, паролю или сертификату.
- *Authorization* (авторизация, проверка полномочий, проверка уровня доступа) – сопоставление учётной записи в системе (и персоны, прошёдшей аутентификацию) и определённых полномочий. В EcoRouter пользователям предоставляется несколько предопределенных уровней доступа к командам системы.
- *Accounting* (учёт) – слежение за потреблением ресурсов (преимущественно сетевых) пользователем. В accounting включается также и запись фактов получения доступа к системе (англ. *access logs*).

3.1 Вход в систему

При соединении с консолью управления EcoRouter пользователю предлагается ввести логин и пароль, соответствующие одной из учетных записей пользователей в системе.

По умолчанию существует учетная запись **admin** с ролью администратора (admin) и с паролем **admin**.

После верификации на консоль выводится версия системы и приглашение командной строки, в котором отображается hostname (в примере "ecorouter") и значок пользовательского режима консоли (в примере '>').

Пример:

```
<<< EcoRouter 3.2.0.0xxxxxxxxxxxxxx (x86_64) - ttys0 >>>
ecorouter login: admin
Password: |
User Access Verification
EcoRouterOS version 3.2.0 EcoRouter 06/29/16 15:35:53
ecorouter>
```

3.2 Уровни доступа

Для разграничения уровней доступа в EcoRouter используются роли пользователей.

Следующие варианты ролей являются предопределенными:

Таблица 10

Роль	Описание	Режимы консоли
admin	Администратор	пользовательский, администрирования, конфигурации
noc	Аудитор	пользовательский, администрирования
helpdesk	Поддержка	пользовательский

Для каждой роли доступен свой набор команд.

Список команд для каждой роли приведен в Справочнике команд.

Для просмотра подробной информации по имеющимся ролям и доступным для каждой роли командам и режимам используется функция административного режима **show role**.

Три предопределенные роли нельзя редактировать. Однако возможно создать новую роль с нужными параметрами.

Для того чтобы создать роль, используется команда конфигурационного режима **role <NAME> [based-on {admin | noc | helpdesk}]**. Здесь имя новой роли **<NAME>** - обязательный параметр. В результате выполнения команды **role <NAME>** будет создана роль, не содержащая никаких прав. Роль также можно создать на основе одной из предопределенных, тогда все команды и режимы, доступные для предопределенной роли, будут автоматически скопированы в новую роль. Первый вариант создания роли более удобен, если нужно создать роль с небольшим набором команд. Второй вариант создания роли (на основе предопределенных) более удобен, если необходимо создать роль с большим набором команд или набором команд, незначительно отличающимся от одной из предопределенных ролей.

Для редактирования созданной роли используется аналогичная команда конфигурационного режима **role <NAME>**.

В контекстном режиме редактирования созданной роли можно добавить описание для роли при помощи команды **description <DESCRIPTION>** и задать или изменить доступ к командам.

Для управления доступом используются команды **permit {config | context-config | enable-exec | user-exec} <COMMAND>**, чтобы добавить доступ, и **no permit {config | context-config | enable-exec | user-exec} <COMMAND>**, чтобы запретить доступ. По умолчанию, те команды, которые не разрешены для роли, - запрещены. В данных командах используются два обязательных параметра. Первый - это указание на режим работы CLI, к которому разрешается/запрещается доступ для роли (access level), где:

- **config** - конфигурационный режим,
- **context-config** - контекстный режим,
- **enable-exec** - административный режим,
- **user-exec** - пользовательский режим.

Второй обязательный параметр - **<COMMAND>** - имя команды. Если название команды состоит из нескольких слов, например, **banner motd**, допускается указывать только первое слово (**banner**). При добавлении команды автоматически добавляется допуск для этой же команды с префиксами **no** и **do** (обратная команда и включение данной команды в конфигурационном режиме). При удалении команды, аналогично, запрещается доступ к обратной команде и включению данной команды в конфигурационном режиме (префиксы **no** и **do**). Поэтому не рекомендуется отдельно вносить в список команды с префиксами!

Если необходимо добавить или удалить несколько команд, то для каждой строки **permit** вводится отдельно.

Пример:

```
ecorouter(config)# role myrole
ecorouter(config-role)# permit enable-exec copy
ecorouter(config-role)# no permit enable-exec copy
```

ВНИМАНИЕ: некоторые команды не могут быть добавлены в роль (доступны только в предустановленной роли admin). Подробнее это описано в разделе Справочник команд.

Для удаления роли в конфигурационном режиме используется команда **no role <NAME>**.

ВНИМАНИЕ! Все изменения и добавления ролей и пользователей применяются в системе только после выполнения команды write.

3.3 Создание учетных записей пользователей

Создать учетную запись пользователя можно только в режиме конфигурации. Для этого используется команда **username <NAME>**.

Далее в пользовательском режиме задаются параметры учетной записи пользователя. Команды, управляющие этими параметрами, описаны в таблице ниже.

Таблица 11

Команда	Описание
description <DESCR>	Добавить описание пользователя
no description	Удалить описание пользователя
password <PASS>	Задать пароль пользователя
no password	Очистить пароль пользователя
role {admin noc helpdesk}	Назначить пользователю предопределенную роль. Указывается одно из значений: admin, noc, helpdesk
no role {admin noc helpdesk}	Лишить пользователя роли
no custom-role <NAME>	
custom-role <NAME>	Назначить пользователю редактируемую роль. Если введенное имя роли не соответствует ни одной из созданных в конфигурации ролей, то автоматически будет создана "пустая" роль
vr <NAME>	Разрешить пользователю доступ к виртуальному маршрутизатору
no vr <NAME>	Запретить пользователю доступ к виртуальному маршрутизатору

ВНИМАНИЕ: пользователь, которому не назначено ни одной роли с правами, не сможет выполнять никаких действий.

Одному пользователю может быть одновременно назначено несколько ролей. Каждая роль может быть назначена нескольким пользователям одновременно.

Для удаления учетной записи пользователя используется команда конфигурационного режима: **no username <NAME>**.

Пример:

```
ecorouter(config)# username user1
ecorouter(config-user)# description sysadmin
ecorouter(config-user)# password administrator
ecorouter(config-user)# role admin
```

Кроме предустановленных ролей можно создать пользовательскую роль (см. предыдущий раздел). Для этого в настройке пользователя используется контекстная команда **custom-role <NAME>**.

Для удаления пользовательской роли используется команда **no custom-role <NAME>**.

В процессе авторизации роль пользователя может быть определена записью в локальной базе данных или получена с RADIUS/TACACS+ сервера. В случае если пользователь существует и в локальной базе пользователей на маршрутизаторе, и в базе пользователей RADIUS/TACACS+ сервера, роль будет определяться способом авторизации.

3.4 Команды просмотра

Для просмотра запущенных терминалов, а также ролей активных пользователей используется команда пользовательского режима **show users connected**. Подробнее данная команда описана в разделе "Общие сведения о работе с CLI".

```
ecorouter>show users connected
Line      User          Logged    Location   PID      Roles
 0 con 0    admin        00:00:15  ttyS0     1979      admin
130 vty 0   ecouser     00:00:00  pts/0     2090      admin_tes
```

Для просмотра учетных записей пользователей, имеющихся в базе данных EcoRouter, используется команда **show users localdb**.

```
ecorouter#show users localdb
User: admin
Description: Administrator User
VR:
pvr
Roles:
admin ''
User: daemon
Description: The user is used to get configuration data
VR:
pvr
Roles:
User: tacacs
Description: The user is used to make authorization through tacacs
VR:
pvr
Roles:
noc ''
```

Для данных команд доступны модификаторы и вывод в файл, как и для других команд **show**.

3.5 Аккаунтинг (Syslog)

Функции аутентификации осуществляются при помощи создания пользователей в локальной базе данных.

Функции авторизации реализуются путем привязки к пользователю конкретной роли с определенным набором команд, который может быть изменен по усмотрению пользователя.

Функции аккаунтинга реализованы через отправку лог-данных на удаленный сервер с помощью встроенных в маршрутизатор функций отправки сообщений стандарта Syslog (rsyslog). Команда настройки отправки Syslog-сообщений выглядит следующим образом: **rsyslog host <address> {mgmt | vr {default | <VR_NAME>}}**. Где **address** - это IP-адрес

сервера, на который будут отправляться логи. В свою очередь, сообщения могут отправляться через management-порт **mgmt** или виртуальный маршрутизатор **vr {default | <VR_NAME>}**, где параметр **VR_NAME** является именем виртуального маршрутизатора, а **default** подразумевает стандартный (невиртуализированный) маршрутизатор.

3.6 Служебные пользователи

По умолчанию в системе также существует служебный пользователь **tacacs** с ролью Аудитора (**noc**).

Когда пользователь аутентифицируется в EcoRouter через TACACS+, в системе этот пользователь будет аутентифицирован как **tacacs**. Соответственно, права пользователя при доступе через TACACS+ будут ограничены ролью соответствующего служебного пользователя. Например, если пользователь **admin** аутентифицируется в EcoRouter через TACACS+, то его доступ будет соответствовать роли Аудитора (**noc**), а не Администратора.

Для пользователя **tacacs** можно изменять роль или создать новую роль с нужным набором доступных команд, как и для обычного пользователя (см. "Уровни доступа").

В файлах логирования (см. "Syslog") будет фиксироваться и действительное имя пользователя, и служебное, если он аутентифицируется через TACACS+.

3.7 Настройки AAA

Для настройки AAA используются несколько команд конфигурационного режима, описанных ниже.

3.7.1 Приоритет авторизации

Для установки приоритета видов авторизации используется команда **aaa precedence <local | radius | tacacs>**.

В качестве параметров данной команды вводятся виды авторизации в порядке их приоритетности:

```
ecorouter(config)#aaa precedence radius local tacacs
```

RADIUS (англ. Remote Authentication in Dial-In User Service) – сетевой протокол , предназначенный для обеспечения централизованной аутентификации , авторизации и учёта (Authentication, Authorization, and Accounting, AAA) пользователей , подключающихся к различным сетевым службам . Используется, например, при аутентификации пользователей WiFi, VPN, в прошлом, dialup-подключений, и других подобных случаях. Описан в стандартах RFC 2058, RFC 2059, RFC 2865 и RFC 2866.

3.7.2 RADIUS

Группы RADIUS-серверов

Для авторизации на EcoRouter поддерживается использование групп RADIUS-серверов. Такая функциональность используется для настройки RADIUS для BRAS (авторизация и аккаунтинг должны производиться на различных RADIUS-серверах).

В текущей реализации допускается создание до 16 различных групп, каждая из которых может содержать до 16 RADIUS-серверов. При этом один и тот же сервер может принадлежать к нескольким группам одновременно.

Для создания группы RADIUS-серверов используется команда конфигурационного режима **radius-group <RADIUS_GROUP>**, где **<RADIUS_GROUP>** - имя создаваемой группы RADIUS-серверов. Если группа с указанным именем уже существует, а также после ее создания в результате выполнения команды будет автоматически произведен переход в контекстный режим конфигурации этой группы, префикс приглашения изменится на (config-radius-group).

Для удаления группы RADIUS-серверов используется команда конфигурационного режима **no radius-group <RADIUS_GROUP>**, где **<RADIUS_GROUP>** - имя удаляемой группы RADIUS-серверов.

В контекстном режиме конфигурации группы RADIUS-серверов оператор может отредактировать или удалить описание группы, настроить режим ее работы, изменить параметры выбранного RADIUS-сервера или удалить выбранный RADIUS-сервер из группы.

Для задания описания группы RADIUS-серверов используется команда контекстного конфигурационного режима (config-radius-group) **description <TEXT>**, где **<TEXT>** - строка описания.

Для удаления описания группы RADIUS-серверов используется команда контекстного конфигурационного режима (config-radius-group) **no description**.

Режим работы группы RADIUS-серверов

Для настройки режима работы группы RADIUS-серверов используется команда контекстного конфигурационного режима (config-radius-group) **mode <MODE>**, где **<MODE>** - режим.

Допустимые значения режима работы группы RADIUS-серверов:

- **active-standby** - для всех запросов используется RADIUS-сервер с наибольшим приоритетом в группе (минимальное значение параметра **priority**). Этот сервер является активным (active), остальные при этом находятся в режиме ожидания (standby). Если RADIUS-сервер с наибольшим приоритетом перестает отвечать на запросы, то запросы начинают поступать на следующий по приоритетности сервер. По истечении определенного периода времени производится попытка повторить отправку запросов на наиболее приоритетный сервер. Если такая попытка удачна, то он снова становится активным;
- **round-robin** - запросы распределяются между всеми RADIUS-серверами группы. Например, если группа состоит из 3 RADIUS-серверов, пришло 5 запросов от клиентов. 1-ый запрос отправляется на 1-ый сервер, 2-ой - на 2-ой сервер, 3-ий - на 3-ий сервер, 4-ый запрос - снова на 1-ый сервер, 5-ый на 2-ой и т.д.

Значение по умолчанию - **active-standby**.

Настройка таймеров

Для настройки таймеров для группы RADIUS-серверов в контекстном конфигурационном режиме (config-radius-group) используются следующие команды:

Таблица 12

Команда	Описание
request-max-tries <NUMBER>	Количество запросов, после отсутствия ответа на которые сервер будет считаться недоступным (DEAD). Значение по умолчанию - 3
request-timeout <INTERVAL>	Временной интервал между отправкой запросов в секундах. Значение по умолчанию - 3 секунды
dead-time-interval <MIN> <MAX>	<p>Временной интервал в секундах, в течение которого сервер будет находиться в состоянии DEAD. Задаются минимальное <MIN> и максимальное <MAX> значения. По умолчанию <MIN> - 15 секунд, <MAX> - 300 секунд. Допустимые значения <MIN> и <MAX> - от 0 до 65535.</p> <p>Принцип использования dead-time-interval</p> <p>После отсутствия ответа RADIUS-сервера на <NUMBER> запросов (параметр request-max-tries), ранее отмеченного как ACTIVE, такой сервер помечается как DEAD на период <MIN>, и роутер, посылающий запросы, перенаправляет их на резервный RADIUS-сервер внутри группы. По окончании этого интервала запросы будут вновь посланы на ставший неактивным RADIUS-сервер. Если он ответит, то вновь станет ACTIVE.</p> <p>Если RADIUS-сервер не ответит, то останется помеченным как DEAD.</p> <p>Интервал для такого его состояния будет увеличен на <MIN> (то есть после первой неудачной попытки интервал составит <MIN>, после второй - 2*<MIN>, после третьей - 3*<MIN> и т.д.). Так будет продолжаться до того момента, пока интервал назначения отметки DEAD не достигнет значения <MAX>. После этого попытки обращения к такому RADIUS-серверу будут делаться раз в интервал <MAX> до первого успешного перехода RADIUS-сервера в состояние ACTIVE.</p> <p>Если <MAX> не кратен <MIN>, то интервал станет равным <MAX> после первого его превышения в результате увеличения на очередной <MIN>.</p>

Настройка параметров отдельного сервера в группе

Для настройки параметров RADIUS-сервера в группе используется следующая команда контекстного конфигурационного режима (config-radius-group):

```
server A.B.C.D secret WORD ({priority <0-65535>|vrf <VRF>|source
A.B.C.D|auth-port <1-65535>|acct-port <1-65535>|coa-listen-port <1-
65535>} )
```

Все параметры, кроме IP-адреса и секретного ключа, опциональны и могут быть заданы в любом порядке. Если при вызове команды указан IP-адрес уже существующего RADIUS-сервера, то будут изменены его указанные параметры. Иначе будет создан RADIUS-сервер с указанным IP-адресом.

Параметры команды описаны в таблице ниже.

Таблица 13

Параметр	Описание
server A.B.C.D	IP-адрес RADIUS-сервера
priority <0-65535>	Приоритет RADIUS-сервера (актуально для режима active/standby). Чем меньше значение, тем выше приоритет
vrf <VRF>	Имя VRF, в котором задан IP-адрес RADIUS-сервера (значение по умолчанию - VRF текущего виртуального маршрутизатора)
source A.B.C.D	IP-адрес, который будет указан в качестве source в пакете запроса (по умолчанию - адрес интерфейса, с которого уходит запрос)

Параметр	Описание
auth-port <1-65535>	Порт для запросов аутентификации (значение по умолчанию 1812)
acct-port <1-65535>	Порт для запросов аккаунтинга (значение по умолчанию 1813)
secret <WORD>	Значение атрибута secret (по умолчанию не задан)
coa-listen-port <1-65535>	Порт, на основе которого на BRAS будет открыт сокет для обработки соа и disconnect запросов.

Для удаления RADIUS-сервера из группы используется команда контекстного конфигурационного режима (config-radius-group) **no server A.B.C.D (vrf <VRF>)!**.

Пример:

```
ecorouter(config)#radius-group test
ecorouter(config-radius-group)#server 3.3.3.2 secret 12121212
ecorouter(config-radius-group)#server 3.3.3.4 secret dsfsfsf
ecorouter(config-radius-group)#mode active-standby
ecorouter(config-radius-group)#description ABRACADABRA
?corouter(config-radius-group)#
RADIUS group commands:
  dead-time-interval  Specify a RADIUS servers dead time interval
  description        Redirect URL description
  exit               Exit from the current mode to the previous
mode
  help               Description of the interactive help system
  mode               Specify a RADIUS group mode
  no                Negate a command or set its defaults
  request-max-tries Specify a RADIUS servers max number of tries
to
  retransmit         retransmit a request
  request-timeout   Specify a RADIUS servers response waiting time
  server            Specify a RADIUS server
  show              Show running system information
ecorouter(config-radius-group)#server 3.3.3.3 vrf test source 12121212
```

Соответствующий фрагмент конфигурации будет иметь следующий вид:

```
!
radius-group test
description ABRACADABRA
mode active-standby
dead-time-interval 15 300
request-max-tries 3
request-timeout 3
server 3.3.3.2 secret 12121212 priority 10
server 3.3.3.4 secret dsfsfsf priority 20
server 3.3.3.3 secret fsfd priority 30 vrf test source 12121212
!
```

Настройка аутентификации при помощи RADIUS

Для аутентификации и/или аккаунтинга при помощи RADIUS необходимо указать, какой абонентский AAA-профиль должен для этого использоваться. Предварительно необходимо создать и настроить абонентский AAA-профиль.

Для создания абонентского AAA-профиля используется команда в конфигурационном режиме **subscriber-aaa <SUBSCRIBER_AAA>**, где **<SUBSCRIBER_AAA>** - имя

абонентского AAA-профиля. Если профиль с указанным именем уже существует, а также после его создания в результате выполнения команды будет автоматически произведен переход в контекстный режим конфигурации этого профиля, префикс приглашения изменится на (config-sub-aaa).

Для удаления абонентского AAA-профиля используется команда конфигурационного режима **no subscriber-aaa <SUBSCRIBER_AAA>**, где <SUBSCRIBER_AAA> - имя удаляемого абонентского AAA-профиля.

В контекстном режиме конфигурации абонентского AAA-профиля оператор может отредактировать или удалить описание профиля, указать группы RADIUS-серверов для аутентификации и/или аккаунтинга.

Для задания описания абонентского AAA-профиля используется команда контекстного конфигурационного режима (config-sub-aaa) **description <TEXT>**, где <TEXT> - строка описания.

Для удаления описания абонентского AAA-профиля используется команда контекстного конфигурационного режима (config-sub-aaa) **no description**.

Для установки режима аутентификации через RADIUS используется команда контекстного конфигурационного режима (config-sub-aaa) **authentication radius <RADIUS_GROUP>**, где <RADIUS_GROUP> - имя группы RADIUS-серверов.

Для установки режима аккаунтинга через RADIUS используется команда контекстного конфигурационного режима (config-sub-aaa) **accounting radius <RADIUS_GROUP>**, где <RADIUS_GROUP> - имя группы RADIUS-серверов.

Для использования настроенного профиля необходимо перейти в контекстный конфигурационный режим (config-subscriber-map) и выполнить команду **set aaa <SUBSCRIBER_AAA>**, где <SUBSCRIBER_AAA> - имя абонентского AAA-профиля для использования.

Пример:

```
ecorouter(config)#subscriber-aaa NEW_AAA
ecorouter(config-sub-aaa)#authentication
    radius RADIUS authentication
ecorouter(config-sub-aaa)#authentication radius
    RADIUS_GROUP RADIUS server group
ecorouter(config-sub-aaa)#authentication radius test
ecorouter(config-sub-aaa)#accounting radius test2
ecorouter(config-sub-aaa)#
Subscriber AAA commands:
    accounting Subscriber AAA profile accounting method
    authentication Subscriber AAA profile authentication method
    description Subscriber AAA profile description
    exit Exit from the current mode to the previous mode
    help Description of the interactive help system
    no Negate a command or set its defaults
    show Show running system information
ecorouter(config-sub-aaa)#[/pre]
```

3.7.3 TACACS+

TACACS+ (англ. Terminal Access Controller Access Control System plus) – сеансовый протокол, результат дальнейшего усовершенствования TACACS, предпринятого Cisco.

Улучшена безопасность протокола (шифрование), а также введено разделение функций аутентификации, авторизации и учёта, которые теперь можно использовать по отдельности.

TACACS+ использует понятия сеансов. В рамках TACACS+ возможно установление трёх различных типов сеансов AAA (англ. authentication, authorization, accounting). Установление одного типа сеанса в общем случае не требует предварительного успешного установления какого-либо другого. Спецификация протокола не требует для открытия сеанса авторизации открыть сначала сеанс аутентификации. Сервер TACACS+ может потребовать аутентификацию, но протокол этого не оговаривает.

Команда **aaa tacacs-config debug** включает выгрузку отладочной информации TACACS в формате syslog.

```
ecorouter(config)#aaa tacacs-config debug
```

Если в параметрах сервера указан ключ шифрования, то информация в логах будет также зашифрована.

Если используется несколько серверов, то по умолчанию запросы будут отправляться до первого ответившего сервера. Для того чтобы опрашивались все серверы, необходимо включить опцию **aaa tacacs-config acct-all**.

```
ecorouter(config)#aaa tacacs-config acct-all
```

Для настройки TACACS-сервера используется команда **aaa tacacs-server**.

Синтаксис команды: **aaa tacacs-server <IP> port <NUM> secret <PASS> (mgmt | vr) (account | auth) timeout <0-300>**.

Параметры команды представлены в таблице ниже.

Таблица 14

Параметр	Описание
<IP>	IP-адрес TACACS-сервера
port <NUM>	Указать порт
secret <PASS>	Ключ шифрования. Если указан, шифрование будет автоматически включено
mgmt	Соединение через management-порт
vr (NAME default)	Соединение через виртуальный маршрутизатор. Можно указать имя виртуального маршрутизатора или выбрать маршрутизатор по умолчанию
account	Разрешить учет (аккаунтинг)
auth	Разрешить аутентификацию и авторизацию
timeout	Установить таймаут в секундах. Допустимые значения от 0 до 300 секунд

Пример:

```
ecorouter(config)#aaa tacacs-server 192.168.0.1 port 80 mgmt timeout 200
account auth
```

3.8 Профили безопасности

Для фильтрации принимаемого EcoRouter трафика используются так называемые профили безопасности. Профиль безопасности представляет собой набор правил, определяющих, пакеты каких протоколов будут пропускаться маршрутизатором (и виртуальными маршрутизаторами в его составе).

Для того чтобы создать профиль безопасности необходимо в режиме конфигурации ввести команду **security-profile <номер>**. В качестве названия профиля задается его порядковый номер.

Внутри профиль безопасности содержит правила, определяющие доступ к системе.

Для задания правила используется команда **rule <0-1023> [permit | deny] <PROTOCOL> <SOURCE> <DESTINATION> (<DEST PORT> <DP NUMBER>)**. Параметры команды описаны в таблице ниже.

Таблица 15

Параметр	Описание
<0-1023>	Порядковый номер правила, от 0 до 1023. Правила применяются, начиная с 0 по 1023.
permit deny	Тип правила: разрешить (permit) или запретить (deny)
PROTOCOL	Пакеты какого протокола подпадают под это правило. Может быть указан номер протокола по спецификации IANA от 0 до 255 или одно из следующих обозначений: any - пакеты любого протокола, gre - GRE пакеты, icmp - ICMP пакеты, igmp - IGMP пакеты, ip - пакеты с IPv4 инкапсуляцией, ipcomp - IPCOMP пакеты, ospf - OSPF пакеты, pim - PIM пакеты, rsvp - RSVP пакеты, tcp - TCP пакеты, udp - UDP пакеты, vrrp - VRRP пакеты
SOURCE	IP-адрес источника с длиной маски. Задается в виде A.B.C.D/M . Если под правило должны попадать все адреса, значение параметра должно быть any . Если под правило должен подпадать единственный адрес, в значении параметра указывается host <IP-адрес>
DESTINATION	IP-адрес назначения с длиной маски. Задается в виде A.B.C.D/M . Если под правило должны попадать все адреса, значение параметра должно быть any . Если под правило должен подпадать единственный адрес, в значении параметра указывается host <IP-адрес>
Фильтрация по порту назначения, доступно для протоколов TCP и UDP	
DEST PORT	Вариант фильтрации. Указывается одно из следующих обозначений: eq - номер порта равен ..., gt - номер порта больше, чем ..., lt - номер порта меньше, чем ..., range - номер порта находится в диапазоне ...
DP NUMBER	Номер или обозначение порта. Возможные значения для TCP: номер порта от 0 до 65535, ftp - FTP (21 порт),

Параметр	Описание
	<p>ssh - SSH (22 порт), telnet - Telnet (23 порт), www - WWW (HTTP, 80 порт). Возможные значения для UDP: номер порта от 0 до 65535, bootp - BOOTP (67 порт), tftp - TFTP (69 порт). Если задается диапазон портов (range), то нижняя и верхняя граница диапазона указываются числами через пробел.</p>

Если трафик не подпадает ни под одно из правил, то он пропускается (permit).

В EcoRouter существует жестко заданный профиль по умолчанию. Изменить его нельзя.

Состав профиля по умолчанию:

```
Security profile default
0: deny tcp any any eq 22
1: deny tcp any any eq 23
2: deny tcp any any eq 161
3: deny udp any any eq 22
4: deny udp any any eq 23
5: deny udp any any eq 161
```

Для management порта по умолчанию разрешены все протоколы.

Для того чтобы назначить созданный профиль безопасности на management порт, используется команда конфигурационного режима **security <SP_NAME> vrf management**, где SP_NAME - имя профиля. Для того чтобы назначить созданный профиль безопасности на VRF, по умолчанию используется команда конфигурационного режима **security <SP_NAME>**. Для того чтобы назначить созданный профиль безопасности на произвольную VRF, используется команда конфигурационного режима **security <SP_NAME> vrf <NAME>**, где NAME - имя VRF.

Для того чтобы назначить профиль безопасности виртуальному маршрутизатору, необходимо войти в виртуальный маршрутизатор. После чего в конфигурационном режиме виртуального маршрутизатора выполнить команды, аналогичные описанным выше.

Для того чтобы отвязать профиль безопасности от VRF или менеджмент порта, используется аналогичная команда с префиксом **no**. После этого к VRF или менеджмент порту применяется пустой профиль безопасности с названием **security none**.

Для удаления всех правил для VRF или менеджмент порта можно назначить пустой профиль безопасности с названием **security none**.

После назначения профиля безопасности его нельзя менять. Чтобы изменить профиль безопасности, его нужно вначале отвязать от VRF и/или менеджмент порта, которым он назначен.

Для корректной работы рекомендуется сначала отвязывать от виртуального маршрутизатора профиль безопасности, а потом удалять сам маршрутизатор.

Для просмотра настроенных профилей безопасности используется команда административного режима **show security-profile**.

Для просмотра текущих настроек безопасности используется команда административного режима **show ip vrf**.

Пример настройки профиля безопасности

Создание нового профиля.

```
ecorouter(config)#security-profile 1
ecorouter(config-security-profile)#rule 0 permit tcp any any eq 23
ecorouter(config-security-profile)#rule 1 deny udp any any eq bootp
ecorouter(config-security-profile)#rule 2 deny ospf host 127.0.0.12 any
ecorouter(config-security-profile)#rule 3 deny tcp any 192.168.10.2/24
range 21 23
ecorouter#show security-profile
Security profile default
0: deny tcp any any eq 22
1: deny tcp any any eq 23
2: deny tcp any any eq 161
3: deny udp any any eq 22
4: deny udp any any eq 23
5: deny udp any any eq 161
Security profile 1
0: permit tcp any any eq 23
1: deny udp any any eq 67
2: deny ospf 127.0.0.12/32 any
3: deny tcp any 192.168.10.2/24 range 21 23
```

Создание VRF и назначение ему профиля безопасности.

```
ecorouter(config)#ip vrf vrf0
ecorouter(config-vrf)#end
ecorouter#show ip vrf
VRF default
Interfaces:
Security profile default
0: deny tcp any any eq 22
1: deny tcp any any eq 23
2: deny tcp any any eq 161
3: deny udp any any eq 22
4: deny udp any any eq 23
5: deny udp any any eq 161
permit any any any

VRF management

VRF vrf0
Interfaces:
ecorouter(config)#security 1 vrf vrf0
ecorouter(config)#end
ecorouter#show ip vrf
VRF default
Interfaces:
Security profile default
```

```

0: deny tcp any any eq 22
1: deny tcp any any eq 23
2: deny tcp any any eq 161
3: deny udp any any eq 22
4: deny udp any any eq 23
5: deny udp any any eq 161
permit any any any

```

VRF management

```

VRF vrf0
  Interfaces:
    Security profile 1
      0: permit tcp any any eq 23
      1: deny udp any any eq 67
      2: deny ospf 127.0.0.12/32 any
      3: deny tcp any 192.168.10.2/24 range 21 23
      permit any any any

```

Внесение изменений в профиль безопасности.

```

ecorouter(config)#security-profile 1
ecorouter(config-security-profile)#rule 4 permit any any any
% Profile is set on 1 namespaces. Unset profile prior to change it.
ecorouter(config-security-profile)#ex
ecorouter(config)#no security 1 vrf vrf0
ecorouter(config)#security-profile 1
ecorouter(config-security-profile)#rule 4 permit any any any
ecorouter(config-security-profile)#ex
ecorouter(config)#ex
ecorouter#show security-profile
Security profile default
  0: deny tcp any any eq 22
  1: deny tcp any any eq 23
  2: deny tcp any any eq 161
  3: deny udp any any eq 22
  4: deny udp any any eq 23
  5: deny udp any any eq 161
Security profile 1
  0: permit tcp any any eq 23
  1: deny udp any any eq 67
  2: deny ospf 127.0.0.12/32 any
  3: deny tcp any 192.168.10.2/24 range 21 23
  4: permit any any any
  permit any any any

ecorouter#conf t
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#security 1 vrf vrf0
ecorouter(config)#end
ecorouter#show ip vrf
VRF default
  Interfaces:

```

```

Security profile default
0: deny tcp any any eq 22
1: deny tcp any any eq 23
2: deny tcp any any eq 161
3: deny udp any any eq 22
4: deny udp any any eq 23
5: deny udp any any eq 161
permit any any any

VRF management

VRF vrf0
Interfaces:
Security profile 1
0: permit tcp any any eq 23
1: deny udp any any eq 67
2: deny ospf 127.0.0.12/32 any
3: deny tcp any 192.168.10.2/24 range 21 23
4: permit any any any
permit any any any

```

Удаление профиля безопасности.

```

ecorouter(config)#no security 1 vrf
ecorouter(config)#no ip vrf vrf0
ecorouter(config)#end
ecorouter#show ip vrf
VRF default
Interfaces:
Security profile default
0: deny tcp any any eq 22
1: deny tcp any any eq 23
2: deny tcp any any eq 161
3: deny udp any any eq 22
4: deny udp any any eq 23
5: deny udp any any eq 161
permit any any any

VRF management
ecorouter#

```

3.9 Инфраструктура открытых ключей

В EcoRouterOS для обеспечения безопасности соединения пользователей используется протокол TLS (*Transport Layer Security* – безопасность транспортного уровня) на основе инфраструктуры открытых ключей (PKI) и сертификатов X.509. Установка безопасного соединения между сервером и клиентом происходит совместно с процессом аутентификации клиента на сервере. EcoRouter при этом выполняет роли Центра сертификации (Certificate Authority - CA) и сервера.

Таким образом при подключении к EcoRouter устройство отправляет пользователю сообщение, содержащее сертификат маршрутизатора и запрос сертификата пользователя. Пользователь, в свою очередь, отправляет сообщение, содержащее его сертификат, после чего

устанавливается безопасное соединение. При таком соединении вся информация, передающаяся между пользователем и устройством, шифруется при помощи закрытого ключа (Private Key). При передаче сообщения маршрутизатором сообщение шифруется закрытым ключом маршрутизатора таким образом, что расшифровать его пользователь может при помощи имеющегося у него открытого ключа (сертификата маршрутизатора). И наоборот, пользователь отправляет сообщения, зашифрованные при помощи закрытого ключа пользователя, которые EcoRouter расшифровывает при помощи переданного ему в начале сессии сертификата пользователя. Для того чтобы организовать этот процесс, у пользователя и EcoRouter должен быть идентичный набор сертификатов и специфический набор закрытых ключей.

Закрытый ключ и сертификат сервера автоматически генерируются в прошивке EcoRouter.

Закрытый ключ и сертификат пользователя генерируются EcoRouter при создании пользователя. При этом EcoRouter выступает в качестве CA, то есть сервера, отвечающего за регистрацию пользователей, обеспечивающего выпуск ключей, хранение реестра выданных ключей и проверку их статуса.

Таким образом для взаимодействия с маршрутизатором по защищенному соединению у пользователя должны храниться: сертификат EcoRouter (CA), сертификат пользователя, закрытый ключ пользователя.

EcoRouter также автоматически генерирует несколько служебных сертификатов для соединения с TACACS и RADIUS серверами.

Для просмотра пользовательских сертификатов в EcoRouter есть несколько команд, по умолчанию доступных только пользователям с ролью admin.

Для просмотра пользовательских сертификатов используется команда административного режима **crypto certificate export**. Для нее доступны модификаторы, при помощи которых можно отфильтровать вывод по конкретным пользователям. Например, исключить из вывода служебные сертификаты пользователей **radius** и **tacacs**.

В приведенном ниже примере сокращен вывод самих сертификатов. Все сертификаты хранятся и выводятся на консоль в кодировке Base64.

```
ecorouter#crypto certificate export
User: admin
Certificate: Valid
-----BEGIN CERTIFICATE-----
ESTCCA...gAyhj
-----END CERTIFICATE-----
User: radius
Certificate: Valid
-----BEGIN CERTIFICATE-----
ESzC...101Bt18=
-----END CERTIFICATE-----
User: tacacs
Certificate: Valid
-----BEGIN CERTIFICATE-----
E...j7tDSM=
-----END CERTIFICATE-----
```

Для экспорта (вывода на экран) закрытого ключа пользователя используется команда административного режима **crypto key export**. Данная команда выводит закрытый ключ того пользователя, который аутентифицирован в системе на данный момент.

В приведенном ниже примере сокращен вывод самого ключа. Все ключи хранятся и выводятся на консоль в кодировке Base64. Закрытые ключи должны передаваться на пользовательские компьютеры безопасным образом, исключающим возможность их получения третьими лицами.

```
ecorouter#crypto key export
User: admin
-----BEGIN RSA PRIVATE KEY-----
IEp...kjUcAQLyrg==
-----END RSA PRIVATE KEY-----
```

Для экспорта (вывода на экран) сертификата EcoRouter (CA) используется команда административного режима **crypto ca export**. Данная команда выводит сертификат сервера вместе с представленными в явном виде полями, такими как поле имени сервера - **Subject: CN=ecorouter**, подписью сервера и самим сертификатом.

В приведенном ниже примере сокращен вывод самого сертификата и подписи сервера. Сертификат CA хранится в базе данных на маршрутизаторе и выводится на консоль в кодировке Base64, а информация о нем - в текстовом виде.

```
ecorouter#crypto ca export
Certificate:
Data:
Version: 3 (0x2)
Serial Number:
9a:14:57:6d:84:76:e9:31
Signature Algorithm: sha256WithRSAEncryption
Issuer: CN=ecorouter
Validity
Not Before: Oct 4 08:17:55 2016 GMT
Not After : Oct 5 08:17:55 2026 GMT
Subject: CN=ecorouter
Subject Public Key Info:
Public Key Algorithm: rsaEncryption
Public-Key: (4096 bit)
Modulus:
00:c3:db:b8:b1:a7:a1:4b:34:82:af:1b:df:6a:2e:
...
0b:49:95
Exponent: 65537 (0x10001)
X509v3 extensions:
X509v3 Subject Key Identifier:
EA:DC:87:08:D8:03:AB:BB:44:C4:80:A1:58:38:91:45:16:E8:53:0A
X509v3 Authority Key Identifier:
keyid:EA:DC:87:08:D8:03:AB:BB:44:C4:80:A1:58:38:91:45:16:E8:53:0
A
X509v3 Basic Constraints:
CA:TRUE
Signature Algorithm: sha256WithRSAEncryption
ac:57:98:1f:5f:00:fa:80:d1:cc:fe:c6:e5:50:06:ff:14:d6:
```

```
...
37:a7:ad:8f:2d:99:1a:0c
-----BEGIN CERTIFICATE-----
MIIE+z...kaDA==
-----END CERTIFICATE-----
```

Для того чтобы экспорттировать выведенные на экран сертификаты и ключ, необходимо скопировать их в файлы с соответствующими названиями:

- cacert.pem - сертификат EcoRouter (CA),
- clientcert.pem - сертификат пользователя,
- clientkey.pem - закрытый ключ пользователя.

Копировать вывод закрытого ключа и сертификата открытого ключа пользователя необходимо от символов "-----BEGIN" до последнего дефиса в строке "-----END CERTIFICATE-----" (или "-----END RSA PRIVATE KEY-----"). Копировать сертификат CA необходимо, начиная с строки "Certificate:".

На пользовательском устройстве эти файлы должны быть размещены в директориях, используемых клиентским программным обеспечением. Для Unix/Linux по умолчанию это:

- /etc/pki/CA/cacert.pem
- /etc/pki/libvirt/private/clientkey.pem
- /etc/pki/libvirt/clientcert.pem

4 Виды интерфейсов

4.1 Порт

Порт (port) – это устройство в составе EcoRouter, которое работает на уровне коммутации. Выходы портов расположены на передней панели маршрутизатора.

Логика именования и нумерации портов описана в разделе Оборудование.

Названия портов чувствительны к регистру и указываются только с маленькой буквы.

По умолчанию все порты на устройстве включены.

Ниже приведены базовые команды настройки порта.

Переход на уровень конфигурации определенного порта. Где `te1` – его имя:

```
ecorouter(config)#port te1
```

Выставление значения `mtu` отличного от стандартного в диапазоне 1504-9728.
Необязательная настройка.

```
ecorouter(config-port)#mtu 1600
```

MTU (maximum transmission unit) означает максимальный размер полезного блока данных одного пакета (payload), который может быть передан протоколом без фрагментации. Когда говорят об MTU, обычно имеют в виду протокол канального уровня сетевой модели OSI.

Значение MTU для многих сетевых протоколов не превышает 1522, однако в EcoRouter существует возможность задать значение MTU в пределах от 82 до 9728. Таким образом становится возможным использование Jumbo frame (ethernet-кадр, в котором можно передать данные, по размеру превышающие 1500 байт).

Для административного выключения порта используется команда `shutdown` в контексте конфигурирования порта.

Для административного включения порта используется команда `no shutdown` в контексте конфигурирования порта.

При выполнении этих команд выводятся сообщения о состоянии линка.

Если порт выключен средствами системы, то в выводе статистики по портам его состояние обозначается "**administratively down**".

При выключении порта все привязанные к нему сущности (интерфейсы и сервисные интерфейсы) также выключаются.

Пример:

```
ecorouter#show port
Gigabit Ethernet [igb] port ge3 is up
  MTU: 9728
  LACP priority: 32767
  Input packets 12757610, bytes 4507446111, errors 0
  Output packets 41139047, bytes 47165314669, errors 0
  Service instance ge3.olia is up
  ingress encapsulation untagged
```

```

ingress rewrite none
egress encapsulation untagged
egress none
Connect bridge raccoon symmetric
Input packets 12757610, bytes 4507446111
Output packets 41139681, bytes 47165195683
Gigabit Ethernet [igb] port ge4 is down
MTU: 9728
LACP priority: 32767
Input packets 1468304, bytes 249589783, errors 0
Output packets 4598726, bytes 5586328327, errors 0
Service instance ge4.sergey is down
ingress encapsulation untagged
ingress rewrite none
egress encapsulation untagged
egress none
Connect bridge raccoon symmetric
Input packets 1468303, bytes 249590010
Output packets 4653951, bytes 5592867728
Gigabit Ethernet [igb] port ge5 is up
MTU: 9728
LACP priority: 32767
Input packets 6878595, bytes 3664083768, errors 0
Output packets 13210832, bytes 14688926470, errors 0
Service instance ge5.alexander is up
ingress encapsulation untagged
ingress rewrite none
egress encapsulation untagged
egress none
Connect bridge raccoon symmetric
Input packets 6878604, bytes 3664084308
Output packets 13212782, bytes 14688868859
Gigabit Ethernet [igb] port ge6 is down
MTU: 9728
LACP priority: 32767
Input packets 3103204, bytes 504476889, errors 0
Output packets 5093754, bytes 4810094601, errors 0
Service instance ge6.timurr is down
ingress encapsulation untagged
ingress rewrite none
egress encapsulation untagged
egress none
Connect bridge raccoon symmetric
Input packets 3103202, bytes 504475973
Output packets 5125510, bytes 4812650924
Gigabit Ethernet [igb] port ge7 is down
MTU: 9728
LACP priority: 32767
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
ecorouter(config)#port te0
ecorouter(config-port)#shutdown

```

```

ecorouter(config-port)#[Fri Sep  2 08:31:10 2016] [INFO] PHYS: LINK is
DOWN on port 'te0(0)'
ecorouter#show port
10 Gigabit Ethernet [none] port te0 is administratively down
  MTU: 9728
  LACP priority: 32767
    link state DOWN;
  Input packets 0, bytes 0, errors 0
  Output packets 0, bytes 0, errors 0
    Service instance te0.100 is down
    ingress encapsulation none
    ingress rewrite none
    egress encapsulation none
    egress none
  Input packets 0, bytes 0
  Output packets 0, bytes 0
  Service instance te0.200 is down
    ingress encapsulation dot1q any
    ingress rewrite none
    egress encapsulation dot1q any
    egress none
  Input packets 0, bytes 0
  Output packets 0, bytes 0
10 Gigabit Ethernet [none] port tel is up
  MTU: 9728
  LACP priority: 32767
    link state UP;
  Input packets 0, bytes 0, errors 0
  Output packets 0, bytes 0, errors 0
ecorouter(config-port)#[Fri Sep  2 08:34:28 2016] [INFO] PHYS: LINK is
UP on port 'te0(0)'
ecorouter#show port
10 Gigabit Ethernet [none] port te0 is up
  MTU: 9728
  LACP priority: 32767
    link state UP;
  Input packets 0, bytes 0, errors 0
  Output packets 0, bytes 0, errors 0
    Service instance te0.100 is up
    ingress encapsulation none
    ingress rewrite none
    egress encapsulation none
    egress none
  Input packets 0, bytes 0
  Output packets 0, bytes 0
  Service instance te0.200 is up
    ingress encapsulation dot1q any
    ingress rewrite none
    egress encapsulation dot1q any
    egress none
  Input packets 0, bytes 0
  Output packets 0, bytes 0

```

```
10 Gigabit Ethernet [none] port tel is up
MTU: 9728
LACP priority: 32767
link state UP;
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
```

4.2 Агрегирование каналов

Агрегирование каналов – объединение нескольких каналов в один логический канал для увеличения пропускной способности и резервирования. Чтобы добавить порты в объединенный канал они должны быть идентично настроены и параллельны. То есть, агрегируемые каналы должны соединять между собой два устройства, параллельно друг другу.

В один агрегированный порт могут быть объединены до 8 портов на одной или разных картах устройства. Для объединения скоростные характеристики портов должны совпадать. Также на портах не должно быть привязанных сервисных интерфейсов. Сервисный интерфейс для операций с метками VLAN настраивается на сконфигурированном агрегированном порту (см. раздел Сервисные интерфейсы).

4.3 Интерфейс

Интерфейс (*interface*) – это логический интерфейс для адресации L3. Название интерфейса задается администратором и чувствительно к регистру (например: `intQQ` и `intqq`, – это разные интерфейсы). В названиях интерфейсов разрешены только строчные и прописные латинские буквы, цифры и знак точки '!'.

В EcoRouterOS существуют L3 интерфейсы, которые служат для поддержки определенного функционала (IP Demux, интерфейсы обратной петли и т.д.) и называются соответственно. В качестве имени обычных логических интерфейсов для адресации L3 нельзя использовать названия специальных интерфейсов (ВСЕ ИМЕНА РЕГИСТРОЗАВИСИМЫЕ):

- **demux.<номер>**,
- **loopback.<номер>**,
- **pproe.<номер>**,
- **Null**,
- **vlan**.

Базовая настройка интерфейса:

```
ecorouter(config)#interface NAME
```

Создание интерфейса. Где NAME – произвольное имя.

```
ecorouter(config-if) #
```

Общий вид командной строки при конфигурировании интерфейса.

```
ecorouter(config-if)#ip address 10.10.10.1/24
```

Назначение IP-адреса с префиксом.

Или:

```
ecorouter(config-if) # ip address 10.10.10.1 255.255.255.0
```

Назначение IP-адреса с маской подсети.

```
ecorouter(config-if) #no shutdown
```

Включение интерфейса.

```
ecorouter(config-if) # shutdown
```

Выключение интерфейса.

4.4 Интерфейс loopback

Интерфейс loopback (Interface Loopback) – это виртуальный петлевой L3 интерфейс. Название интерфейса loopback задается администратором и чувствительно к регистру (например: Int loopback.QQ и Int loopback.qq, – это разные интерфейсы). Формат названия такого интерфейса: **loopback.<название>**.

В EcoRouterOS номера интерфейсов loopback должны быть уникальными среди всех созданных виртуальных маршрутизаторов. То есть имя **loopback.100** не может быть использовано в VR1 и VR2. При попытке использовать одно и то же имя в другом виртуальном устройстве EcoRouterOS выдаст сообщение об ошибке поясняющее, что интерфейс используется в другом устройстве.

Базовая настройка интерфейса loopback:

```
ecorouter(config) #interface loopback.NAME
```

Создание интерфейса loopback. Где NAME – произвольный номер.

```
ecorouter(config-if-loopback) #ip address 1.1.1.1/32
```

Назначение IP-адреса с префиксом.

Или:

```
ecorouter(config-if-loopback) #ip address 1.1.1.1 255.255.255.255
```

Назначение IP-адреса с маской подсети.

```
ecorouter(config-if-loopback) #no shutdown
```

Команда включения интерфейса.

```
ecorouter(config-if-loopback) #shutdown
```

Команда выключения интерфейса.

4.5 Интерфейс demux

Интерфейс IP demux – это виртуальный L3 интерфейс, на который может быть назначен IP-адрес из маршрутизируемой подсети. Пересылка пакетов в другие подсети будет осуществляться за счёт привязки к определенному порту с набором service instance.

Базовая настройка интерфейса IP demux:

Таблица 16

Команда	Описание
interface demux.<NAME>	Создание интерфейса demux. Где <NAME> – произвольное число

Команда	Описание
ip address <IP>/<MASK>	Назначение IP-адреса с префиксом

Пример:

```
ecorouter(config)#interface demux.0
ecorouter(config-if-demux)#ip address 10.10.10.1/24
```

4.6 Bridge domain

Bridge domain – это локальный широковещательный домен второго уровня модели OSI, который существует отдельно от понятия VLAN и оперирует идентификаторами виртуальных подсетей. Bridge domain создается на каждом устройстве отдельно и имеет значение только на нём. Подобное разделение позволяет определять различные виртуальные подсети на порт и гибко управлять отдельными виртуальными доменами. Тем самым снимается ограничение масштабируемости, обусловленное глобальной привязкой VLAN к конкретному устройству сегмента. Bridge domain строится из одного или нескольких L2 сервисных интерфейсов, называемых service-instance.

Команда создания bridge domain: **bridge <NAME>**. Где NAME – произвольное имя.

4.7 Интерфейс bridge domain

Интерфейс bridge domain (Bridge Domain Interface, BDI) – это логический интерфейс, позволяющий организовать двунаправленный поток трафика между сетями из bridge domain в L3 интерфейсы для маршрутизации.

Базовая настройка интерфейса:

Таблица 17

Команда	Описание
interface <NAME>	Создание интерфейса бридж домена. Где NAME – произвольное имя
ip address <IP><MASK>	Назначение IP-адреса с маской подсети
connect to bridge <NAME>	Привязка к созданному ранее bridge

Пример:

```
ecorouter(config)#interface NAME
ecorouter(config-if)#ip address 10.10.10.1 255.255.255.255
ecorouter(config-if)#connect to bridge NAME
```

4.8 Интерфейс PPPoE

PPPoE (Point-to-point protocol over Ethernet) - сетевой протокол канального уровня (второй уровень сетевой модели OSI) передачи кадров PPP через Ethernet. В основном используется xDSL-сервисами. Предоставляет дополнительные возможности (аутентификация, сжатие данных, шифрование).

Описание команд для настройки PPPoE-сервера на EcoRouter представлено в таблице ниже.

Таблица 18

Команда	Описание
pppoe-profile <PROFILE_NAME>	Команда вводится в конфигурационном режиме (config). В результате выполнения этой команды будет создан профиль, в котором можно будет изменять настройки протокола PPPoE, настройки для создания PPP-соединений, указать subscriber map и способ раздачи ip-адресов абонентам.
interface pppoe.<IF_NUMBER>	Команда вводится в конфигурационном режиме (config). В результате выполнения этой команды будет создан интерфейс, настройки которого будут использованы для создания PPPoE-сессий.
profile <PROFILE_NAME>	Команда вводится в контекстном режиме настройки pppoe-интерфейса (config-if-pppoe). В результате выполнения этой команды на интерфейсе будет включен протокол PPPoE с настройками, указанными в выбранном профиле.

4.9 Service Instance

Service instance (Субинтерфейс, SI, Сервисный интерфейс) является логическим субинтерфейсом, работающим между L2 и L3 уровнями. Данный вид интерфейса необходим для соединения физического порта с интерфейсами L3, интерфейсами bridge, портами. Используется для гибкого управления трафиком на основании наличия меток VLANов в фреймах, или их отсутствия. Сквозь сервисный интерфейс проходит весь трафик, приходящий на порт. На одном порту может существовать много сервисных интерфейсов, которые будут обрабатывать разные метки VLAN'ов по-разному.

Команда создания сервисного интерфейса: **service-instance <NAME>**.

Название субинтерфейса задается администратором. В каждой строчке service instance может содержаться только один признак трафика.

Пример:

```
ecorouter(config)#port te0
```

Сервисный интерфейс создаётся в режиме конфигурации порта.

```
ecorouter(config-port)#service-instance 100
```

Создание сервисных интерфейсов.

```
ecorouter(config-service-instance)#encapsulation dot1q 4
```

Указание номера, обрабатываемого VLAN.

```
ecorouter(config-service-instance)#rewrite pop 1
```

Указание выполняемой операции.

```
ecorouter(config-service-instance)#connect ip interface e1
```

Указание в какой интерфейс нужно отправить обработанные кадры.

4.10 Команды просмотра состояний интерфейсов

Просмотр состояния и текущей конфигурации портов, интерфейсов и субинтерфейсов осуществляется при помощи команды **show**. Ниже приведено несколько примеров.

Просмотр состояния и текущей конфигурации всех портов:

```
ecorouter#show port
```

```

te0 is up
Type: [10 Gigabit Ethernet]
MTU: 9728[82-9728]
link state UP;
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
tel1 is up
Type: [10 Gigabit Ethernet]
MTU: 9728[82-9728]
link state UP;
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
Service instance tel1/QQ1 is up

```

Просмотр состояния и конфигурации определенного порта:

```

ecorouter#show port te0
te0 is up
Type: [10 Gigabit Ethernet]
MTU: 9728[82-9728]
link state UP;
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0

```

Просмотр состояния интерфейса port channel:

```
ecorouter#show port channel
```

Подробный вывод состояния всех созданных интерфейсов:

```

ecorouter#show interface
Interface e56[11] is up, line protocol is up
Ethernet address 0000.ab80.d303
MTU: 1500 [68-65536]
NAT: no
ICMP redirection is on
Label switching is disabled
<UP,BROADCAST,RUNNING,MULTICAST>
inet 10.10.10.1/24 broadcast 10.10.10.255/24
    Input packets 0, bytes 0
    Output packets 0, bytes 0
Interface e3[10] is up, line protocol is up
Ethernet address 0000.ab80.d303
MTU: 1500 [68-65536]
NAT: no
ICMP redirection is on
Label switching is disabled
<UP,BROADCAST,RUNNING,MULTICAST>
DHCP Proxy is enabled
128.66.1.1
    Input packets 0, bytes 0
    Output packets 0, bytes 0

```

Подробный вывод состояния и конфигурации определенного интерфейса:

```

ecorouter#show interface e3
Interface e3[10] is up, line protocol is up
Ethernet address 0000.ab80.d303
MTU: 1500 [68-65536]

```

```
NAT: inside
ICMP redirection is on
Label switching is disabled
<UP,BROADCAST,RUNNING,MULTICAST>
DHCP Proxy is enabled
128.66.1.1
Input packets 0, bytes 0
Output packets 0, bytes 0
```

Краткий вывод статусов всех интерфейсов:

```
ecorouter#show interface brief
Interface      Status      Protocol      Description
-----
e56          up          up
e3           up          up          Users
```

Просмотр информации о сессиях через интерфейс ip demux. Где указаны логический и физический адреса хоста, номер порта маршрутизатора за которым он включен и номер VLAN.

```
ecorouter#show ip-unnumbered-table e10
IP Address      MAC Address      Port      C-tag      S-tag
-----
10.10.10.2     0050.7966.6800    <1>        2          -----
```

Все интерфейсы и порты по умолчанию включены. Для того, чтобы выключить интерфейс или порт нужно дать команду **shutdown** в режиме конфигурации интерфейса или порта.

```
ecorouter#configure terminal
ecorouter(config)#port te0
ecorouter(config-port)#shutdown
ecorouter(config-port)#
ecorouter#show port te0
te0 is administratively down
```

Строчка «administratively down» указывает на то, что данный порт сейчас выключен.

```
Type: [10 Gigabit Ethernet]
MTU: 9728 [82-9728]
link state DOWN;
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
```

4.11 Команды просмотра SFP модулей

Для просмотра краткой информации о SFP-модулях используется команда административного режима **show transceiver**.

Команда **show transceiver** показывает информацию по всем портам, а ее модификация **show transceiver port <NAME>** показывает информацию по конкретному порту.

Для данной команды возможно использование модификаторов и вывода в файл так же, как и для других команд **show**.

Для SFP-модулей выводится информация, представленная в таблице ниже.

Таблица 19

Параметр	Описание
Module Type	Тип передатчика: 1000BASE-T - модуль стандарта 1000BASE-T - 1 Гбит/с, витая пара, длина сегмента до 100 метров; 100BASE-FX - модуль стандарта 100BASE-FX - 100 Мбит/с, максимальная длина сегмента 412 метров для полнодуплексного режима и 2 километра для полудуплексного режима по мультимодовому волокну; 1000BASE-SX - модуль стандарта 1000BASE-SX - 1 Гбит/с, мультимодовое оптоволокно с длиной сегмента 220/550 метров; 1000BASE-LX - модуль стандарта 1000BASE-LX - 1 Гбит/с, максимальная длина сегмента 550 метров для мультимодового оптоволокна и 5 километров для одномодового режима; 100BASE-LX - модуль стандарта 100BASE-LX - 100 Мбит/с, максимальная длина сегмента 15 километров в полнодуплексном режиме по паре одномодовых оптических волокон; 10GBASE-SR - модуль стандарта 10GBASE-SR - 10 Гбит/с, максимальная длина сегмента 300 метров для мультимодового оптоволокна; 10GBASE-LR - модуль стандарта 10GBASE-LR - 10 Гбит/с, максимальная длина сегмента 10 километров для одномодовых оптических волокон; 10GBASE-LRM - модуль стандарта 10GBASE-LRM - 10 Гбит/с, максимальная длина сегмента 220 метров для мультимодового оптоволокна; 10GBASE-ER - модуль стандарта 10GBASE-ER - 10 Гбит/с, максимальная длина сегмента 40 километров для одномодовых оптических волокон; Unspecified - неизвестный тип модуля.
Module Vendor Name	Производитель
Module Part Number	Артикул
Module Serial Number	Серийный номер
Module Revision	Версия
Module Manufacturing Date	Дата изготовления. Формат: ГГММДД
Module supports DDM	Есть ли поддержка функции цифрового контроля параметров модуля (температуры, напряжения и т.д.)
Module temperature	Температура модуля в градусах по Цельсию. Параметр доступен при поддержке DDM
Module voltage	Напряжение на модуле, Вольт. Параметр доступен при поддержке DDM

Для "медных" портов данная информация недоступна, вместо нее выводится строка "Module doesn't identify itself as SFF-compatible".

Пример вывода информации:

```
ecorouter#show transceiver
Port: te0
Module doesn't identify itself as SFF-compatible
Port: te1
Module doesn't identify itself as SFF-compatible
Port: te2
Module doesn't identify itself as SFF-compatible
Port: ge0
Module Type: 1000BASE-T
```

```
Module Vendor Name: FiberTrade
Module Part Number: SFP-Copp-10-1000
Module Serial Number: FT1601190702
Module Revision: A
Module Manufacturing Date: 160119
Module supports DDM: no
Port: fe0
  Module Type: 100BASE-FX
  Module Vendor Name: FiberTrade
  Module Part Number: FT-SFP-GE-100FX
  Module Serial Number: FGF32M03
  Module Revision: 1.0
  Module Manufacturing Date: 160527
  Module supports DDM: no
Port: gel
  Module Type: 1000BASE-T
  Module Vendor Name: OptiCin
  Module Part Number: SFP-RJ45
  Module Serial Number: TA2C010008
  Module Revision: A
  Module Manufacturing Date: 100305
  Module supports DDM: no
Port: fe1
  Module Type: Unspecified
  Module Vendor Name: OEM
  Module Part Number: PPH-1302-02CD
  Module Serial Number: P0816060804
  Module Revision: 1.0
  Module Manufacturing Date: 160613
  Module supports DDM: yes
  Module temperature: 24.00 C
  Module voltage: 3.28 Volt
  Port: te3
  Module Type: 10GBASE-LR
  Module Vendor Name: OEM
  Module Part Number: SFP+-10G-LR
  Module Serial Number: P1309040335
  Module Revision: A
  Module Manufacturing Date: 130904
  Module supports DDM: yes
  Module temperature: 37.00 C
  Module voltage: 3.27 Volt
  Module TX power: 0.63 mW (-2.01 dBm)
```

5 Сервисные интерфейсы

При входе на порт кадр с меткой VLAN будет помещен в сервисный интерфейс, выделенный для обработки данной метки VLAN. После сервисный интерфейс может метку VLAN заменить, добавить или снять и передать в другой порт или интерфейс. То есть сервисный интерфейс связывает порт и порт или порт и интерфейс (порт и bridge domain) в пределах устройства.

5.1 Виды инкапсуляции

5.1.1 Виды инкапсуляции

Кадр помещается в тот или иной сервисный интерфейс на порту по признаку инкапсулированного в него тега dot1q или по его отсутствию. На одном порту может быть несколько сервисных интерфейсов. На маршрутизаторе может существовать до 4000 сервисных интерфейсов.

5.1.2 Команды настройки инкапсуляции

Настройка инкапсуляции осуществляется в контекстном режиме конфигурирования сервисного интерфейса. Который, в свою очередь, доступен в контексте конфигурирования порта.

То есть для того, чтобы приступить к настройкам инкапсуляции необходимо ввести, например, следующую последовательность команд:

```
ecorouter#configure terminal
ecorouter(config)#port te0
ecorouter(config-port)#service-instance 100
```

Инкапсуляция настраивается на сервисном интерфейсе при помощи команды **encapsulation**. В таблице ниже приведено описание параметров данной команды.

Таблица 20

Вид инкапсуляции	Описание
encapsulation untagged	Нетегированные кадры
encapsulation default	Указание, что данным сервисным интерфейсом будут обрабатываться все остальные метки, не указанные до этого в других сервисных интерфейсах на порту. Применяется в L3 бриджах и в соединениях без участия L3 маршрутизации
encapsulation dot1q any	Инкапсуляция IEEE 802.1q с любым тегом в кадре
encapsulation dot1q <TAG>	Инкапсуляция IEEE 802.1q с конкретным тегом в кадре
encapsulation dot1q <TAG> second-dot1q <TAG>	Инкапсуляция IEEE 802.1q с 2-мя тегами, содержащимися в кадре. Значения тегов указываются по порядку начиная с внутреннего
encapsulation dot1q <TAG1>-<TAG2>	Инкапсуляция IEEE 802.1q с диапазоном тегов
encapsulation dot1q <TAG> exact	Аргумент exact указывает на то, что данный сервисный интерфейс будет обрабатывать кадр только с одной указанной меткой, или одной меткой из диапазона

Аргумент **exact** является обязательным в случае дальнейшей передачи кадра на L3 уровень (за исключением Demux интерфейса). В случае передачи кадра в bridge или порт, аргумент можно не указывать.

5.2 Операции над метками

После того, как кадр был помещен в определенный сервисный интерфейс над меткой может выполняться операция замены, удаления или добавления значения. Для этого выполняется команда **rewrite** с различными аргументами.

Если кадр после прохождения сервисного интерфейса будет передаваться на интерфейс для последующей обработки на L3 (исключение интерфейс BDI, интерфейс IP-demux), над ним должна быть выполнена команда с аргументом **pop**. Операция **pop** удаляет метку из кадра.

Если кадр после прохождения через сервисный интерфейс будет передан в порт или bridge, то тут могут быть выполнены все возможные операции над метками.

5.2.1 Команды операций над метками

Таблица 21

Вид операции над меткой	Описание
Rewrite pop VALUE	Операция снятия метки. VALUE равен 1 или 2
Rewrite push VALUE VALUE	Добавление метки. VALUE значение метки. Верхняя метка - первая
Rewrite translate 1-to-1 VALUE	Замена одной метки на другую. Где VALUE значение новой метки
Rewrite translate 1-to-2 VALUE VALUE	Замена одной метки на две других
Rewrite translate 2-to-2 VALUE VALUE	Замена двух меток на две других
Rewrite translate 2-to-1 VALUE	Замена двух меток на одну

5.2.2 Направление движения трафика через сервисный интерфейс

Операции над меткой в кадре осуществляются при движении в обоих направлениях через сервисный интерфейс. Например, при прохождении кадра от порта к присоединенному интерфейсу и от интерфейса к порту. Правила обработки метки в обратном направлении создаются автоматически.

Разновидность работы сервисного интерфейса, работающего в две стороны симметрично, называется **ambiguous**. Если в сервисном интерфейсе задана операция **pop** при движении кадра от порта к интерфейсу, то при движении пакета от интерфейса к порту будет выполняться **push**. Создание такого сервисного интерфейса возможно при явном указании нужной метки.

Пример:

```
encapsulation dot1q 3 exact
rewrite pop 1
```

В данном примере при движении в одну сторону метка 3 будет сниматься, при движении в обратную сторону – добавляться.

Разновидность работы сервисного интерфейса, работающего несимметрично в две стороны, называется **unambiguous**. Такой сервисный интерфейс создаётся при общем правиле обработки диапазона меток.

Пример:

```
encapsulation dot1q 1-3 exact
```

При движении трафика в одну сторону единственная метка, попадающая в указанный диапазон, будет сниматься, при движении в обратную сторону кадр будет передаваться без метки, так как не очевидно, какую метку из диапазона в него необходимо поместить. Эта особенность накладывает ограничения на использование такой разновидности сервисных интерфейсов в некоторых сценариях.

5.2.3 Операции над метками в сервисных интерфейсах

Есть три варианта операций над метками: удаление существующей метки/меток, добавление новой метки/меток и трансляция метки/меток из одного значения в другое.

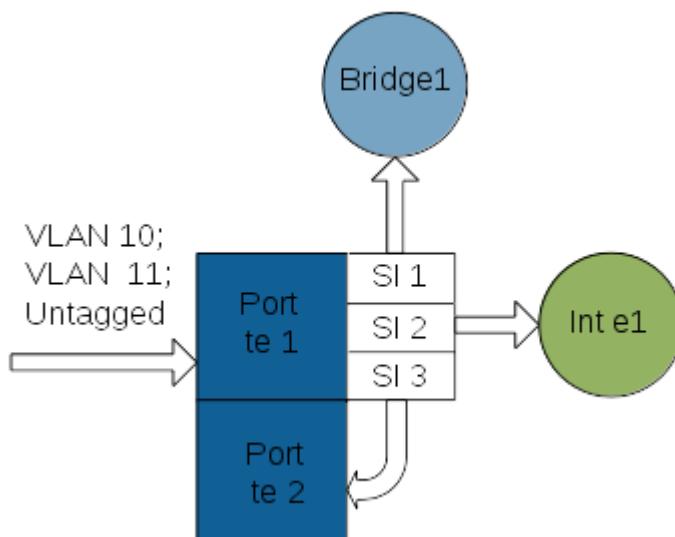


Рисунок 6

Рассмотрим возможные варианты действий над метками в случае, представленном на рисунке. На порт te1 устройства приходят 10, 11 VLAN и нетегированный трафик.

Трансляция меток

Трафик, принадлежащий 10 VLAN, нужно перенаправить в порт te2 с меткой 5 VLAN.

На порту, куда приходит VLAN 10, создаем сервисный интерфейс для операции с этими метками.

```
ecorouter(config)#port te1
ecorouter(config-port)#service-instance 3
```

Из всего объема трафика выбираем трафик с меткой VLAN 10. Аргумент **exact** указывает, что этот сервисный интерфейс обрабатывает кадры с единственной 10 меткой.

```
ecorouter(config-service-instance)#encapsulation dot1q 10 exact
```

Меняем метку 10 на метку VLAN 5. Трансляция из 1 в 1.

```
ecorouter(config-service-instance)#rewrite translate 1-to-1 5
```

Указываем, куда отправлять трафик после операции над меткой.

```
ecorouter(config-service-instance) #connect port te2
```

Service-instance 3 является симметричным. Когда трафик пойдёт в обратном направлении, то service-instance будет иметь такую конфигурацию.

```
encapsulation dot1q 5 exact
rewrite translate 1-to-1 10
```

И, таким образом, в порт te1 будет отдавать трафик с меткой VLAN 10.

Все возможности трансляции меток VLAN

Трансляция одной метки в две метки.

Данная команда заменяет одну метку двумя другими. Операция выполняется только в случае единственной входящей метки.

rewrite translate 1-to-2 <МЕТКА1> <МЕТКА2>

Пример настройки:

```
ecorouter(config)#port te1
ecorouter(config-port)#service-instance 31
ecorouter(config-service-instance)#encapsulation dot1q 10 exact
ecorouter(config-service-instance)#rewrite translate 1-to-2 5 15
```

Заменили одну метку 10, на метки 5 и 15. Метка 5 будет первой по порядку в кадре.

Трансляция двух меток в две другие:

rewrite translate 2-to-2 <МЕТКА1> <МЕТКА2>

Пример настройки:

```
ecorouter(config)#port te1
ecorouter(config-port)#service-instance 31
ecorouter(config-service-instance)#encapsulation dot1q 20 second-dot1q
40
ecorouter(config-service-instance)#rewrite translate 2-to-2 5 15
```

Заменили метки 20 и 40 на метки 5 и 15. Метка 5 будет первой по порядку в кадре.

Трансляция двух меток в одну:

rewrite translate 2-to-1 <МЕТКА>

Пример настройки:

```
ecorouter(config)#port te1
ecorouter(config-port)#service-instance 31
ecorouter(config-service-instance)#encapsulation dot1q 20 second-dot1q
40
ecorouter(config-service-instance)#rewrite translate 2-to-1 5
```

2 пришедшие в порт метки будут заменены на одну.

Добавление меток

Весь нетегированный трафик обработаем с помощью команды **rewrite** с аргументом **push** в service-instance 1.

```
ecorouter(config)#port te1
ecorouter(config-port)#service-instance 2
```

Указываем, что весь нетегированный трафик будет обрабатываться этим сервисным интерфейсом.

```
ecorouter(config-service-instance) #encapsulation untagged
```

Указываем, что в каждый кадр помещаем метку 5.

```
ecorouter(config-service-instance) #rewrite push 5
```

Указываем, куда отправлять трафик после операции над меткой.

```
ecorouter(config-service-instance) #connect bridge 1
```

Bridge 1 должен быть предварительно создан.

На выходе из данного сервисного интерфейса весь трафик будет помечен меткой 5 VLAN.

При обратном движении из bridge 1 в порт te1 весь трафик будет уходить в порт без какой-либо метки.

Операции **translate** и **push** возможны только в случае привязки service instance к уровню L2, то есть к порту или bridge.

На третий уровень пакеты должны приходить без признака VLAN.

Метки VLAN снимаются с помощью команды **rewrite pop**.

Снятие меток

В service-instance 2 будем обрабатывать VLAN 11 на порту te1. Создаем service instance с именем 2.

```
ecorouter(config)#port te1  
ecorouter(config-port)#service-instance 2
```

Фильтруем 11 VLAN.

```
ecorouter(config-service-instance) #encapsulation dot1q 11 exact
```

Снимаем метку VLAN, чтобы передать кадр на L3 интерфейс. В данном случае команда **rewrite** с аргументом **pop 1**, указывает, что в кадре содержится только одна метка, и она будет удалена.

```
ecorouter(config-service-instance) #rewrite pop 1
```

Устанавливаем связку порта и интерфейса L3.

```
ecorouter(config-service-instance) #connect ip interface e1
```

Таким образом трафик попадает на интерфейс e1 без признака VLAN.

Для обратного направления будет верно следующее:

```
encapsulation untagged  
rewrite push 1
```

Добавляем метку 11 VLAN.

В service instance существует ещё один тип инкапсуляции: **encapsulation default**. Под такой тип инкапсуляции попадёт абсолютно весь трафик, не выделенный в отдельный service instance. Так как конкретно не указывается, какое количество меток содержится в кадре, и что это за метки, маршрутизатор не может проделать над ними никаких операций (снять, сменить итд.). Поэтому перенаправить кадры возможно тоже только в L2: bridge или порт.

Настройка service instance для маршрутизации 2 VLAN'ов

Имеется следующая схема сети.

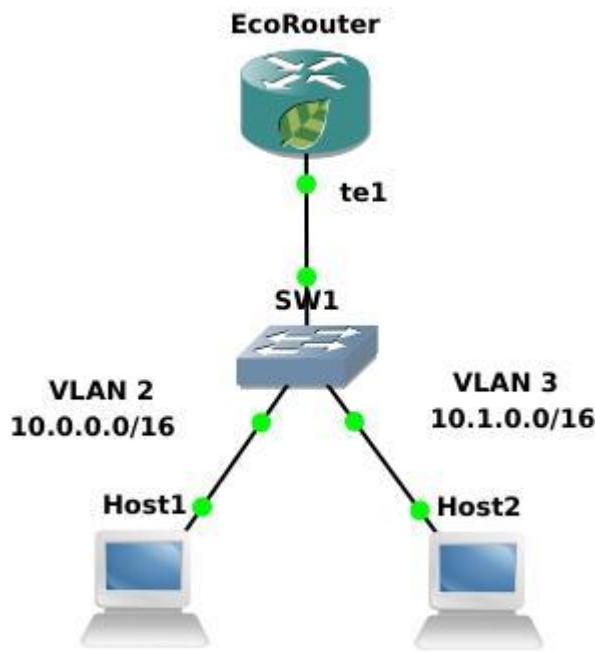


Рисунок 7

Шаг 1. Создаем интерфейсы и присваиваем IP-адреса.

```

ecorouter(config)#interface QQ1
ecorouter(config-if)#ip address 10.0.0.1/16
ecorouter(config)#interface QQ2
ecorouter(config-if)#ip address 10.1.0.1/16
  
```

Шаг 2. Создаем service-instance на порту для 2-го VLAN.

```

ecorouter(config)#port te1
ecorouter(config-port)#service-instance te1/QQ1
  
```

Шаг 3. Объявляем инкапсуляцию. Эта запись говорит, что мы ждём метку VLAN 2. Опция exact показывает, что под это правило попадут кадры только с меткой равной 2.

```

ecorouter(config-service-instance)#encapsulation dot1q 2 exact
  
```

Шаг 4. Снимаем метку опцией pop. Ключ 1 показывает, что снимаем только одну, верхнюю метку. На L3 кадр должен поступать без признаков VLAN.

```

ecorouter(config-service-instance)#rewrite pop 1
  
```

Шаг 5. Привязываем созданный сервисный интерфейс к L3 интерфейсу.

```

ecorouter(config-service-instance)#connect ip interface QQ1
  
```

Шаг 6. Симметричная настройка для 3-го VLAN.

```

ecorouter(config)#port te1
ecorouter(config-port)#service-instance te1/QQ2
  
```

Шаг 7. Объявляем инкапсуляцию. Эта запись говорит, что мы ждём метку VLAN 3. Опция exact показывает, что под это правило попадут кадры только с меткой равной 3.

```

ecorouter(config-service-instance)#encapsulation dot1q 3 exact
  
```

Шаг 8. Снимаем метку опцией pop. Ключ 1 показывает, что снимаем только одну метку, верхнюю. На L3 кадр должен поступать без признаков VLAN.

```

ecorouter(config-service-instance)#rewrite pop 1
  
```

Шаг 9. Привязываем созданный сервисный интерфейс к L3 интерфейсу.

```
ecorouter(config-service-instance) #connect ip interface QQ2
```

В случае движения кадра из сегмента сети вверх по схеме к маршрутизатору, на порту te1 выполняется действие снятия метки (см. Шаг 4). В случае движения пакета по схеме вниз от маршрутизатора к сегменту, будет происходить действие обратное этому, а именно **rewrite push 1**. Это возможно, так как номер VLAN в service-instance указан явно.

Настройка сервисного интерфейса для функционирования EcoRouter в роли L2 устройства

Имеется следующая схема сети.

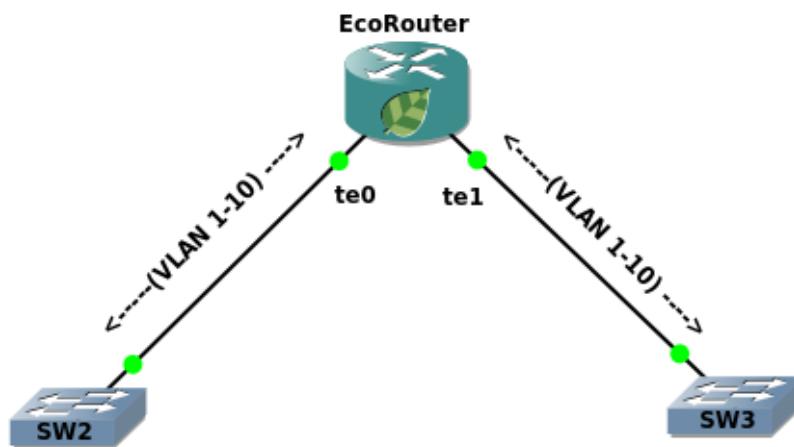


Рисунок 8

Шаг 1. Создаем service-instance на порту te0 для диапазона VLAN 1-10.

```
ecorouter(config)#port te0
ecorouter(config-port)#service-instance for_vlan(1-10)
ecorouter(config-service-instance)#encapsulation dot1q 1-10
```

Шаг 2. Привязываем сервисный интерфейс к выходному порту.

```
ecorouter(config-service-instance) #connect port te1
```

Шаг 3. Создаем service-instance на порту te1 для диапазона VLAN 1-10.

```
ecorouter(config)#port te1
ecorouter(config-port)#service-instance for_vlan(1-10)
ecorouter(config-service-instance)#encapsulation dot1q 1-10
```

Шаг 4. Привязываем сервисный интерфейс к выходному порту.

```
ecorouter(config-service-instance) #connect port te0
```

При подобной настройке EcoRouter выполняет коммутацию фреймов с тегами от 1 до 10 с порта te0 на порт te1 и наоборот. Порты коммутаторов в сторону маршрутизатора сконфигурированы как транковые и используют инкапсуляцию dot1q. Как видно, в двух разных сервисных интерфейсах `for_vlan(1-10)` инкапсуляция указана без ключевого слова `exact`, что позволительно лишь в случае отсутствия операций над метками (`pop`, `push`, `translate`) и подключения этих сервисных интерфейсов к порту или L2-домену (`bridge-domain`). Стоит заметить, что операция над тегами все еще возможна при конфигурировании L3 интерфейса (BDI). Возникающие ограничения сразу станут понятными, если представить ситуацию, когда

маршрутизатору на выходе кадра из порта необходимо добавить тег из некоторого диапазона локально сконфигурированных тегов (в примере, при указании в сервисном интерфейсе опции **rewrite pop 1**, на выходе из порта должна была бы применяться обратная операция добавления тегов от 1 до 10, что явно вносит неоднозначность, поскольку неизвестно, какой тег навешивать, EcoRouter исключает подобные ситуации и предупредит администратора о некорректно сконфигурированных фильтрах). Подобная гибкость управления трафиком в EcoRouter требует внимательности и четкого понимания происходящих операций над пакетами на интерфейсах и портах маршрутизатора. В CLI есть несколько команд группы **show** для просмотра сконфигурированных фильтров.

5.3 Просмотр настроек сервисных интерфейсов

5.3.1 Просмотр всех сервисных интерфейсов на всех портах

Для просмотра настроек сервисных интерфейсов, имеющихся на всех портах, используется команда **show port** или ее сокращенная форма: **sh port**.

Ingress – описание порядка обработки кадра при движении через порт в одном направлении. Как описано в сервисном интерфейсе администратором.

Egress – описание порядка обработки кадра при движении через порт в обратном направлении. Автоматически созданное ответное правило.

```
ecorouter#sh port
te0 is up
Type: [10 Gigabit Ethernet]
MTU: 9728[82-9728]
link state UP;
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
te1 is up
Type: [10 Gigabit Ethernet]
MTU: 9728[82-9728]
link state UP;
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
Service instance 1 is up
ingress encapsulation dot1q 12 exact
ingress rewrite pop 1
egress encapsulation untagged
egress push 12
Input packets 0, bytes 0
Output packets 0, bytes 0
te2 is up
Type: [10 Gigabit Ethernet]
MTU: 9728[82-9728]
link state UP;
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
te3 is up
Type: [10 Gigabit Ethernet]
MTU: 9728[82-9728]
```

```
link state UP;
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
te4 is up
Type: [10 Gigabit Ethernet]
MTU: 9728[82-9728]
link state UP;
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
ecorouter#
```

5.3.2 Просмотр сервисных интерфейсов на отдельном порту

Для просмотра настроек сервисных интерфейсов, имеющихся на конкретном порту, используется команда **show port <NAME>** или ее сокращенная форма: **sh port <NAME>**.

```
ecorouter#sh port tel
tel is up
Type: [10 Gigabit Ethernet]
MTU: 9728[82-9728]
link state UP;
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
Service instance 1 is up
ingress encapsulation dot1q 12 exact
ingress rewrite pop 1
egress encapsulation untagged
egress push 12
Input packets 0, bytes 0
Output packets 0, bytes 0
ecorouter#
```

5.3.3 Просмотр сервисных интерфейсов по номеру

Для просмотра настроек конкретного сервисного интерфейса, используется команда **show port <NAME> service-instance <SI_NAME>** или ее сокращенная форма: **sh port <NAME> service-instance <SI_NAME>**.

```
ecorouter#sh port tel service-instance 1
Service instance 1 is up
ingress encapsulation dot1q 12 exact
ingress rewrite pop 1
egress encapsulation untagged
egress push 12
Input packets 0, bytes 0
Output packets 0, bytes 0
```

6 Агрегирование каналов

Агрегирование каналов – объединение нескольких каналов в один логический канал для увеличения пропускной способности и резервирования. Чтобы добавить порты в объединенный канал они должны быть идентично настроены и параллельны. То есть, агрегируемые каналы должны соединять между собой два устройства, параллельно друг другу.

В один агрегированный порт могут быть объединены до 8 портов на одной или разных картах устройства. Для объединения скоростные характеристики портов должны совпадать. Также на портах не должно быть привязанных сервисных интерфейсов. Сервисный интерфейс для операций с метками VLAN настраивается на сконфигурированном агрегированном порту (см. раздел Сервисные интерфейсы).

6.1 Вычисление хэш-функции

Балансировка трафика осуществляется по потокам. Распределение кадров по каналам агрегированного порта происходит на основании данных в заголовках в кадре. На основании этой информации маршрутизатор принимает решение о использовании одного из физических каналов агрегированного порта. Для этого используется алгоритм хэширования.

Поля, используемые для вычисления хэш-функции по умолчанию:

Таблица 22

Router ID 4 Байта	S\C-Src Mac Последние 4 байта	S\C-Dst Mac Последние 4 байта	S\C-Src IP 4 Байта	S\C-Dst IP 4 Байта	Hash seed 1 Байт	Protocol Type 1 Байт	Port.no 1 Байт
----------------------	----------------------------------	----------------------------------	-----------------------	-----------------------	---------------------	-------------------------	-------------------

Router ID - неизменяемый идентификатор маршрутизатора.

S\C-Src Mac (Service\Client-Source Mac address) - MAC-адрес отправителя.

S\C-Dst Mac (Service\Client-Destination Mac address) - MAC-адрес получателя.

S\C-Src IP (Service\Client-Source IP) - IP-адрес отправителя.

S\C-Dst IP (Service\Client-Destination) - IP-адрес получателя.

Hash seed - изменяющее значение, уникальное в пределах маршрутизатора. Может принимать значения от 0 до 255.

Protocol Type - протокол транспортного уровня.

Port.no - номер порта, принял пакет.

Для пакетов с одинаковыми исходными данными результат вычисления хэш-функции всегда будет одинаков. Таким образом пакеты одного потока будут попадать в один порт (в один физический канал).

Результатом вычисления хэш-функции является 32-битное число. Первые его 16 бит используются для балансировки в Link Aggregation Control Protocol (LACP), остальные - для балансировки в Equal-cost multi-path routing (ECMP).

6.2 LACP

LACP (Link Aggregation Control Protocol) - сигнальный протокол для обеспечения работы агрегированного порта. Для определения принадлежности портов к одному логическому каналу LACP отсылает во все порты, где он включен, PDU сообщения. LACP может работать в пассивном и активном режимах. Устройство, на котором настроен LACP в пассивном режиме, не отсылает PDU (Protocol Data Unit) самостоятельно при настроенном агрегированном канале, а ждет получения PDU от соседнего устройства и только в случае получения отсылает свои. В активном режиме LACP постоянно шлет PDU пакеты.

В PDU содержатся собственные и ожидаемые от соседа параметры. Параметры содержат идентификатор системы, идентификатор группы интерфейсов, идентификатор физического интерфейса, с которого PDU был отправлен, и его текущее состояние. Агрегированный порт из состояния слушания переводится в состояние передачи трафика в случае одновременного выполнения следующих условий:

- битовое слово **state** идентифицирует порт соседнего устройства как присоединенный и работающий в группе,
- пришедшие от соседа параметры соответствуют ожидаемым,
- параметры, ожидаемые соседом, соответствуют собственным параметрам порта.

6.2.1 Настройка параметров

Для управления параметрами PDU используются команды контекстного режима конфигурирования агрегированного порта **ecorouter(config-port-channel)#**, представленные в таблице ниже.

Таблица 23

Команда	Описание
lacp enable	Включает функционал LACP на агрегированном порту. По умолчанию функционал выключен
lacp key <NUM>	Значение по умолчанию равно порядковому номеру порта в агрегированном канале. Изменяется в пределах от 0 до 65535
lacp mode (active passive)	Режим работы LACP
lacp period (fast slow)	Период отправки PDU сообщений и время их действия: Fast - сообщение раз в 1 секунду, 3 секунды таймаут (по умолчанию). Slow - сообщение раз в 30 секунд, 90 секунд таймаут.
lacp system-id <ID>	Идентификатор системе в формате XXXX:XXXX:XXXX
lacp system-priority <NUM>	Задает приоритет системы для разрешения конфликтов в выборе агрегированных портов. Чем меньше значение, тем выше приоритет. Значение по умолчанию равно 32768, изменяется в пределах от 0 до 65535

Параметр **port priority** задает приоритет порта в агрегированном канале. Чем меньше значение, тем выше приоритет. По умолчанию равно 32768. Для изменения значения в контекстном режиме конфигурирования порта необходимо вызвать команду **lacp-priority <NUM>**, где **NUM** - приоритет порта, изменяемый в пределах от 0 до 65535.

6.2.2 Команды просмотра

Для просмотра статистики по LACP и состояния агрегированных портов используются следующие команды типа **show**.

Для просмотра счетчиков используется команда **show counters lacp (| port)** с указанием конкретного агрегированного порта при необходимости.

Пример вывода команды:

```
ecorouter#show counters lacp
Port channel: ae.01
Port      LACPDU recv pkts    LACPDU sent pkts    Unknown recv
pkts    Illegal recv pkts
tel          0                1648                 0
                                         0
```

Для просмотра настроек LACP на портах EcoRouter используется команда **show lacp internal**.

```
ecorouter#sh lacp internal
Flags:  S - Device is requesting Slow LACPDUs
        F - Device is requesting Fast LACPDUs
        A - Device is in Active mode      P - Device is in Passive mode
Port channel: ae.1
              LACP port Admin Port  Port
Port      Flags State priority  Key  Number State
tel/0     SA    bndl   32767    0x10  8    0x3D
tel/1     SA    bndl   32767    0x10  9    0x3D
```

Для детального вывода настроек используется команда **show lacp internal detail**.

```
ecorouter#sh lacp internal detail
Flags:  S - Device is requesting Slow LACPDUs
        F - Device is requesting Fast LACPDUs
        A - Device is in Active mode      P - Device is in Passive mode
Port channel: ae.1
Actor (internal) information:
      Actor      Actor      Actor
Port      System ID      Port Number Age   Flags
tel/0     32767,000d.4838.8067 8      19   SA
      LACP Actor    Actor      Actor
      Port Priority  Oper Key    Port State
      32767       0x10      0x3D
      Port State Flags Decode:
      Activity: Timeout: Aggregation: Synchronization:
      Active    Long      Yes       Yes
      Collecting: Distributing: Defaulted: Expired:
      Yes       Yes      No       No
      Actor      Actor      Actor
Port      System ID      Port Number Age   Flags
tel/1     32767,000d.4838.8067 9      27   SA
      LACP Actor    Actor      Actor
      Port Priority  Oper Key    Port State
      32767       0x10      0x3D
      Port State Flags Decode:
      Activity: Timeout: Aggregation: Synchronization:
      Active    Long      Yes       Yes
```

Collecting:	Distributing:	Defaulted:	Expired:
Yes	Yes	No	No

Для просмотра информации о соседях используется команда **show lacp neighbour (| detail) (| port)**. Опционально можно указать отдельный порт и вывод детализированной информации.

Пример краткого и детализированного вывода команды:

```

ecorouter#sh lacp neighbor
Flags: S - Device is requesting Slow LACPDUs
      F - Device is requesting Fast LACPDUs
      A - Device is in Active mode      P - Device is in Passive mode
Port channel: ae.1
Partner's information:
      LACP port          Port   Port
Port    Flags priority Dev ID     Age   Number State
tel/0    FA    32768    908d.7845.9bc0 1    28    0x3F
tel/1    FA    32768    908d.7845.9bc0 9    27    0x3F
ecorouter#sh lacp neighbor detail
Flags: S - Device is requesting Slow LACPDUs
      F - Device is requesting Fast LACPDUs
      A - Device is in Active mode      P - Device is in Passive mode
Port channel: ae.1
Partner's information:
      Partner        Partner        Partner
Port    System ID       Port Number Age   Flags
tel/0    32768,908d.7845.9bc0 28      18    FA
      LACP Partner    Partner        Partner
      Port Priority  Oper Key     Port State
      32768         0x1        0x3F
      Port State Flags Decode:
      Activity: Timeout: Aggregation: Synchronization:
      Active     Short     Yes      Yes
      Collecting: Distributing: Defaulted: Expired:
      Yes       Yes      No       No
      Partner        Partner        Partner
Port    System ID       Port Number Age   Flags
tel/1    32768,908d.7845.9bc0 27      26    FA
      LACP Partner    Partner        Partner
      Port Priority  Oper Key     Port State
      32768         0x1        0x3F
      Port State Flags Decode:
      Activity: Timeout: Aggregation: Synchronization:
      Active     Short     Yes      Yes
      Collecting: Distributing: Defaulted: Expired:
      Yes       Yes      No       No

```

Для указанных команд могут использоваться модификаторы и вывод в файл, как и для любых других команд **show**.

6.3 ECMP

ECMP (Equal-cost multi-path routing) - механизм выбора лучшего пути до сети назначения среди равнозначных. Выбор выходного интерфейса и маршрута осуществляется на основании вычисления хэш-функции. Функционал включен по умолчанию.

6.4 Настройка Link aggregation

6.4.1 Именование агрегированных портов

Возможное количество агрегированных портов на устройстве равно $n/2$, где n - количество физических портов на устройстве. Имена агрегированных портов начинаются с букв **ae**, за которыми следует точка и порядковый номер.

6.4.2 Команды настройки агрегированного порта

Таблица 24

Команда	Описание
port ae.<номер>	Команда создания порта агрегированного канала, где ae - указание на вид порта, через точку указывается порядковый номер (в конфигурационном режиме)
bind <имя порта>	Добавление порта в агрегированный канал (в контекстном режиме конфигурирования агрегированного канала). При работе с ER-2008 необходимо учитывать ограничения (см. Оборудование)
description <строка>	Добавление описания порта агрегированного канала
mtu <значение>	Указание параметра mtu для агрегированного порта
add-mirror-session <значение>	Указание на созданное правило зеркалирования
service-instance <имя>	Создание сервисного интерфейса на агрегированном порту

Порт в уже существующий агрегированный канал также можно добавить в контекстном режиме конфигурирования порта при помощи команды **group <имя агрегированного порта>**.

6.4.3 Базовая настройка агрегированного порта. Способ 1

Агрегированный порт настраивается в режиме конфигурирования.

```
ecorouter(config)#port ae.10
```

где **ae** - обязательная часть в имени порта, а **10** - его идентификатор.

Добавление портов в агрегированный порт в контекстном режиме конфигурирования агрегированного канала:

```
ecorouter(config-port-channel)#bind te0
ecorouter(config-port-channel)#bind te1
ecorouter(config-port-channel)#bind te2
ecorouter(config-port-channel)#bind te3
```

Задание значения mtu на агрегированном порту:

```
ecorouter(config-port-channel)#mtu 1500
```

После создания агрегированного порта им можно управлять так же, как обычным портом.

6.4.4 Базовая настройка агрегированного порта. Способ 2

Агрегированный порт настраивается в режиме конфигурирования.

```
ecorouter(config)#port ae.100
```

где **ae** - обязательная часть в имени порта, а **100** - его идентификатор

Добавление порта в агрегированный канал в контекстном режиме конфигурирования порта:

```
ecorouter(config)#port te0
ecorouter(config-port)#group ae.100
ecorouter(config)#port te1
ecorouter(config-port)#group ae.100
ecorouter(config)#port te2
ecorouter(config-port)#group ae.100
```

По умолчанию значение mtu равно 9728. Задание значения mtu на агрегированном порту (значения на портах ae и te должны совпадать):

```
ecorouter(config-port-channel)#mtu 1500
```

После создания агрегированного порта им можно управлять так же, как обычным портом.

6.4.5 Команды просмотра состояния агрегированного порта

Просмотр состояния всех портов:

```
ecorouter#show port
Port te0 is up
Type: 10 Gigabit Ethernet
MTU: 9728 max 9728
link state UP;
Input packets 8391086176507358240, bytes 2322538359385584737, errors 0
Output packets 0, bytes 0, errors 0
Port te1 is up
Type: 10 Gigabit Ethernet
MTU: 9728 max 9728
link state UP;
Input packets 8391086176507358240, bytes 2322538359385584737, errors 0
Output packets 0, bytes 0, errors 0
Port te2 is up
Type: 10 Gigabit Ethernet
MTU: 9728 max 9728
link state UP;
Input packets 8391086176507358240, bytes 2322538359385584737, errors 0
Output packets 0, bytes 0, errors 0
Port te3 is up
Type: 10 Gigabit Ethernet
MTU: 9728 max 9728
link state UP;
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
```

```
Port te4 is up
Type: 10 Gigabit Ethernet
MTU: 9728 max 9728
link state UP;
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
Port ae.10 is up
Link te0
Link te1
Link te2
MTU: 9728
link state DOWN;
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
```

Просмотр состояния определенного порта:

```
ecorouter#sh port ae.10
Port ae.10 is up
Link te0
Link te1
Link te2
MTU: 9728
link state DOWN;
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
```

Просмотр счётчиков агрегированного порта:

```
ecorouter#sh counters port ae.100
Port ae.100
Received packets
Total received packets: 0
Total received bytes: 0
Transmitted packets
Total received bytes: 0
Total transmitted packets: 0
Total transmitted bytes: 0
Transmission errors
giants: 0
Total transmission errors: 0
```

7 Списки доступа

В EcoBNGOS используются различные списки доступа. Списки доступа представляют собой набор текстовых выражений-инструкций, которые позволяют "заглянуть" внутрь фрейма/пакета, сопоставить текстовое правило списка с данными в этом сообщении и на основании этого принять решение, что делать с этим фреймом/пакетом далее. В EcoBNGOS применяются следующие списки доступа (краткая характеристика ниже, более подробно о работе с каждым в соответствующих разделах настоящего руководства):

- Policy-filter-list;
- Filter-map;
- Prefix-list.

Policy-filter-list применяются при фильтрации маршрутных политик в различных протоколах юникастовой и мультикастовой маршрутизации, их рекламе, редистрибуции, добавлении специальных правил при работе с маршрутной информацией. Они НЕ МОГУТ применяться для блокировки или разрешения прохождения трафика через маршрутизатор.

Filter-мары применяются для блокировки или разрешения прохождения транзитного трафика через маршрутизатор. Они также применимы в сценариях QoS, PBR и HTTP-редиректа.

Prefix-list по функциональности аналогичны Policy-filter-list с той лишь разницей, что позволяют пользователю более гибко управлять масками подсетей. Эти списки широко применяются при конфигурировании BRAS.

7.1 Policy-filter-list

Policy-filter-list - функционал, позволяющий создавать списки правил для фильтрации, редистрибуции, суммаризации и управления маршрутными политиками в различных протоколах маршрутизации.

Сущность policy-filter-list представляет из себя вариант списка доступа, где можно указать лишь IP-адрес и инверсную маску.

Списки фильтров создаются в конфигурационном режиме. В одном списке фильтров может существовать несколько правил. Адрес сети, который передается в маршрутном обновлении, указывается с wildcard.

Синтаксис создания и добавления правил в policy-filter-list: **policy-filter-list <PFL_NAME> [deny | permit] <ADDRESS> <WILDCARD>**.

Для policy-filter-list можно задать описание командой: **policy-filter-list <PFL_NAME> remark <DESCRIPTION>**.

Параметры policy-filter-list описаны в таблице ниже.

Таблица 25

Параметр	Описание
PFL_NAME	Номер списка фильтрации. Нумерация списков осуществляется из диапазона от 1 до 99 и от 1300 до 1999
permit deny	Тип правила: разрешить (permit) или запретить (deny)

Параметр	Описание
ADDRESS	IP-адрес сети, задается в виде A.B.C.D . Если под правило должны попадать все адреса, значение параметра должно быть any
WILDCARD	Инверсная маска, задается в виде A.B.C.D

После создания списка фильтров, он должен быть применен к определенному процессу маршрутизации на устройстве.

Команды добавления фильтров различаются в зависимости от протокола.

Таблица 26

Команда	Описание
Distribute-list <номер>	Команда добавления списка фильтров в контекст маршрутизации OSPF
In	Указание применять список фильтров на вход
Out	Указание применять список фильтров на выход

7.1.1 Базовая конфигурация списка фильтров

```
ecorouter(config)#policy-filter-list 99 permit 172.168.1.0 0.0.0.255
```

где **99** – имя данного списка фильтров,

permit 172.168.1.0 0.0.0.255 – аргумент, указывающий, что маршрутное обновление о данной сети разрешено.

После создания списка фильтров он должен быть применен к определенному процессу маршрутизации на устройстве.

Команды добавления фильтров различаются в зависимости от протокола.

7.1.2 Настройка фильтрации маршрутной информации в BGP

Настройка списков фильтрации делается аналогично OSPF.

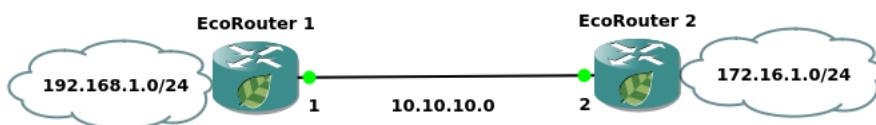


Рисунок 9

Применение списка фильтрации отличается.

Для фильтрации маршрутных обновлений BGP список фильтров применяется к определенному соседу с указанием направления.

Пример настройки

Создан список фильтров, который отфильтровывает все сети, начинающиеся на **192**.

```
policy-filter-list 99 permit 192.0.0.0 0.255.255.255
```

Сконфигурирован процесс маршрутизации BGP, объявлены сети и соседи.

```

router bgp 100
network 10.1.1.0/24
network 10.2.0.0/16
network 172.64.1.0/24
network 172.64.2.0/24

```

```
network 172.64.3.0/24
network 192.1.1.0/24
network 192.1.2.0/24
network 192.2.3.0/24
network 192.128.1.0/30
network 192.129.1.0/30
neighbor 10.0.0.13
remote-as 200
```

Список фильтров применяется к соседу с указанием номера списка и направления фильтрации.

```
neighbor 10.0.0.13 distribute-list 99 out
```

Таким образом, сосед 10.0.0.13 получит в маршрутных обновлениях только следующие сети:

```
network 192.1.1.0/24  
network 192.1.2.0/24  
network 192.2.3.0/24  
network 192.128.1.0/30  
network 192.129.1.0/30
```

7.1.3 Настройка фильтрации маршрутной информации в IS-IS

Между маршрутизаторами 1, 2 и 3 настроена динамическая маршрутизация с помощью протокола IS-IS.

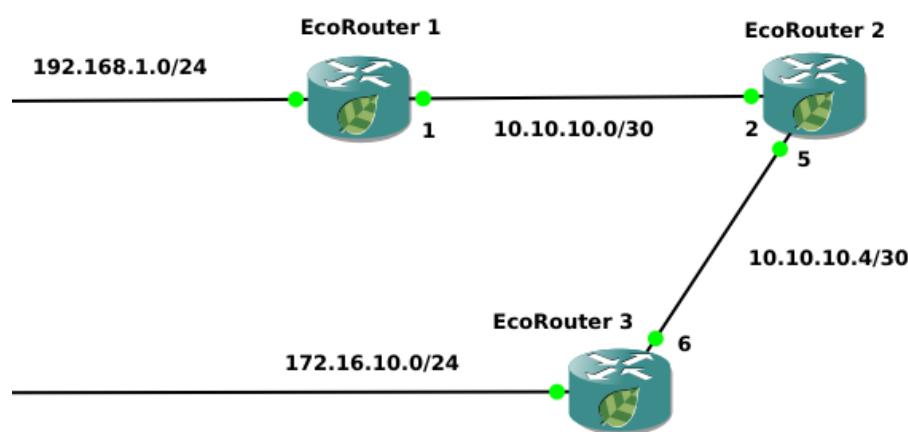


Рисунок 10

В протоколе IS-IS фильтрация может осуществляться только в процессе редистрибуции.

Текущая конфигурация на маршрутизаторах следующая.

Маршрутизатор 1 работает на первом уровне как маршрутизатор внутри зоны.

```
EcoRouter_1#show run
router isis 1
is-type level-1
net 49.0001.0000.0000.0001.00
!
interface e2
```

```

ip mtu 1500
ip address 192.168.1.1/24
ip router isis 1
!
interface e1
ip mtu 1500
ip address 10.10.10.1/30
ip router isis 1
!
!
port te0
mtu 9728
service-instance 1
encapsulation untagged
no rewrite
connect ip interface e1

```

Маршрутизатор 2 работает на уровне 1 и 2.

```

EcoRouter_2#show run
router isis 1
  net 49.0001.0000.0000.0002.00
!
interface e2
  ip mtu 1500
  ip address 10.10.10.5/30
  ip router isis 1
!
interface e1
  ip mtu 1500
  ip address 10.10.10.2/30
  ip router isis 1
!
port te0
  mtu 9728
  service-instance 1
  encapsulation untagged
  no rewrite
  connect ip interface e1
!
port tel
  mtu 9728
  service-instance 1
  encapsulation untagged
  no rewrite
  connect ip interface e2

```

Маршрутизатор 3 работает только на 2 уровне.

```

EcoRouter_3#show run
router isis 1
  is-type level-2-only
  net 49.0001.0000.0000.0003.00
!
interface e2
  ip mtu 1500

```

```

ip address 172.16.10.1/24
ip router isis 1
!
interface e1
ip mtu 1500
ip address 10.10.10.6/30
ip router isis 1
!
port te0
mtu 9728
service-instance 1
encapsulation untagged
no rewrite
connect ip interface e1

```

Вывод таблиц маршрутизации для топологии.

```

EcoRouter_1#sh ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter
area
      * - candidate default

IP Route Table for VRF "default"
C    10.10.10.0/30 is directly connected, e1
i L1  10.10.10.4/30 [115/20] via 10.10.10.2, e1, 00:00:21
C    192.168.1.0/24 is directly connected, e2
EcoRouter_2#sh ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter
area
      * - candidate default

IP Route Table for VRF "default"
C    10.10.10.0/30 is directly connected, e1
C    10.10.10.4/30 is directly connected, e2
i L2  172.16.10.0/24 [115/20] via 10.10.10.6, e2, 00:00:02
i L1  192.168.1.0/24 [115/20] via 10.10.10.1, e1, 00:00:03
EcoRouter_3#sh ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter
area
      * - candidate default

IP Route Table for VRF "default"
i L2  10.10.10.0/30 [115/20] via 10.10.10.5, e1, 00:00:09

```

```
C      10.10.10.4/30 is directly connected, e1
C      172.16.10.0/24 is directly connected, e2
i L2    192.168.1.0/24 [115/30] via 10.10.10.5, e1, 00:00:09
```

Создание списка фильтров для ограничения маршрутного обновления о сети 192.168.1.0/24 от EcoRouter_1 к EcoRouter_3.

```
EcoRouter_3(config)#policy-filter-list 20 deny 192.168.1.0 0.0.0.255
```

где **20** - номер списка фильтров,

deny - запрещающий аргумент,

192.168.1.0 0.0.0.255 - сеть, маршрутное обновление о которой ограничено.

После этого следует размещение списка фильтров в контекст маршрутизации граничного маршрутизатора.

```
EcoRouter_2(config)#router isis 1
EcoRouter_2(config-router)#redistribute isis level-1 into level-2
distribute-list 20
```

где **redistribute** - команда перераспределения маршрутов,

isis level-1 into level-2 - аргумент, указывающий, что маршрут забирается из isis внутри зоны и передается за границы зоны,

distribute-list 20 - аргумент, указывающий на созданный список фильтров с именем.

Результатом выполнения данной команды будет отсутствие информации о данной сети на EcoRouter 3.

```
EcoRouter_3#sh ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter
area
      * - candidate default

IP Route Table for VRF "default"
i L2    10.10.10.0/30 [115/20] via 10.10.10.5, e1, 01:35:24
C      10.10.10.4/30 is directly connected, e1
C      172.16.10.0/24 is directly connected, e2
```

7.1.4 Настройка фильтрации маршрутной информации в OSPF

Между маршрутизаторами 1 и 2 настроена динамическая маршрутизация с помощью протокола OSPF.

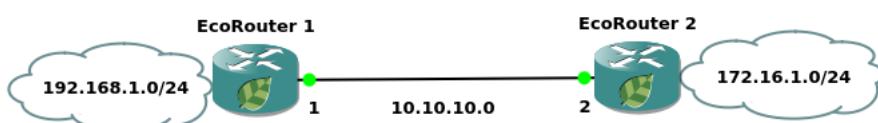


Рисунок 11

Текущая конфигурация на маршрутизаторах следующая:

```

EcoRouter_1#show run
!
router ospf 1
log-adjacency-changes
network 10.10.10.0/24 area 0.0.0.0
network 192.168.1.0/24 area 0.0.0.0
!
interface e2
ip mtu 1500
ip address 192.168.1.1/24
!
interface e1
ip mtu 1500
ip address 10.10.10.1/24
!
port te0
mtu 9728
service-instance 1
  encapsulation untagged
  no rewrite
  connect ip interface e1
EcoRouter_2#show run
!
router ospf 1
log-adjacency-changes
network 10.10.10.0/24 area 0.0.0.0
network 172.168.1.0/24 area 0.0.0.0
!
interface e2
ip mtu 1500
ip address 172.168.1.1/24
!
interface e1
ip mtu 1500
ip address 10.10.10.2/24
!
port te0
mtu 9728
service-instance 1
  encapsulation untagged
  no rewrite
  connect ip interface e1

```

Вывод таблицы маршрутизации на EcoRouter_1 и EcoRouter_2.

```

EcoRouter_1#show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter
area
      * - candidate default

IP Route Table for VRF "default"

```

```
C      10.10.10.0/24 is directly connected, e1
O      172.168.1.0/24 [110/20] via 10.10.10.2, e1, 00:18:28
C      192.168.1.0/24 is directly connected, e2

Gateway of last resort is not set

EcoRouter_2#sh ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter
area
      * - candidate default

IP Route Table for VRF "default"
C      10.10.10.0/24 is directly connected, e1
C      172.168.1.0/24 is directly connected, e2
O      192.168.1.0/24 [110/20] via 10.10.10.1, e1, 00:18:47
```

Gateway of last resort is not set

Настройка фильтрации получения анонсов маршрутной информации от Ecorouter 2 на маршрутизаторе Ecorouter 1.

```
EcoRouter_1(config)#policy-filter-list 10 remark FilterForER2
```

Создание списка фильтров с номером **10**. Добавление комментария для этого списка фильтров.

```
EcoRouter_1(config)#policy-filter-list 10 deny 172.168.1.0 0.0.0.255
```

Создание правила списка фильтров, которое запрещает помещение маршрута в сеть 172.168.1.0/24 с таблицей маршрутизации.

После создания списка фильтров нужно применить к процессу маршрутизации. До применения фильтр работать не будет.

```
EcoRouter_1(config)#router ospf 1
EcoRouter_1(config-router)#distribute-list 10 in
```

В контексте конфигурации протокола маршрутизации следует указать номер нужного списка фильтров и направление.

Для OSPF использование списков фильтров возможно только во входящем направлении, так как в этом направлении не фильтруются LSA, а фильтруются маршруты, которые помещаются в таблицу маршрутизации.

```
EcoRouter_1#sh ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter
area
      * - candidate default
```

```

IP Route Table for VRF "default"
C    10.10.10.0/24 is directly connected, e1
C    192.168.1.0/24 is directly connected, e2

Gateway of last resort is not set

```

В таблице маршрутизации данная сеть отсутствует.

```

EcoRouter_1#sh ip ospf database

OSPF Router with ID (192.168.1.1) (Process ID 1 VRF default)

    Router Link States (Area 0.0.0.0)

Link ID      ADV Router      Age      Seq#      CkSum      Link count
172.168.1.1   172.168.1.1   1552      0x80000007 0x8c39 2
192.168.1.1   192.168.1.1   1556      0x80000006 0x4447 2

    Net Link States (Area 0.0.0.0)

Link ID      ADV Router      Age      Seq#      CkSum
10.10.10.1    192.168.1.1   1556      0x80000001 0x1fcda
EcoRouter_1#

```

Информация о этой сети присутствует в базе состояния каналов OSPF.

7.2 Префиксные списки (prefix-list)

Префиксные списки (prefix-list) представляют собой альтернативу policy-filter листам, применяемым во многих командах фильтрации маршрутов, и обладают рядом преимуществ. Префиксные списки в меньшей степени загружают процессор, что повышает производительность маршрутизаторов.

7.2.1 Команды просмотра списков префиксов

Команды **show ip prefix-list <имя>** и **show ip prefix-list summary** выводят общую информацию о списке префиксов, а **show ip prefix-list detail <имя>** выдает статистику по совпадениям в списке префиксов (hit count) и по совпадению в приложениях (route-map), где используется список префиксов (refcount).

Таблица 27

Команда	Описание
show ip prefix-list <имя>	Просмотр определенного списка префиксов
show ip prefix-list summary	Просмотр всех списков
show ip prefix-list detail <имя>	Просмотр статистики по совпадениям со списком префиксов (hit count), по совпадению в приложениях (route-map), где используется префикс лист (refcount)

7.2.2 Настройка префиксных списков

Префиксные списки проверяются по порядку, строка за строкой, до тех пор, пока не будет обнаружено соответствие тому или иному критерию. Как только соответствие обнаруживается, начинается обработка пакета. По умолчанию все пакеты, в явном виде не разрешенные в списке префиксов, запрещены (неявный оператор **deny all** для всех пакетов, которые не удовлетворяют ни одному из критериев).

Для создания префиксного списка требуется в режиме конфигурации ввести команду **ip prefix-list**, после которой должно быть указано имя списка. Можно воспользоваться нумерацией операторов, для чего употребляется ключевое слово **seq** с указанием после него номера, который присваивается записи. Запись может иметь любой номер из диапазона <1-4294967295> (чем меньше номер, тем раньше проверяется запись). Если номер первой записи 10, а последней 15, то в любое время в список можно будет добавить записи с номерами 11,12,13,14. Если в новом списке не указать номер первой записи, то по умолчанию он будет назначен равным 5. Последующие записи автоматически будут нумероваться с шагом 5. Для отключения режима автоматического присвоения номера записям используется команда **no ip prefix-list sequence-number**. Для определения сети, информация о которой должна передаваться другим маршрутизаторам, служит ключевое слово **permit**, для запрета - **deny**, соответственно. Таким образом, команда приобретает следующий вид:

```
ip prefix-list <имя> seq <номер> (permit | deny) <подсеть/маска> (ge | le | eq <значение>).
```

Для префиксного списка можно указать **description** (до 80 символов) командой: **ip prefix-list <имя> description <текст>**.

Помимо указания конкретной подсети и маски, гибкость префиксных списков позволяет отбирать подсети с учетом длины масок с помощью операторов **ge**, **le**, **eq**. Параметр **ge** применяется для отбора префиксов, длина которых больше, чем указанное значение в поле «значение». С помощью ключевого слова **le** можно отобрать префиксы, длина которых меньше, чем указанное значение. Ключевое слово **eq** точно определяет значение маски для префикса. Если не введены ни **ge**, ни **le**, ни **eq**, это соответствует условию точного совпадения префикса с тем, который указывается в списке. Приведем пример для 6 указанных подсетей:

1. 10.0.0.0/8
2. 10.128.0.0/9
3. 10.1.1.0/24
4. 10.1.2.0/24
5. 10.128.10.4/30
6. 10.128.10.8/30

Соответствие префиксных списков

Таблица 28

Команда	Номера подсетей, соответствующие условию
ip prefix-list permit 10.0.0.0/8	1

Команда	Номера подсетей, соответствующие условию
ip prefix-list permit 10.128.0.0/9	2
ip prefix-list permit 10.0.0.0/8 ge 9	2,3,4,5,6
ip prefix-list permit 10.0.0.0/8 eq 24	3,4
ip prefix-list permit 10.0.0.0/8 le 28	1,2,3,4
ip prefix-list permit 0.0.0.0/0	Нет совпадений
ip prefix-list permit 0.0.0.0/0 le 32	Все подсети. В этом случае вместо 0.0.0.0/0 le 32 при конфигурировании префикс-листа можно указать параметр any

Пример команды только для рекламы подсетей 10.0.0.0 с масками от 10 до 20 может выглядеть следующим образом:

```
ip prefix-list TEST seq 5 permit 10.0.0.0/8 ge 10 le 20
ip prefix-list TEST seq 10 deny all
```

ВНИМАНИЕ: в текущей реализации при использовании префиксных списков в конфигурации BRAS условия **ge**, **le**, **eq** не учитываются.

Для удаления префиксного списка служит команда **no ip prefix-list <имя>**.

7.3 Filter-map

Для фильтрации трафика на уровнях L2 и L3 в EcoRouterOS применяются списки доступа (filter-map), содержащие правила.

В EcoRouterOS общая логика при создании filter-map следующая:

- Собственно создание filter-map при помощи команды **filter-map {ethernet | ipv4} <FILTER_MAP_NAME> [<SEQUENCE_NUMBER>]**.
- Задание правила вида **match <CONDITION>**, где **<CONDITION>** - условие или условия для проверки пакетов (подробнее см. в соответствующих разделах).
- Задание действия вида **set <ACTION>**, где **<ACTION>** - действие, которое будет применено к пакетам, удовлетворяющим критериям из **<CONDITION>** (подробнее см. в соответствующих разделах).

Правила в зависимости от протоколов и условий могут задаваться по-разному.

Для каждого filter-map правила проверяются последовательно, в том порядке, в котором они присутствуют в выводе команды **show filter-map ipv4** или **show filter-map ethernet** соответственно.

Если в самом правиле присутствуют несколько признаков трафика одновременно, это эквивалентно логической операции "И", то есть, правило будет применено только, если пакет удовлетворяет всем признакам, перечисленным в правиле.

Пример:

```
filter-map ipv4 example01 10
match tcp 10.0.0.0/24 eq 40 any eq 179 not-rst syn ack
set discard
```

Этот filter-map **example01** запрещает TCP-пакеты с IP-адресами источника (**10.0.0.0-10.0.0.255**) и **40** портом до любого IP-адреса получателя с портом **179**, который содержит флаги **SYN, ACK** и не содержит **RST**.

Для реализации логической операции "ИЛИ" необходимо создать несколько правил. Тогда к пакету будет применено то правило, условиям которого он удовлетворяет.

Например, если необходимо разрешить любой TCP-пакет с флагами SYN и ACK или пакет с флагом FIN, то конструкция списка будет состоять из следующих записей:

```
filter-map ipv4 example02 10
match tcp any any syn ack
match tcp any any fin
set accept
```

В конце каждого filter-map есть неявное правило, запрещающее всё, что не разрешено в данном списке доступа: **any any discard**.

7.3.1 Настройка L2 filter-map

Еще один вид списка доступа в EcoRouterOS - это filter-map ethernet, который позволяет фильтровать фреймы по значениям полей в L2-заголовке.

По структуре правил filter-map ethernet отличается тем, что в правилах указываются MAC-адреса источника и назначения, инверсные маски (wildcard) MAC-адресов и значения поля ethertype (опционально).

filter-map ethernet создается в конфигурационном режиме. Для одного действия может существовать несколько правил.

Синтаксис создания filter-map ethernet, добавления правил и действий в filter-map ethernet требует указать следующие параметры:

- имя и sequence самого filter-map ethernet - <**FILTER_MAP_ETHERNET_LIST**> [<**SEQUENCE_NUMBER**>];
- правило - **match {<SOURCE_MAC> <SRC_WILDCARD> | any | host <SOURCE_MAC>} {<DESTINATION_MAC> <DST_WILDCARD> | any | host <DESTINATION_MAC>} [<ETHERTYPE>];**
- действие - **set <ACTION>**.

Параметры filter-map ethernet описаны в таблице ниже.

Таблица 29

Параметр	Описание
FILTER_MAP_ETHERNET_LIST	Имя списка фильтрации, может принимать любое значение

Параметр	Описание
SEQUENCE_NUMBER	Номер приоритета выполнения, допустимые значения 0-65535. Если значение не задано, то параметр для созданного filter-map ethernet автоматически получит последующее свободное значение с шагом 10
SOURCE_MAC	MAC-адрес источника, задается в одном из трех форматов: XX-XX-XX-XX-XX-XX, XX:XX:XX:XX:XX:XX, XXXX.XXXX.XXXX. Если под правило должны попадать все адреса, значение параметра должно быть any . Если под правило должен подпадать единственный адрес, в значении параметра указывается host <MAC-адрес> .
SRC_WILDCARD	Инверсная маска источника, задается в одном из трех форматов: XX-XX-XX-XX-XX-XX, XX:XX:XX:XX:XX:XX, XXXX.XXXX.XXXX.
DESTINATION_MAC	MAC-адрес назначения, задается в одном из трех форматов: XX-XX-XX-XX-XX-XX, XX:XX:XX:XX:XX:XX, XXXX.XXXX.XXXX. Если под правило должны попадать все адреса, значение параметра должно быть any . Если под правило должен подпадать единственный адрес, в значении параметра указывается host <MAC-адрес> .
DST_WILDCARD	Инверсная маска назначения, задается в одном из трех форматов: XX-XX-XX-XX-XX-XX, XX:XX:XX:XX:XX:XX, XXXX.XXXX.XXXX.
ETHERTYPE	Значение поля ethertype. Может быть указано шестнадцатеричное значение поля в диапазоне (0x600 - 0xffff) или одно из следующих обозначений: 802dot1x - IEEE 802.1X Ethertype - 0x888E, ip4 - IPv4 Ethertype - 0x0800, ip6 - IPv6 Ethertype - 0x86dd, l2-is-is - L2 IS-IS Ethertype - 0x22F4, lldp - LLDP Ethertype - 0x88CC, mpls - MPLS Ethertype - 0x8847, pppoe-discovery - PPPoE Discovery Ethertype - 0x8863, pppoe-session - PPPoE Session Ethertype - 0x8864, qinq - QinQ Ethertype - 0x88A8, vlan - VLAN Ethertype - 0x8100.
set <ACTION>	
set accept	Разрешить
set discard	Запретить без отправки ICMP-уведомления
set reject	Запретить с отправкой ICMP-уведомления
set class-map <NAME>	Пакетам, попавшим под действие правила, присваивается указанный класс трафика class-map. Класс должен быть заранее создан (подробнее см. QoS)

Параметр	Описание
set port <NAME>	Пакеты, попавшие под действие правила, перенаправляются в указанный порт. NAME - имя порта (обозначения портов подробнее описаны в разделе Виды интерфейсов)

В конце любого filter-map ethernet в неявном виде встроено запрещающее правило **any any reject**.

После того как filter-map ethernet создан, наполнен правилами, и для них указано действие, его можно назначить для сервисного интерфейса с указанием направления. Под направлением в данном случае подразумевается момент, когда пакеты, проходящие через интерфейс, будут обработаны списком доступа: для filter-map ethernet возможно только направление **in** (при "входе" в интерфейс). На одном интерфейсе может быть применено несколько filter-map ethernet.

Для назначения filter-map ethernet на сервисный интерфейс используется команда контекстного режима настройки сервисного интерфейса **set filter-map in <FILTER_MAP_ETHERNET_LIST> [<SEQUENCE>]**.

Пример настройки filter-map ethernet

Задача: запретить arp-запросы от клиента с адресом **0000.0000.000c**.

```
ecorouter(config)#filter-map ethernet primer 10
ecorouter(filter-map-ethernet)#match host 0000.0000.000c any 0x806
ecorouter(filter-map-ethernet)#set discard
ecorouter(filter-map-ethernet)#ex
ecorouter(config)#filter-map ethernet primer 15
ecorouter(filter-map-ethernet)#match 0000.0000.0010 ffff.ffff.ffff any
ecorouter(filter-map-ethernet)#set port ge0
ecorouter(filter-map-ethernet)#ex
ecorouter(config)#filter-map ethernet primer 20
ecorouter(filter-map-ethernet)#match any any
ecorouter(filter-map-ethernet)#set accept
ecorouter(filter-map-ethernet)#ex
```

0x806 – значение ethertype, соответствующее протоколу **arp**. **Filter-map ethernet primer 20** разрешает весь остальной трафик, без этого правила по умолчанию сработало бы правило **any any discard**.

```
ecorouter(config)#port
te0
ecorouter(config-port)#service-instance
1
ecorouter(config-service-instance)#set filter-map in primer 10
ecorouter(config-service-instance)#set filter-map in primer 15
ecorouter(config-service-instance)#set filter-map in primer 20
```

7.3.2 Настройка L3 filter-map

Для управления трафиком разных направлений для L3 интерфейса могут применяться списки доступа filter-map. Под направлением в данном случае подразумевается момент, когда

пакеты, проходящие через интерфейс, будут обработаны списком доступа: при "входе" в интерфейс – указание направления in, при "выходе" – направление out. На одном интерфейсе может быть применено несколько списков доступа в одном направлении. Каждый список доступа может быть применен к нескольким интерфейсам одновременно.

Использование filter-map производится в два этапа.

1. Создание и наполнение правилами.
2. Привязка к интерфейсу.

Создание filter-map производится в конфигурационном режиме. Для создания filter-map требуется выполнить следующие действия (в результате будет создан filter-map, содержащий одно правило):

1. Первая строка. Ввести команду **filter-map ipv4 <FILTER_MAP_NAME> [<SEQUENCE_NUMBER>]**, где <FILTER_MAP_NAME> - имя списка доступа, <SEQUENCE_NUMBER> - порядковый номер правила в списке доступа. Подробнее параметры описаны в таблице ниже.
2. Вторая строка. Указать правило, на соответствие которому будут проверяться пакеты, следующего вида: **match <PROTOCOL> <SRC_ADDRESS> [<PORT_CONDITION>] <DST_ADDRESS> [<PORT_CONDITION>] [dscp <DSCPVALUE>] [<FLAG>]**. Подробнее параметры описаны в таблицах ниже.
3. Третья строка. Указать действие, которое будет применяться к пакетам, удовлетворяющим условиям правила, следующего вида **set <ACTION>**. Подробнее параметры описаны в таблице ниже.

Список доступа может содержать несколько правил. Для добавления правила в существующий список доступа следует повторить шаги, описанные выше. В качестве <FILTER_MAP_NAME> следует указывать имя списка доступа, куда правило должно быть добавлено. Правило должно иметь уникальный номер <SEQUENCE> в рамках одного filter-map.

В конце любого **filter-map ipv4** в неявном виде встроено запрещающее правило **any any reject**.

Общие параметры **filter-map ipv4** описаны в таблице ниже.

Таблица 30

Параметр	Описание
FILTER_MAP_NAME	Имя списка фильтрации, может принимать любое значение
SEQUENCE_NUMBER	Номер приоритета выполнения, допустимые значения 0-65535. Если значение не задано, то параметр для созданного filter-map ethernet автоматически получит последующее свободное значение с шагом 10
PROTOCOL	Значение поля protocol. Может быть указано значение поля в диапазоне (0-255) или одно из следующих обозначений: ipinip; icmp; gre; igmp; pim;

Параметр	Описание
	rsvp; ospf; vrrp; ipcomp; any (любой протокол); udp (внимание, для данного протокола доступны дополнительные параметры <PORT_CONDITION>); tcp (внимание, для данного протокола доступны дополнительные параметры <PORT_CONDITION> и <FLAG>)
SRC_ADDRESS	IP-адрес источника, задается в одном из следующих форматов: A.B.C.D/M (IP-адрес с маской), A.B.C.D K.L.M.N (IP-адрес с инверсной маской), host A.B.C.D (если под правило должен подпадать единственный адрес), any (если под правило должны попадать все адреса)
DST_ADDRESS	IP-адрес назначения, задается в одном из следующих форматов: A.B.C.D/M (IP-адрес с маской), A.B.C.D K.L.M.N (IP-адрес с инверсной маской), host A.B.C.D (если под правило должен подпадать единственный адрес), any (если под правило должны подпадать все адреса)
DSCPVALUE	Значение DSCP (Differentiated Services Code Point) для проверки пакета, целое число от 0 до 63
set <ACTION>	
set accept	Разрешить
set discard	Запретить без отправки ICMP-уведомления
set reject	Запретить с отправкой ICMP-уведомления
set nexthop <A.B.C.D>	Указать IP-адрес next hop. Пакеты, попавшие под действие правила, отсылаются на адрес next-hop с учетом существующих маршрутов в RIB
set class-map <NAME>	Пакетам, попавшим под действие правила, присваивается указанный класс трафика class-map. Класс должен быть заранее создан (подробнее см. QoS Архитектура QoS)
set vrf <VRF NAME> [A.B.C.D]	Для пакетов, попавших под действие правила, будет использоваться таблица маршрутизации vrf, где VRF NAME - имя необходимого vrf. Для данного vrf может быть указан IP-адрес next hop (оноционально)

При указании протокола **udp** вторая строка команды создания filter-map будет иметь следующий вид: **match udp <SRC_ADDRESS> [<PORT_CONDITION>] <DST_ADDRESS> [<PORT_CONDITION>] [dscp <DSCPVALUE>]**.

Дополнительные параметры при указании **udp** описаны в таблице ниже.

Таблица 31

Параметр	Описание
PORT_CONDITION	Условие для значения порта. Может быть указано одно из следующих значений: {{eq gt lt} {tftp bootp <0-65535>} range <0-65535> <0-65535>}
Значения PORT_CONDITION	
eq	Номер порта равен
gt	Номер порта больше, чем
lt	Номер порта меньше, чем
tftp	UDP(69)
bootp	UDP(67)
<0-65535>	Точный номер порта, любое значение из указанного диапазона

Параметр	Описание
range <0-65535> <0-65535>	Номер порта входит в диапазон

При указании протокола **tcp** вторая строка команды создания filter-map будет иметь следующий вид: **match tcp <SRC_ADDRESS> [<PORT_CONDITION>] <DST_ADDRESS> [<PORT_CONDITION>] [dscp <DSCPVALUE>] [<FLAG>]**.

Дополнительные параметры при указании **tcp** описаны в таблице ниже.

Таблица 32

Параметр	Описание
PORT_CONDITION	Условие для значения порта. Может быть указано одно из следующих значений: {{eq gt lt} {ftp ssh telnet www <0-65535>} range <0-65535> <0-65535>}
FLAG	Значения флага, по которым может производиться обработка пакетов. Может быть указано одно из следующих значений (префикс not- означает, что указанный флаг не установлен): ack not-ack fin not-fin psh not-psh rst not-rst syn not-syn urg not-urg ack - установлен флаг ACK (номер подтверждения), fin - установлен флаг FIN (завершение соединения), psh - установлен флаг PSH (инструктирует получателя протолкнуть данные, накопившиеся в приёмном буфере, в приложение пользователя), rst - установлен флаг RST (оборвать соединение, очистить буфер), syn - установлен флаг SYN (синхронизация номеров последовательности), urg - установлен флаг URG (указатель важности), not-ack - не установлен флаг ACK, not-fin - не установлен флаг FIN, not-psh - не установлен флаг PSH, not-rst - не установлен флаг RST, not-syn - не установлен флаг SYN, not-urg - не установлен флаг URG. Можно перечислить несколько флагов через пробел. При этом правило сработает, если в пакете будут установлены все перечисленные флаги. Например, правило not-rst syn ack сработает, если пакет содержит флаги SYN и ACK, но не содержит RST
Значения PORT_CONDITION	
eq	Номер порта равен
gt	Номер порта больше, чем
lt	Номер порта меньше, чем
ftp	TCP(21)
ssh	TCP(22)
telnet	TCP(23)
www	TCP(HTTP-80)
<0-65535>	Точный номер порта, любое значение из указанного диапазона
range <0-65535> <0-65535>	Номер порта входит в диапазон

Пример создания списка доступа и добавления правил в него

Создание списка доступа производится в конфигурационном режиме:

```
ecorouter(config)#filter-map ipv4 example 10
match udp 10.10.10.0/24 20.20.20.0/24 eq 22
set accept
```

Здесь:

- example – имя списка доступа,

- 10 - номер последовательности правила в списке доступа,
- udp – указание на ожидаемый протокол,
- 10.10.10.0/24 – указание сети-источника пакетов с префиксом, трафик из которой разрешается для прохождения,
- 20.20.20.0/24 – указание сети назначения с префиксом, трафик в которую разрешается для прохождения,
- eq 22 – аргумент, указывающий на точный номер порта назначения,
- accept – разрешающий аргумент (трафик, удовлетворяющий условиям правила будет пропускаться).

Добавление правила к данному списку доступа (для пакетов, удовлетворяющих правило, также будет выполняться accept, правило будет проверяться вторым в списке доступа с именем example). Правило добавляет условие для проверки. Действие для всего списка выполняется одно и то же. Проверка правил внутри списка доступа производится в соответствии с указанными для них значениями <SEQUENCE>.

```
ecorouter(config)#filter-map ipv4 example 20
match 1 host 122.168.1.15 host 172.20.100.1
```

Здесь:

- example – имя списка доступа,
- 20 - номер последовательности правила в списке доступа,
- 1 – указание на протокол, в данном случае ICMP,
- host 122.168.1.15 – аргумент, указывающий на конкретный IP-адрес источник пакетов (указание маски не требуется),
- host 172.20.100.1 – аргумент, указывающий на конкретный IP-адрес назначения пакетов (указание маски не требуется).

Добавление правила к данному списку доступа (для пакетов, удовлетворяющих правило, также будет выполняться accept, правило будет проверяться третьим в списке доступа с именем example).

```
ecorouter(config)#filter-map ipv4 example 30
match ospf 192.168.32.0 0.0.7.255 any
```

Здесь:

- example – имя списка доступа,
- 30 - номер последовательности правила в списке доступа,
- ospf – указание на протокол, в данном случае ospf,
- 192.168.32.0 0.0.7.255 – аргумент, указывающий на IP-адрес источника пакетов с инверсной маской,
- any - аргумент, указывающий на все IP-адреса назначения пакетов.

Просмотр filter-map

Для просмотра созданных списков доступа L3 служит команда show filter-map ipv4. Она показывает только списки доступа без указания их привязок к интерфейсам.

```
ecorouter#show filter-map ipv4
Filter map example
```

```

Filter 10
match udp 10.10.10.0/24 20.20.20.0/24 eq 22
match 1 host 192.168.1.15 host 172.20.100.1
match ospf 192.168.32.0 0.0.7.255 any
set accept
Filter map TEST
Filter 20
match any host 10.210.10.151 any
set accept

```

Для назначения списка доступа на интерфейс используется команда контекстного режима настройки интерфейса `set filter-map {in | out} <FILTER_MAP_NAME> [<SEQUENCE>]`. К одному интерфейсу можно привязать несколько filter-map. Здесь параметр `<SEQUENCE>` в явном виде задается для каждого filter-map (а не для входящих в него правил!). Все привязанные к интерфейсу filter-map будут выполняться в порядке увеличения значений `<SEQUENCE>`. Неявное правило "запретить все" будет размещено после правил из всех привязанных filter-map.

Пример привязки filter-map к интерфейсу

```

ecorouter(config)#interface e20
ecorouter(config-if)#set filter-map in example 10
ecorouter(config-if)#set filter-map out TEST 20

```

Если при привязке filter-map к интерфейсу не указывать значение `<SEQUENCE>`, то для каждого привязываемого списка доступа его значение будет присваиваться автоматически с инкрементом 10.

Один и тот же список доступа может быть назначен на несколько интерфейсов одновременно.

В EcoRouterOS может быть создано до 64 тысяч filter-map. Однако существует ограничение на количество "активных" экземпляров filter-map, то есть, назначенных на L3 интерфейс. Можно настроить не более 64-х привязок списков доступа к интерфейсам. Это ограничение не зависит от количества созданных списков доступа или интерфейсов.

Управление списками доступа может осуществляться как из основного маршрутизатора, так и из виртуальных. При этом списки доступа виртуального маршрутизатора будут действовать только в его пределах, а списки доступа основного - соответственно, только в пределах основного.

Просмотр привязанных к интерфейсу списков доступа производится, например, при помощи команды `show counters interface <INTERFACE_NAME> filter-map {in | out}`.

```

show counters interface e20 filter-map out
Interface e20
Filter map TEST
Filter 10 [0 packets]
    match any host 10.210.10.151 any
    set accept

```

7.3.3 Команды просмотра L2 filter-map

Для просмотра информации по всем созданным L2 спискам фильтрации используется команда режима администрирования

```
show filter-map ethernet [<FILTER_NAME>]
```

, где **FILTER_NAME** - название списка фильтрации.

Пример:

Таблица 33

Консоль	Комментарий
ecorouter#show filter-map ethernet	Вывести информацию обо всех списках фильтрации L2
Filter map FILTER Filter 10 match host 0000.0000.0001 host 0000.0000.0004 match host 0000.0000.0001 any 0x806 set accept Filter map test Filter 10 match host 0000.0000.0001 any 0x806 set discard	Вывод информации обо всех списках фильтрации L2
ecorouter#show filter-map ethernet FILTER	Вывести информацию о списке фильтрации с именем FILTER
Filter map FILTER Filter 10 match host 0000.0000.0001 host 0000.0000.0004 match host 0000.0000.0001 any 0x806 set accept	Вывод информации о списке с именем FILTER

Просмотр счетчиков

Для просмотра показателей счетчиков для L2 списков фильтрации используется команда режима администрирования

```
show counters port <NAME> filter-map {in | out}
```

Параметры команды описаны в таблице ниже.

Таблица 34

Название	Описание
<NAME>	Название порта (см. Виды интерфейсов)
in	Направление трафика
out	

Счетчики отображаются по каждому блоку filter-map, но не по каждому правилу.

Пример:

Таблица 35

Консоль	Комментарий
ecorouter#show counters port te0 filter-map in	Вывести значения счетчиков filter-map для порта te0 по входящему трафику
Service instance 1 Filter map FILTER Filter 10 [5 packets] match host 0000.0000.0001 host 0000.0000.0004 match host 0000.0000.0001 any 0x806	Вывод команды

Консоль	Комментарий
set accept Filter 20 [6 packets] match host 0000.0000.0002 any set discard	

Для того чтобы узнать, какие списки фильтрации привязаны к данному порту, используется команда режима администрирования **show port <NAME>**, где **<NAME>** - название порта.

Пример:

Таблица 36

Консоль	Комментарий
ecorouter#show port te0 10 Gigabit Ethernet [none] port te0 is up MTU: 9728 LACP priority: 32767 Input packets 13, bytes 3308, errors 0 Output packets 10, bytes 1340, errors 0 Service instance te0.1 is up ingress encapsulation untagged ingress rewrite none egress encapsulation untagged egress none Connect bridge test symmetric filter-map in FILTER Input packets 13, bytes 3308 Output packets 10, bytes 1340	Вывести информацию по порту te0 Вывод команды

7.3.4 Команды просмотра L3 filter-map

Просмотр всех созданных списков доступа L3 осуществляется при помощи команды административного режима **show filter-map ipv4**.

```
ecorouter#show filter-map ipv4
Filter map NAME
Filter 10
match any any any
set discard
Filter map TEST
Filter 10
match any host 10.210.10.151 any
set accept
```

Для просмотра определенного списка доступа L3 команда вводится с именем списка: **show filter-map ipv4 <NAME>**.

```
ecorouter#show filter-map ipv4 TEST
Filter map TEST
Filter 10
match any host 10.210.10.151 any
set accept
```

Просмотр всех присоединенных списков доступа L3 на определенном интерфейсе осуществляется командой **show counters interface <NAME> filter-map {in | out}**.

```
ecorouter#show counters interface EXAMPLE
filter-map in
```

```
Interface EXAMPLE
Filter map TEST
Filter 10 [0 packets]
match any any any
set discard
```

7.3.5 Настройка политики для абонентской сессии

Для фильтрации трафика в рамках абонентской сессии (subscriber-service) применяются политики subscriber-policy. Для одной сессии может быть назначено до 10 таких политик. Трафик последовательно будет обрабатываться в соответствии с каждой политикой в соответствии с ее порядковым номером.

Создание subscriber-policy производится в конфигурационном режиме при помощи команды **subscriber-policy <NAME>**, где <NAME> - имя создаваемой сущности.

```
ecorouter(config)#subscriber-policy ?
SUBSCRIBER_POLICY Subscriber policy name
```

После создания subscriber-policy автоматически производится переход в контекстный режим редактирования ее параметров.

```
ecorouter(config)#subscriber-policy subspolname
ecorouter(config-sub-policy) #
```

Параметры subscriber-policy показаны в таблице ниже.

Таблица 37

Параметр	Описание
<BANDWIDTH>	Ширина полосы пропускания в Мбит/сек от 1 до 200
<DESCRIPTION>	Текстовое описание политики

Каждой политике subscriber-policy пользователь может назначить 2 разных правила обработки (filter-map policy): одно для входящего (in) и одно для исходящего (out) трафика. Если filter-map policy не назначен на направление, то трафик соответствующего вида политикой не обрабатывается и не претерпевает никаких изменений. **Внимание:** без задания filter-map policy с ограничениями и привязки его к тому же направлению для subscriber-policy трафик до заданной полосы пропускания ограничиваться не будет!

Назначение для политики subscriber-policy на выбранное направление трафика (in или out) нужной filter-map policy производится в контекстном режиме редактирования параметров subscriber-policy при помощи команды **set filter-map {in | out} <NAME>**, где <NAME> - имя filter-map policy.

Пример настройки subscriber-policy (в данном примере предполагается, что filter-map policy с именем **FMPname** уже создана и настроена; создание и настройка filter-map policy описаны ниже).

```
ecorouter(config)#subscriber-policy subspolname
ecorouter(config-sub-policy) #description Testsubscrpolicy
ecorouter(config-sub-policy) #bandwidth in 200
ecorouter(config-sub-policy) #set filter-map in FMPname
```

Создание и настройка filter-map policy

Создание filter-map policy производится в конфигурационном режиме при помощи команды **filter-map policy ipv4 <NAME>**, где <NAME> - имя создаваемой сущности.

```
ecorouter(config)#filter-map policy ipv4 ?
  FILTER_MAP_POLICY_IPV4 Filter map name
```

После создания filter-map policy автоматически производится переход в контекстный режим редактирования ее параметров.

```
ecorouter(config)#filter-map policy ipv4 FMPname
ecorouter(config-filter-map-policy-ipv4) #
```

Для настройки filter-map policy требуется выполнить следующие действия (в результате внутри filter-map policy будет создано одно правило):

1. Первая строка. Ввести команду **filter-map policy ipv4 <FILTER_MAP_NAME> [<SEQUENCE_NUMBER>]**, где <FILTER_MAP_NAME> - имя списка доступа, <SEQUENCE_NUMBER> - порядковый номер правила в списке доступа. Подробнее параметры описаны в таблице ниже.
2. Вторая строка. Указать правило, на соответствие которому будут проверяться пакеты, следующего вида: **match <PROTOCOL> <SRC_ADDRESS> [<PORT_CONDITION>] <DST_ADDRESS> [<PORT_CONDITION>] [dscp <DSCPVALUE>] [<FLAG>]**. Подробнее параметры описаны в таблицах ниже.
3. Третья строка. Указать действие, которое будет применяться к пакетам, удовлетворяющим условиям правила, следующего вида **set <ACTION>**. Подробнее параметры описаны в таблице ниже.

Список доступа может содержать несколько правил. Для добавления правила в существующий список доступа следует повторить шаги, описанные выше. В качестве <FILTER_MAP_NAME> следует указывать имя списка доступа, куда правило должно быть добавлено. Правило должно иметь уникальный номер <SEQUENCE> в рамках одной filter-map policy.

Общие параметры **filter-map policy** описаны в таблице ниже.

Таблица 38

Параметр	Описание
DIRECTION	Направление трафика, in - входящий трафик, out - исходящий трафик
FILTER_MAP_NAME	Имя списка фильтрации, может принимать любое значение
SEQUENCE_NUMBER	Номер приоритета выполнения, допустимые значения 0-65535. Если значение не задано, то параметр для созданного filter-map ethernet автоматически получит последующее свободное значение с шагом 10
PROTOCOL	Значение поля protocol. Может быть указано значение поля в диапазоне (0-255) или одно из следующих обозначений: ipinip; icmp; gre; igmp; pim; rsvp; ospf; vrrp; ipcomp; any (любой протокол); udp (внимание, для данного протокола доступны дополнительные параметры <PORT_CONDITION>);

Параметр	Описание
	tcp (внимание, для данного протокола доступны дополнительные параметры <PORT_CONDITION> и <FLAG>)
SRC_ADDRESS	IP-адрес источника, задается в одном из следующих форматов: A.B.C.D/M (IP-адрес с маской), A.B.C.D K.L.M.N (IP-адрес с инверсной маской), host A.B.C.D (если под правило должен подпадать единственный адрес), any (если под правило должны попадать все адреса)
DST_ADDRESS	IP-адрес назначения, задается в одном из следующих форматов: A.B.C.D/M (IP-адрес с маской), A.B.C.D K.L.M.N (IP-адрес с инверсной маской), host A.B.C.D (если под правило должен подпадать единственный адрес), any (если под правило должны подпадать все адреса)
DSCPVALUE	Значение DSCP (Differentiated Services Code Point) для проверки пакета, целое число от 0 до 63
set <ACTION>	
set accept	Разрешить. Если в subscriber-policy, где используется данная filter-map policy, задана полоса пропускания (параметр bandwidth), то для этого типа трафика будет применено ограничение скорости до указанных в bandwidth значений
set discard	Запретить без отправки ICMP-уведомления
set redirect <REDIRECTNAME>	Перенаправить HTTP GET на указанный <REDIRECTNAME>, где <REDIRECTNAME> - имя заранее заданного URL (адрес для перенаправления должен начинаться с http://). Пример настройки перенаправления приведен ниже.
set reject	Запретить с отправкой ICMP-уведомления

При указании протокола **udp** вторая строка команды создания filter-map policy будет иметь следующий вид: **match udp <SRC_ADDRESS> [<PORT_CONDITION>] <DST_ADDRESS> [<PORT_CONDITION>] [dscp <DSCPVALUE>]**.

Дополнительные параметры при указании **udp** описаны в таблице ниже.

Таблица 39

Параметр	Описание
PORT_CONDITION	Условие для значения порта. Может быть указано одно из следующих значений: {eq gt lt} {tftp bootp <0-65535>} range <0-65535> <0-65535>}
Значения PORT_CONDITION	
eq	Номер порта равен
gt	Номер порта больше, чем
lt	Номер порта меньше, чем
tftp	UDP(69)
bootp	UDP(67)
<0-65535>	Точный номер порта, любое значение из указанного диапазона
range <0-65535> <0-65535>	Номер порта входит в диапазон

При указании протокола **tcp** вторая строка команды создания filter-map policy будет иметь следующий вид: **match tcp <SRC_ADDRESS> [<PORT_CONDITION>] <DST_ADDRESS> [<PORT_CONDITION>] [dscp <DSCPVALUE>] [<FLAG>]**.

Дополнительные параметры при указании **tcp** описаны в таблице ниже.

Таблица 40

Параметр	Описание
PORT_CONDITION	Условие для значения порта. Может быть указано одно из следующих значений: {{eq gt lt} {ftp ssh telnet www <0-65535>} range <0-65535> <0-65535>}
FLAG	Значения флага, по которым может производиться обработка пакетов. Может быть указано одно из следующих значений (префикс not- означает, что указанный флаг не установлен): ack not-ack fin not-fin psh not-psh rst not-rst syn not-syn urg not-urg ack - установлен флаг ACK (номер подтверждения), fin - установлен флаг FIN (завершение соединения), psh - установлен флаг PSH (инструктирует получателя протолкнуть данные, накопившиеся в приёмном буфере, в приложение пользователя), rst - установлен флаг RST (оборвать соединение, очистить буфер), syn - установлен флаг SYN (синхронизация номеров последовательности), urg - установлен флаг URG (указатель важности), not-ack - не установлен флаг ACK, not-fin - не установлен флаг FIN, not-psh - не установлен флаг PSH, not-rst - не установлен флаг RST, not-syn - не установлен флаг SYN, not-urg - не установлен флаг URG. Можно перечислить несколько флагов через пробел. При этом правило сработает, если в пакете будут установлены все перечисленные флаги. Например, правило not-rst syn ack сработает, если пакет содержит флаги SYN и ACK, но не содержит RST
Значения PORT_CONDITION	
eq	Номер порта равен
gt	Номер порта больше, чем
lt	Номер порта меньше, чем
ftp	TCP(21)
ssh	TCP(22)
telnet	TCP(23)
www	TCP(HTTP-80)
<0-65535>	Точный номер порта, любое значение из указанного диапазона
range <0-65535> <0-65535>	Номер порта входит в диапазон

Задание адреса для перенаправления

```
ecorouter(config)#redirect-url SITEREDIRECT
ecorouter(config-redirect-url)#url http://forredirect.org
```

Пример настроек для обработки трафика в абонентской сессии

В данном примере настроен статический IPoE.

В результате выполнения приведенных ниже настроек на вход (применяется **filter-map policy NAME1**) будет отбрасываться весь icmp-трафик, udp-трафик будет ограничен до 20 Мбит/сек, tcp-трафик будет пропускаться без изменений.

Трафик на выход (применяется **filter-map policy NAME2**) будет ограничен до 5 Мбит/сек, tcp-трафик порта 80 будет перенаправлен на адрес <http://forredirect.org>.

```
!
filter-map policy ipv4 NAME1 10
match icmp any any
set discard
```

```
filter-map policy ipv4 NAME1 20
  match udp any any
    set accept
filter-map policy ipv4 NAME2 10
  match tcp any any eq 80
    set redirect SITEREDIRECT
filter-map policy ipv4 NAME2 20
  match any any any
    set accept
!
subscriber-policy NAME
  bandwidth in 20
  set filter-map in NAME1 10
  bandwidth out 5
  set filter-map out NAME2 10
!
subscriber-service NAME
  set policy NAME
!
ip prefix-list NAME seq 5 permit 10.10.10.100/32 eq 32
!
subscriber-map NAME 10
  match static prefix-list NAME
  set service NAME
!
interface ipoe.1
  ip mtu 1500
  ip address 10.10.10.1/24
```

8 Виртуальные маршрутизаторы

Виртуальный маршрутизатор - технология, позволяющая настроить несколько независимых друг от друга таблиц маршрутизации на одном физическом маршрутизаторе.

Каждая таблица маршрутизации будет находиться в так называемом виртуальном маршрутизаторе (VR). Количество поддерживаемых на одном устройстве виртуальных маршрутизаторов зависит от аппаратной платформы. Диапазон варьируется от 510 до 4094 экземпляров.

Виртуальные маршрутизаторы полностью изолированы между собой и между основным маршрутизатором (Default Router), в котором они созданы.

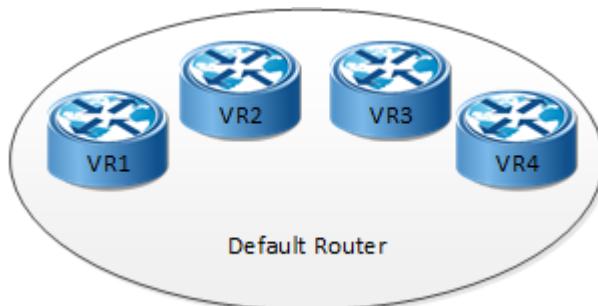


Рисунок 12

8.1 Команды настройки виртуальных маршрутизаторов

Для создания виртуального маршрутизатора (или изменения настроек уже созданного) используется команда конфигурационного режима **virtual-router <NAME>**. Задаваемое имя маршрутизатора чувствительно к регистру и не должно превышать 12 символов. В названиях маршрутизаторов разрешены только строчные и прописные латинские буквы и цифры.

При создании виртуального маршрутизатора ему автоматически добавляется профиль безопасности по умолчанию.

В режиме настройки виртуального маршрутизатора доступны команды, приведенные в таблице ниже.

Таблица 41

Команда	Описание
bind <INTERFACE_NAME>	Привязать интерфейс к виртуальному маршрутизатору. ВНИМАНИЕ При передаче интерфейса из основного маршрутизатора в виртуальный или обратно все настройки интерфейса сбрасываются
configuration file <имя файла>	Создание файла для сохранения конфигурации виртуального маршрутизатора
description <TEXT>	Создание комментария к виртуальному маршрутизатору
load {bgp isis ospf pim rip vrrp}	Команда добавления протоколов в виртуальный маршрутизатор: bgp добавить протокол bgpv4, isis добавить протокол isis, ospf добавить протокол ospfv2, pim добавить протокол pimv2, rip добавить протокол ripv2,

Команда	Описание
	vrrp добавить протокол vgrp

Для входа в CLI созданного виртуального маршрутизатора используется команда административного режима **login virtual-router <NAME>**.

CLI виртуального маршрутизатора аналогичен основному, но урезан по функционалу. Например, в виртуальных маршрутизаторах нет портов (L2 интерфейсов), нельзя создавать L3 интерфейсы (только настраивать переданные из основного маршрутизатора).

L3 - interfaces

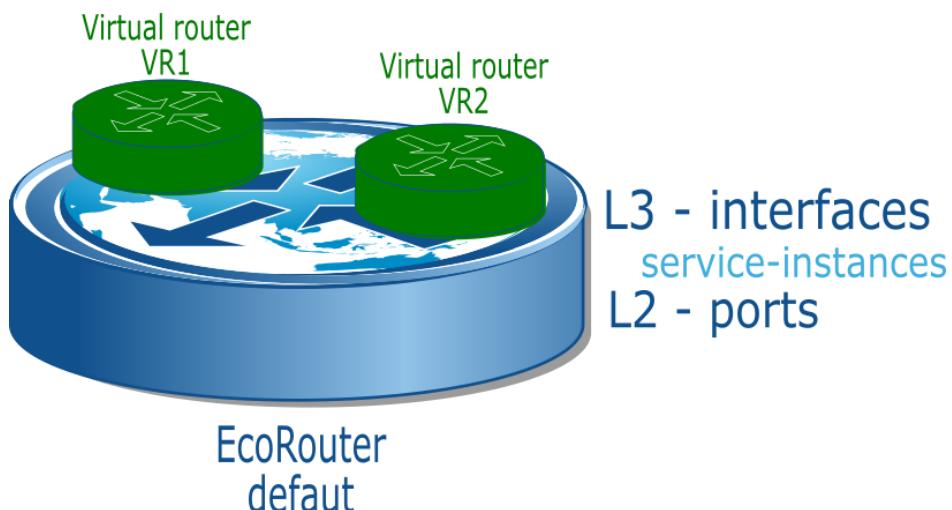


Рисунок 13

Настройки L2 функций всегда осуществляются в основном маршрутизаторе.

Например, если требуется создать бридж и погрузить в него L3 интерфейс из виртуального маршрутизатора, то необходима следующая последовательность действий:

- создать бридж и интерфейс в основном маршрутизаторе,
- в нем же привязать к бриджу порты и интерфейс,
- настроить операции над тегами,
- после чего передать интерфейс в виртуальный маршрутизатор,
- зайти в его CLI и задать IP-адрес интерфейса.

8.2 Пример настройки виртуального маршрутизатора

Создание интерфейса в основном маршрутизаторе. Дальнейшее его конфигурирование будет происходить в виртуальном маршрутизаторе.

```
ecorouter(config)#interface e2
ecorouter(config-int)#exit
```

Создание виртуального маршрутизатора с именем VR10 в режиме конфигурирования основного маршрутизатора.

```
ecorouter(config)#virtual-router VR10
```

Добавление в виртуальный маршрутизатор протокола BGP.

```
ecorouter(config-vr)#load bgp
ecorouter(config-vr)#exit
```

Передача интерфейса в виртуальный маршрутизатор.

```
ecorouter(config-vr) #bind e2
```

Также интерфейс может быть передан в виртуальный маршрутизатор командой режима конфигурации интерфейса **virtual-router-forwarding <VR_NAME>**.

Для сохранения конфигурации виртуального маршрутизатора необходимо создать файл. Команда **configuration file <имя файла>** выполняется в режиме конфигурации основного маршрутизатора, в контексте конфигурации виртуального маршрутизатора.

```
ecorouter(config-vr) #configuration file VR10
```

Дальнейшая настройка интерфейсов (задание IP-адреса, описание, включение в протокол маршрутизации, административное управление) и маршрутизации виртуального устройства осуществляется в CLI виртуального маршрутизатора.

```
ecorouter#login virtual-router VR10
```

```
EcoRouterOS version 3.2.0 EcoRouter 07/06/16 15:53:00  
ecorouter>enable
```

Просмотр подробных настроек виртуального маршрутизатора осуществляется из виртуального маршрутизатора командой административного режима **show running-config**.

```
VR10#show running-config  
!  
no service password-encryption  
!  
hostname VR10  
!  
logging monitor 7  
!  
mpls propagate-ttl  
!  
line con 0  
login  
line vty 0 802  
login  
!  
interface e2  
ip mtu 1500  
ip address 1.1.1.1/24  
!  
end
```

8.3 Команды просмотра

Для вывода информации о созданных в системе виртуальных маршрутизаторах и загруженных в них протоколах используется команда административного режима **show virtual-router**.

```
ecorouter#show virtual-router  
Virtual Router VR10  
VR ID: 1  
Router ID: 1.1.1.1
```

Loaded Protocols: bgp

Также можно посмотреть в выводе команды административного режима **show running-config** секции, относящиеся к виртуальным маршрутизаторам и привязанным к ним интерфейсам.

```
ecorouter#show running-config
!
...
!
virtual-router VR10
configuration file VR10
load bgp
!
...
!
interface e2
ip mtu 1500
connect port tel service-instance 100
virtual-router-forwarding VR10
ip access-group 001 in
!
```

9 Маршрутизация

9.1 Введение в маршрутизацию

Доступность IP-подсетей, получение информации об IP-подсетях от смежных устройств, анонсирование маршрутной информации, выбор наилучшего маршрута, корректное реагирование на изменение топологии сети в операционной системе EcoRouterOS поддерживается за счет статической маршрутизации и динамических протоколов маршрутизации.

Маршрутизатор EcoRouter работает как с протоколами, разработанными для использования внутри одной автономной системы (RIPv2, OSPFv2, IS-IS), так и предназначенными для работы между ними (MP-BGP,), поддерживая при этом и статическую маршрутизацию.

В документации можно найти подробные инструкции по настройке для каждого протокола. Значения по умолчанию административных расстояний указаны в таблице ниже.

Таблица 42

Тип маршрута	Административное расстояние
Connected	0
Static	1
eBGP	20
OSPF	110
IS-IS	115
RIP	120
iBGP	200
Unreachable	255

9.2 Настройка статических маршрутов

Статический маршрут – постоянный маршрут в сеть назначения, установленный администратором сети вручную.

Статические маршруты используются в различных сценариях. Основная область применения – участки сети с простым дизайном и ожидаемым поведением сетевого трафика. Стандартный вариант использования - это отсутствие динамического маршрута в сеть назначения или необходимость переписать маршрут, полученный с помощью динамического протокола маршрутизации. Статические маршруты используют меньшую полосу пропускания, чем динамические протоколы маршрутизации, и не требуют процессорного времени для вычисления и анализа маршрутных обновлений.

Статические маршруты задаются в режиме конфигурации командой **ip route (ip-prefix | ip-addr ip-mask) (ip-gateway | interface) (<0-255>) (description <description>) (tag <0-4294967295>)**, где (0-255) – это значение административной дистанции.

9.2.1 Базовая настройка статических маршрутов

```
ecorouter>en  
ecorouter#conf t
```

Настройка происходит в режиме конфигурации.

```
Enter configuration commands, one per line. End with CNTL/Z.  
ecorouter(config)#ip route 192.168.1.0 255.255.255.0 172.16.10.1
```

Эта запись будет аналогична записи следующего вида:

```
ecorouter(config)#ip route 192.168.1.0/24 172.16.10.1
```

В данном виде записи сеть назначения описывается с помощью префикса.

```
ecorouter(config-if)#ip route 192.168.1.0/24 e1
```

В данном виде записи вместо адреса шлюза используется указание на интерфейс, где доступен адрес шлюза.

9.2.2 Административная дистанция статических маршрутов

По умолчанию, статический маршрут имеет административную дистанцию равную 1, что дает данному типу маршрутов больший приоритет перед всеми протоколами динамической маршрутизации.

Значение административной дистанции может быть изменено с помощью указания нужного значения в конце строки конфигурации статического маршрута.

```
ecorouter(config)#ip route 192.168.1.0 255.255.255.0 172.16.10.1 125
```

Пример использования:

Если есть динамические маршруты с административной метрикой 120 и нужно, чтобы они имели больший приоритет перед статическим маршрутом, то есть использовались маршрутизатором, то нужно указать значение административной дистанции статического маршрута больше 120.

9.3 Настройка RIP

Routing Information Protocol (RIP) – протокол динамической маршрутизации. Характеризуется тем, что устройства под управлением этого протокола отправляют сообщения с известными им маршрутами через определенные фиксированные интервалы и когда происходят изменения топологии. В сообщениях о маршрутных обновлениях также содержится значения метрики для каждой известной маршрутизатору сети.

В EcoRouterOS поддерживается RIP версии 2.

9.3.1 Метрика RIP

Для вычисления метрики RIP использует алгоритм Беллмана-Форда для поиска кратчайшего пути до сети назначения. При расчёте метрики данный алгоритм не учитывает загруженность канала и пропускную способность интерфейсов на пути до сети назначения. Результатом вычисления метрики будет количество «переходов» - маршрутизаторов, через которое сеть будет доступна. Лучшим маршрутом, который будет помещен в таблицу маршрутизации, будет считаться маршрут с минимальным возможным значением метрики.

Административная дистанция протокола по умолчанию равна 120.

Обновления маршрутной информации рассылаются на multicast адрес 224.0.0.9. Его слушают все маршрутизаторы под управлением RIP версии 2.

9.3.2 Таймеры RIP

По умолчанию маршрутизатор под управлением протокола RIP рассыпает пакеты с обновлением маршрутной информации каждые 30 секунд (update timer) с небольшим времененным отклонением. Маршрут помечается недостижимым (invalid, метрика 16), если в течение 6 интервалов по 30 секунд (invalid timer) маршрутизатор не получил обновление маршрутной информации. Через время, заданное flush timer, недостижимый маршрут удаляется из таблицы маршрутизации. Значение flush timer по умолчанию составляет 60 секунд, которые отсчитываются с момента назначения маршрута недостижимым.

Таким образом, когда информация о маршруте недоступна, то максимальное время нахождения такого маршрута в таблице маршрутизации равно 240 с.

Допустимые значения и значения по умолчанию для таймеров приведены в таблице:

Таблица 43

Таймер	Диапазон значений, с	Значение по умолчанию, с
update	1-4294967295	30
flush	1-4294967295	60
invalid	1-4294967295	180

Внимание: Настройка таймеров приводит к перезапуску RIP-сервиса, соответственно, это может вызвать прерывание передачи данных в сети.

9.3.3 Split horizon

Для предотвращения образования маршрутных петель в EcoRouterOS используется технология Split horizon. Технология заключается в том, что маршрутизатор не будет распространять информацию о маршруте через интерфейс, который является источником данной информации. Использование метода расщепления горизонта основано на том, что нет необходимости в отправке информации о маршруте в том направлении, по которому этот маршрут поступил.

9.3.4 Функция ручной суммаризации маршрутов

EcoRouterOS поддерживает функцию ручной суммаризации маршрутов RIP. Ручная суммаризация маршрутов работает следующим образом:

- суммаризация настраивается на интерфейсе маршрутизатора;
- настроенный суммарный маршрут анонсируется на интерфейсе в случае, если на маршрутизаторе есть хотя бы один RIP-маршрут, входящий в диапазон суммарного маршрута (дочерний маршрут);
- метрика суммарного маршрута равна наименьшей метрике среди дочерних маршрутов.

9.3.5 Команды настройки

Команды настройки протокола RIP представлены в таблице ниже.

Таблица 44

Команда	Описание
router rip	Включение протокола на устройстве

Команда	Описание
redistribute <connected static ospf isis bgp> metric <0-16>	Помещение маршрутов полученных в других протоколах маршрутизации в контекст маршрутизации RIP с указанием метрики для маршрута. По умолчанию метрика для таких маршрутов равна 0
neighbor <A.B.C.D> distribute-list <1-199 1300-2699> <in out>	Фильтрация маршрутов, отдаваемых или получаемых от соседа
distance <1-255>	Задание административной дистанции для получаемых протоколом маршрутов от других маршрутизаторов под управлением RIP
load rip	Включение протокола в виртуальном маршрутизаторе
default-information originate metric <0-16>	Включение анонса о маршруте по умолчанию в обновление протокола маршрутизации
network <A.B.C.D/M>	Анонс подсети в контексте маршрутизации RIP
passive-interface <имя интерфейса>	Команда выключает рассылку маршрутных обновлений RIP на интерфейсе
timer update <1-4294967295>	Настройка таймера update
timer invalid <1-4294967295>	Настройка таймера invalid
timer flush <1-4294967295>	Настройка таймера flush
ip summary-address rip <A.B.C.D> <mask>	Включение суммаризации маршрутов на интерфейсе. Команда вводится в режиме настройки интерфейса config-if

Все сети, объявленные на интерфейсах, будут помещены в контекст маршрутизации.

9.3.6 Пример базовой настройки

Шаг 1. Настройка интерфейсов.

```
ecorouter#conf t
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#interface e1
ecorouter(config-if)#ip add 10.10.10.1/24
ecorouter(config-if)#interface e2
ecorouter(config-if)#ip add 192.168.1.1/24
ecorouter(config-if)#interface loopback.1
ecorouter(config-lo)#ip add 1.1.1.1/32
```

Интерфейсы должны быть присоединены к портам с помощью сервисных интерфейсов.

Шаг 2. Включение протокола маршрутизации RIP.

```
ecorouter(config)#router rip
ecorouter(config-router) #
```

Шаг 3. Помещение присоединенных сетей в контекст маршрутизации RIP.

```
ecorouter(config-router) #network 10.10.10.0/24
ecorouter(config-router) #network 192.168.1.0/24
ecorouter(config-router) #network 1.1.1.1/32
```

Шаг 4. Помещение присоединенных сетей в контекст маршрутизации с желаемой метрикой.

```
ecorouter(config-router) #redistribute connected metric 1
ecorouter#sh ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
```

```
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter
area
* - candidate default

IP Route Table for VRF "default"
C 1.1.1.1/32 is directly connected, loopback.1
C 10.10.10.0/24 is directly connected, e1
C 192.168.1.0/24 is directly connected, e2
```

9.3.7 Включение протокола в виртуальном маршрутизаторе

Включение производится в режиме конфигурации физического маршрутизатора.

```
ecorouter>enable
```

```
ecorouter#configure terminal
```

Создание виртуального маршрутизатора с именем vr1.

```
ecorouter(config)#virtual-router vr1
```

Включение протокола в виртуальном маршрутизаторе.

```
ecorouter(config-vr)#load rip
```

9.3.8 Команды просмотра

Для диагностики работы протокола используется команда **show ip protocols rip**.

```
ecorouter#show ip protocols rip
Routing Protocol is "rip"
  Redistributing: default connected static
  Default version control: send version 2, receive version 2
  Interface e1: State is Up, Metric 1
    Sending updates every 30 seconds, next in 1 seconds
    Invalid after 180 seconds, flushed after 120
  Neighbors active: 1
    Neighbor IP address Metric Routes Seen
      10.0.0.2 1 1 29
  Interface e2: State is Up, Metric 1
    Sending updates every 30 seconds, next in 15 seconds
    Invalid after 180 seconds, flushed after 120
  Neighbors active: 0
  Maximum path: 16
  Routing Information:
    #0: 10.2.2.0/24 valid via 10.0.0.2 dev e1 from 10.0.0.2 metric 2 age 73
      seco
  Distance: (default is 120)
```

9.4 Настройка OSPF

Конфигурирование протокола OSPF состоит из нескольких обязательных этапов и множества необязательных. После того как был выбран дизайн OSPF-сети, а это очень непростая задача в сложных топологиях, конфигурирование в простейшем случае сводится к включению протокола OSPF в маршрутизаторах и размещению интерфейсов в нужных зонах.

Этапы конфигурирования:

Этап 1.

Перейдите в режим конфигурирования протокола с помощью команды **router ospf <номер процесса>**, где номер в пределах <0-65535> в режиме глобальной конфигурации.

Этап 2.

Сконфигурируйте OSPF идентификатор маршрутизатора (необязательный этап). Используйте команду **ospf router-id <значение>**, значение в виде IPv4 адреса или задайте IP-адрес для loopback интерфейса.

Этап 3.

В режиме конфигурирования протокола OSPF укажите одну или более команд **network <IP-адрес> <инверсная маска> area <идентификатор зоны>**, параметры которых соответствуют настройкам интерфейсов. Для исключения интерфейсов из процесса OSPF используйте команду **passive-interface <имя интерфейса>**.

Этап 4. (Необязательный этап)

Если тип сети не поддерживает мультикастную адресную рассылку, то необходимо будет указать соседей вручную.

Тип сети задается в режиме конфигурирования интерфейса командой **ip ospf network**. Укажите соседей вручную в режиме конфигурирования протокола с помощью команды **neighbor**.

Этап 5. (Необязательный этап)

Измените таймеры в режиме конфигурирования интерфейса с помощью команд **ip ospf dead-interval** и **ip ospf hello- interval**.

Этап 6. (Необязательный этап)

Настройте вручную стоимости интерфейсов, если нужно повлиять на выбор оптимального маршрута: укажите значение в режиме конфигурирования интерфейсов командой **ip ospf cost <значение>**. Для изменения множителя в формуле расчета стоимости маршрута по полосе пропускания интерфейсов используйте команду режима конфигурирования протокола OSPF **auto-cost reference-bandwidth**.

Этап 7. (Необязательный этап)

Сконфигурируйте аутентификацию протокола OSPF: на отдельных интерфейсах с помощью команды **ip ospf authentication** или для всех интерфейсов в определенной зоне в режиме конфигурирования протокола маршрутизации с помощью команды **area authentication**.

9.4.1 Пример настройки

Схема конфигурирования многозонового дизайна для OSPF-топологии показана на рисунке ниже:

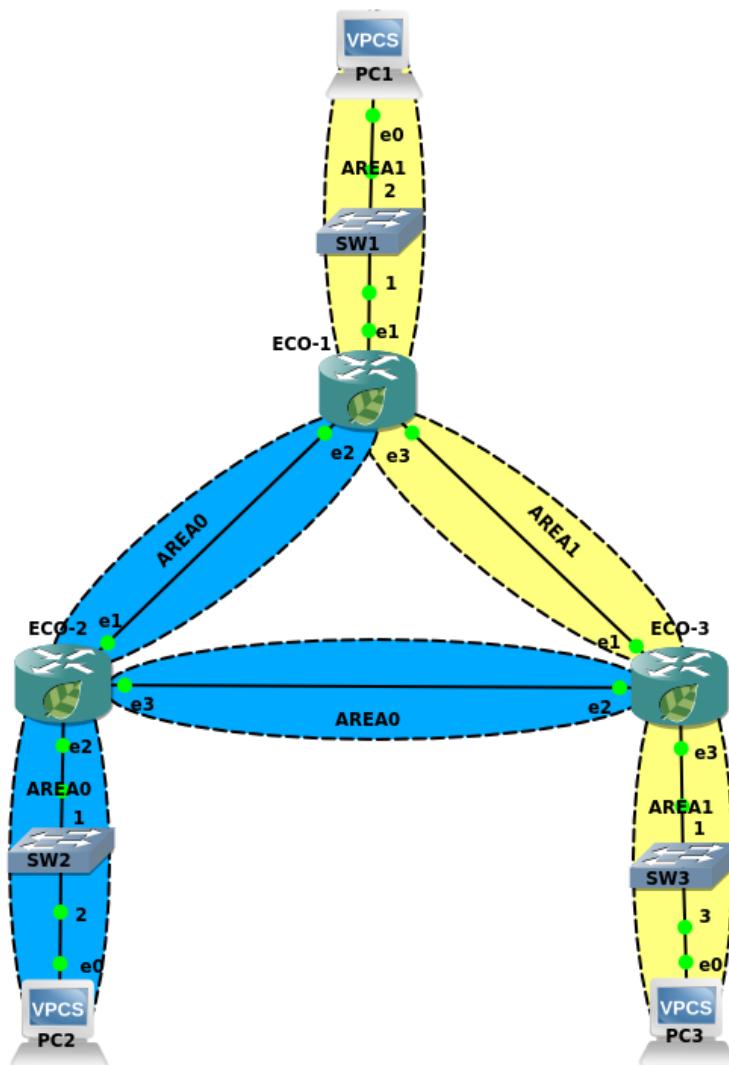


Рисунок 14

Пример конфигураций маршрутизаторов

ECO-1

Шаг 1. Задание имени устройства.

```
(config)#hostname ECO-1
```

Шаг 2. Настройка портов, интерфейсов и сервисных интерфейсов.

```
(config)#interface e1
(config-if)#ip address 10.10.0.1/16
(config)#interface e2
(config-if)#ip address 10.12.0.1/16
(config)#interface e3
(config-if)#ip address 10.13.0.1/16
(config)#port ge1
(config-port)#service-instance ge1/e1
(config-service-instance)#encapsulation untagged
(config-service-instance)#connect ip interface e1
(config)#port ge2
(config-port)#service-instance ge2/e2
```

```
(config-service-instance) #encapsulation untagged  
(config-service-instance) #connect ip interface e2  
(config) #port ge3  
(config-port) #service-instance ge3/e3  
(config-service-instance) #encapsulation untagged  
(config-service-instance) #connect ip interface e3
```

Шаг 3. Включение маршрутизации и объявление присоединенных сетей.

```
(config) #router ospf 1  
(config-router) #network 10.10.0.1 0.0.0.0 area 1  
(config-router) #network 10.12.0.1 0.0.0.0 area 0  
(config-router) #network 10.13.0.1 0.0.0.0 area 1
```

Конфигурация оставшихся маршрутизаторов будет аналогичной.

```
hostname ECO-2  
interface e1  
ip address 10.12.0.2/16  
interface e2  
ip address 10.20.0.2/16  
interface e3  
ip address 10.23.0.2/16  
port gel  
service-instance gel/e1  
encapsulation untagged  
connect ip interface e1  
port ge2  
service-instance ge2/e2  
encapsulation untagged  
connect ip interface e2  
port ge2  
service-instance ge2/e2  
encapsulation untagged  
connect ip interface e2  
router ospf 2  
network 10.12.0.2 0.0.0.0 area 0  
network 10.20.0.2 0.0.0.0 area 0  
network 10.23.0.2 0.0.0.0 area 0  
hostname ECO-3  
interface e1  
ip address 10.13.0.3/16  
interface e2  
ip address 10.23.0.3/16  
interface e3  
ip address 10.30.0.3/16  
port gel  
service-instance gel/e1  
encapsulation untagged  
connect ip interface e1  
port ge2  
service-instance ge2/e2  
encapsulation untagged  
connect ip interface e2  
port ge2  
service-instance ge2/e2
```

```
encapsulation untagged
connect ip interface e2
router ospf 2
network 10.13.0.3 0.0.0.0 area 1
network 10.23.0.3 0.0.0.0 area 0
network 10.30.0.3 0.0.0.0 area 1
```

9.4.2 Аутентификация

В OSPFv2 предусмотрена возможность настройки аутентификации между соседями. Для её включения необходимо создать authentication-key в режиме настройки интерфейса, а также включить поддержку аутентификации либо на интерфейсе, либо глобально внутри процесса ospf для всей area. Также при создании authentication-key необходимо выбрать, в каком виде ключ будет передаваться между соседями: в открытом виде или в виде md5 хеша.

Команды конфигурирования:

Таблица 45

Команда	Режим	Описание
ip ospf authentication [message-digest / null]	(config-if) #	Включение режима аутентификации на интерфейсе
ip ospf authentication-key	(config-if) #	Задание plain-text ключа
ip ospf message-digest-key <key id> md5 <key>	(config-if) #	Задание ключа и использование хеша md5
area 0 authentication [message-digest]	(config-router) #	Включение аутентификации на всех интерфейсах зоны ospf

Рассмотрим различные примеры настроек аутентификации в приведенной выше топологии:

Настройка plain-text аутентификации между маршрутизаторами ECO-1 и ECO-2 с ключом “ecorouter”.

```
ECO-1
(config) #interface e2
(config-if) #ip ospf authentication
(config-if) #ip ospf authentication-key ecorouter
```

На маршрутизаторе ECO-2 должны быть аналогичные настройки, за исключением номера интерфейса.

Настройка plain-text аутентификации между маршрутизаторами ECO-2 и ECO-3 с ключом “ecorouter” и включением из режима конфигурации.

```
ECO-2
(config) #router ospf 1
(config-router) #area 0 authentication
(config-router) #exit
(config) #interface e3
(config-if) #ip ospf authentication-key ecorouter
```

В данном примере режим аутентификации будет применен ко всем интерфейсам внутри зоны 0 (e1, e2, e3). Настройка маршрутизатора ECO-3 будет аналогичной, за исключением номера интерфейса.

Настройка md5 аутентификации между маршрутизаторами ECO-1 и ECO-3 с ключом "ecorouter".

```
ECO-1
(config) #interface e3
(config-if) #ip ospf authentication message-digest
(config-if) #ip ospf message-digest-key 1 md5 ecorouter
```

На маршрутизаторе ECO-3 должны быть аналогичные настройки, за исключением номера интерфейса.

Настройка md5 аутентификации между маршрутизаторами ECO-1 и ECO-3 с ключом "ecorouter" и включением из режима конфигурации.

```
ECO-1
(config) #interface e3
(config-router) #area 1 authentication message-digest
(config-router) #exit
(config) #interface e3
(config-if) #ip ospf message-digest-key 1 md5 ecorouter
```

На маршрутизаторе ECO-3 должны быть аналогичные настройки, за исключением номера интерфейса.

9.4.3 Фильтрация и суммаризация маршрутов OSPF

Внутренняя логика работы OSPF позволяет осуществлять фильтрацию и суммаризацию только на ABR и ASBR маршрутизаторах домена. Фильтрацию можно осуществлять с помощью filter-list и distribute-list, которые в своей работе полагаются на prefix-list или policy-filter-list. Пример использования filter-list показан на рисунке ниже:

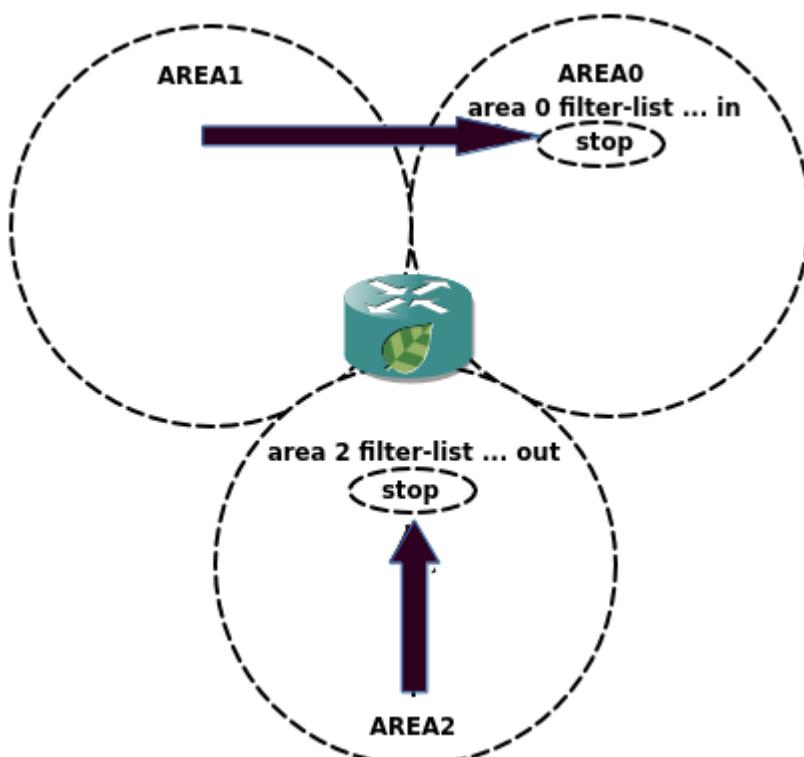


Рисунок 15

Для того чтобы отфильтровать маршруты из области 1 и области 2, на ABR в режиме конфигурирования маршрутизации OSPF следует использовать команду **area 0 filter-list <номер prefix-list/policy-filter-list> in**. Для того чтобы отфильтровать маршруты из области 2, на ABR следует использовать команду **area 2 filter-list <номер prefix-list/policy-filter-list> out**, где **prefix-list** и **policy-filter-list** соответствуют определенным подсетям. Подробнее об этих списках читайте в соответствующих разделах.

EcoRouterOS позволяет фильтровать маршруты и с помощью **distribute-list**. Внимание: при этом информация о маршруте будет содержаться в базе топологии OSPF, а в таблице маршрутизации нет, что может привести к увеличению времени поиска и обнаружения проблем в сети. Для фильтрации используйте команду **distribute-list <номер policy-filter-list> in**.

Суммаризация в OSPF возможна как на ABR, так и на ASBR. Команды для разных типов маршрутизаторов в домене также различны.

На ABR используется команда **area <area-id> range <ip-address/mask> [advertise | not-advertise]**, где параметр **advertize** стоит по умолчанию, параметр **not-advertise** отключает анонсирование суммарного маршрута.

На ASBR команда выглядит следующим образом: **summary-address <ip-address/mask> [tag] [not-advertise]**, как видно есть возможность пометить маршрут тегом с помощью ключевого слова **tag** и отфильтровать маршрут.

По умолчанию, при суммаризации используется наибольшая метрика из всего набора метрик для внутренних маршрутов. Для изменения этого поведения можно воспользоваться командой **compatible rfc1583** в режиме конфигурации маршрутизации, тогда будет выбираться наименьшая.

9.4.4 Маршрут по умолчанию

Для настройки маршрута по умолчанию в режиме конфигурирования роутера используется команда **default-information originate [always] [metric <значение>] [metric-type 1 | metric-type 2] [route-map <имя>]**

После ввода команды конфигурируемый маршрутизатор начинает рекламировать себя в качестве дефолтного (если маршрут по умолчанию есть в таблице маршрутизации самого маршрутизатора).

Если неизвестно, присутствует ли маршрут по умолчанию в таблице маршрутизации выбранного маршрутизатора, при вводе команды следует указать параметр **always**. Таким образом отменяется обязательность выполнения этого условия.

Параметр **metric** задает значение метрики, параметр **metric-type** указывает тип метрики OSPF, параметр **route-map** ссылается на условия в карте маршрутов. Важно помнить, что маршрут по умолчанию будет рекламироваться в виде LSA type 5.

9.4.5 Зоны OSPF

При правильном дизайне OSPF сети для уменьшения размера базы данных топологии может потребоваться использование тупиковых зон OSPF. EcoRouterOS поддерживает эту функциональность.

Таблица 46

Тип области	ABR передает LSA type 5 в область?	ABR передает LSA type 3 в область?	Позволена редистрибуция в тупиковую зону?	Команда конфигурирования
Stubby	Нет	Да	Нет	area <номер> stub
Totally stubby	Нет	Нет	Нет	area <номер> stub no-summary
NSSA	Нет	Да	Да	area <номер> nssa
Totally NSSA	Нет	Нет	Да	area <номер> nssa no-summary

9.4.6 Редистрибуция OSPF

Редистрибуция из различных протоколов маршрутизации, статических и непосредственно подключенных маршрутов в OSPF может быть настроена в режиме конфигурирования роутера с помощью команд: **redistribute <bgp/ospf/isis/rip/connected/static> [metric <значение>] [metric-type 1 | metric-type 2] [route-map <имя>] [tag]**, где параметр **metric** задает значение метрики, параметр **metric-type** указывает тип метрики OSPF, параметр **route-map** ссылается на условия в карте маршрутов, **tag** – тегирует редистрибутированные сети. С помощью команды **default-metric** можно задать значение для всех редистрибутированных маршрутов. Команда **distance** задает значение административной дистанции для протокола OSPF.

9.4.7 Виртуальные линки и Multi-Area соседства

Виртуальный линк следует создавать с осторожностью, так как его использование на постоянной основе может вызвать сложности в администрировании при дальнейшем росте OSPF-топологии. Если же выбора не остается, то для конфигурации виртуального линка используйте в режиме конфигурирования роутера команду **area <номер> virtual-link <ip-адрес>**, где **номер area** - это область, через которую создается виртуальный линк, **ip-адрес** - адрес соседа. Дальнейшие опции команды помогут настроить тайминг в линке и аутентификацию.

Для решения задач маршрутизации может возникнуть необходимость создания multi-area. EcoRouterOS поддерживает эту функциональность. Для создания multi-area введите команду **area <номер> multi-area-adjacency <имя интерфейса> neighbor <IP-адрес>**, где номер area соответствует области, для которой настраивается маршрутизация, имя интерфейса соответствует имени выходного интерфейса в направлении соседа. Обратите внимание, команда требует указания адреса соседа.

9.4.8 Команды просмотра OSPF

Таблица 47

Команда	Описание
show ip route ospf	Просмотр маршрутов в таблице маршрутизации полученных через OSPF
show ip ospf neighbor	Просмотр сведений о соседских отношениях между OSPF маршрутизаторами
show ip ospf interface	Просмотр данных о состоянии и сконфигурированных настройках на интерфейсах, участвующих в OSPF процессе
show ip protocols	Просмотр информации о запущенных процессах маршрутизации

Команда	Описание
show ip ospf database	Просмотр базы данных OSPF топологии
show ip ospf virtual-links	Просмотр информации о OSPF виртуальном линке
show ip ospf border-routers	Просмотр информации о пограничных маршрутизаторах
show ip ospf multi-area-adjacencies	Просмотр информации о multi-area соседях
show ip ospf	Просмотр сведений о OSPF процессах запущенных на маршрутизаторе

9.4.9 Дополнительные команды конфигурирования OSPF

Таблица 48

Команда	Режим	Описание
capability restart graceful	(config)#	Включение функционала мягкого перезапуска (graceful restart)
max-concurrent-dd <1-65535>	(config)#	Количество одновременно обработанных дескрипторов БД (DD)
maximum-area <1-4294967294>	(config)#	Максимально возможное количество областей
ospf flood-reduction	(config)#	Уменьшение сигнальной нагрузки путем установки DNA бита
overflow database	(config)#	Уменьшение максимального количества объявлений о состоянии канала (LSA), которые могут быть обработаны
timers lsa arrival <0-600000>	(config)#	Установка минимального интервала приема того же LSA от соседа
ip ospf database-filter all out	(config-int)#	Отключение рассылки LSA через интерфейс
ip ospf disable all	(config-int)#	Отключение OSPF функционала
ip ospf flood-reduction	(config-int)#	Уменьшение сигнальной нагрузки путем установки DNA бита
ip ospf mtu <576-65535>	(config-int)#	Установка MTU для OSPF пакетов
ip ospf mtu-ignore	(config-int)#	Отключение проверки MTU в DD сообщениях
ip ospf priority <0-255>	(config-int)#	Установка OSPF приоритета
ip ospf retransmit-interval <1-65535>	(config-int)#	Установка временного интервала для рассылки LSA соседям
ip ospf transmit-delay <1-3600>	(config-int)#	Установка приблизительного времени передачи LSU через интерфейс

9.4.10 Команды перезапуска процесса маршрутизации

Для перезапуска процесса маршрутизации OSPF используется команда **clear ip ospf process** или **clear ip ospf <номер процесса> process**. Команда выполняется из режима администрирования.

9.4.11 Loop-Free Alternate (LFA) в OSPF

Для быстрого переключения с основного маршрута на резервный в протоколе OSPF используется технология LFA (Loop-Free Alternate).

При включении данной опции создается новая таблица с резервными, надежными маршрутами для быстрого переключения маршрутов (fast-reroute). Под надежностью маршрута здесь понимается защищенность его от петель.

Если маршрутизатор детектирует падение локального линка, по которому строился основной маршрут, то в FIB моментально отправляется заранее выбранный резервный маршрут.

Пересчет дерева по алгоритму SPF осуществляется независимо от переключения на резервный маршрут и может происходить как во время переключений, так и после.

Для того чтобы резервный маршрут был добавлен в таблицу быстрого переключения маршрутов, необходимо и достаточно выполнения следующего условия:

$$D(N,D) < D(N,S) + D(S,D)$$

где:

$D(x,y)$ - расстояние между x и y , выраженное в ospf-метрике;

N - соседний маршрутизатор, через который ищется резервный путь;

D - маршрут назначения;

S - источник.

Резервный маршрут может быть только один. Когда на роль резервного маршрута есть несколько претендентов, то работают следующие правила:

1. Выигрывает маршрут с наименьшей метрикой.
2. Если метрики равны, то выбирается маршрут с наименьшим адресом соседнего маршрутизатора.

Изменить эти правила нельзя.

Если в основной таблице маршрутизации RIB находятся два активных маршрута, т.е. работает ECMP, то таблица для маршрутов быстрого переключения будет пуста.

Резервный маршрут рассчитывается для каждого основного маршрута отдельно (per-prefix LFA). В случае для ECMP альтернативным для каждого основного маршрута будет второй активный маршрут. Поскольку эти маршруты и так находятся в основной таблице маршрутизации, то нет необходимости их помещать в таблицу для быстрого переключения маршрутов.

Для включения данной технологии используется команда **fast-reroute keep-all-paths** в режиме конфигурации протокола OSPF.

Для отключения технологии на конкретных интерфейсах используется команда **ip ospf fast-reroute per-prefix candidate disable**.

Просмотреть потенциальные резервные маршруты можно с помощью команды **show ip route fast-reroute**. Вывод данной команды аналогичен формату вывода команды **show ip route**.

Данный функционал доступен также при работе в VRF. Команда для просмотра - **show ip route vrf <NAME> fast-reroute**, где <NAME> - имя VRF.

9.5 Настройка IS-IS

IS-IS (Intermediate System to Intermediate System) – внутренний протокол динамической маршрутизации.

Конфигурирование протокола IS-IS состоит из нескольких этапов. После того как был выбран дизайн IS-IS сети, конфигурирование в простейшем случае сводится к запуску протокола IS-IS в маршрутизаторах, настройке уникального NET-адреса и включению протокола на интерфейсах.

Этапы конфигурирования:

Этап 1.

Перейдите в режим конфигурирования протокола с помощью команды **router isis <имя процесса>**, где имя экземпляра может состоять из букв и цифр или вовсе отсутствовать.

Этап 2.

Сконфигурируйте NET-адрес маршрутизатора, используя команду **net <адрес>**, где адрес может иметь размер от 8 до 20 байт, последний байт всегда n-селектор (SEL) и должен быть равен 0. 6 байт перед n-селектором является системным идентификатором (System-ID), байты перед системным идентификатором (1-13) являются идентификатором области (area ID). По умолчанию на маршрутизаторе можно задать три NET-адреса в различных областях, но системный идентификатор должен быть одинаков. Повысить количество задаваемых NET-адресов можно, используя команду: **max-area-address <значение>**.

Этап 3.

В режиме конфигурирования протокола IS-IS укажите уровень, на котором будет работать маршрутизатор, командой **is-type <level-1/level-1-2/level-2-only>**, по умолчанию уровень L1/L2. Также можно указать тип соединения на интерфейсе командой **isis circuit-type <level-1/level-1-2/level-2-only>**, по умолчанию L1/L2.

Этап 4.

Задайте тип сети в режиме конфигурирования интерфейса с помощью команды **isis network**. Тип сети может быть broadcast или point-to-point.

Этап 5.

Задайте значения таймеров в режиме конфигурирования интерфейсов с помощью команд **isis hello-interval** или с помощью задания множителя для расчета **hold-timer** с помощью команды **isis hello-multiplier <значение>**.

Этап 6.

Настройте вручную стоимости интерфейсов, если необходимо повлиять на выбор оптимального маршрута. Для этого в режиме конфигурирования интерфейсов укажите значение командой **isis metric <значение>**.

Этап 7.

Аутентификация протокола IS-IS. EcoRouterOS поддерживает clear-text и md5 аутентификацию с помощью цепочек ключей.

Настройте аутентификацию на каждом интерфейсе в отдельности. Для clear-text аутентификации в режиме конфигурирования интерфейса используйте команду **isis password <слово> [level-1/level-2]**, где слово представляет собой набор не более чем 254 символов. Для конфигурирования md5 аутентификации используйте команды **isis authentication mode md5** и **isis authentication key-chain <название цепочки> [level-1/level-2]**. Название цепочки задается через отдельный режим конфигурирования цепочек ключей с помощью команды **key chain <название цепочки>**, в этом режиме позволено указывать несколько ключей и паролей.

9.5.1 Пример настройки

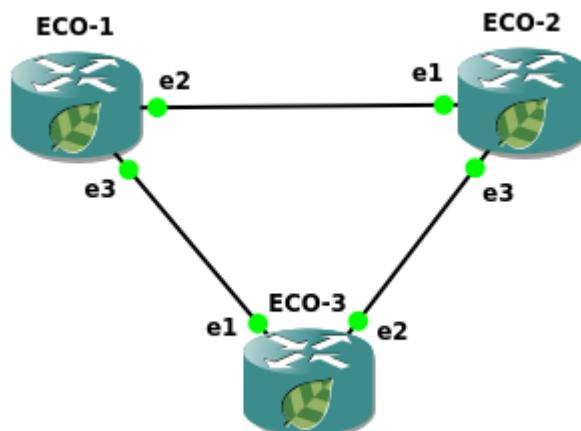


Рисунок 16

Шаг 1. Задание имени устройства.

```
ecorouter(config)#hostname ECO-1
```

Шаг 2. Настройка портов, интерфейсов и сервисных интерфейсов.

```
ecorouter(config)#interface e2
ecorouter(config-if)#ip address 10.12.0.1/16
ecorouter(config)#interface e3
ecorouter(config-if)#ip address 10.13.0.1/16
ecorouter(config)#port ge2
ecorouter(config-port)#service-instance ge2/e2
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface e2
ecorouter(config)#port ge3
ecorouter(config-port)#service-instance ge3/e3
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface e3
```

Шаг 3. Включение маршрутизации.

```

ecorouter(config)#router isis
ecorouter(config-router)#net 49.0001.0000.0000.0001.00
ecorouter(config-router)#exit
ecorouter(config)#interface e2
ecorouter(config-int)#ip router isis
ecorouter(config-int)#interface e3
ecorouter(config-int)#ip router isis
ecorouter(config-int)#exit

```

Шаг 4. Включение аутентификации между соседями.

```

ecorouter(config)#key chain test
ecorouter(config-keychain)#key 1
ecorouter(config-keychain-key)#key-string ecorouter
ecorouter(config-keychain-key)#exit
ecorouter(config-keychain)#exit
ecorouter(config)#interface e2
ecorouter(config-if)#isis authentication mode md5
ecorouter(config-if)#isis authentication key-chain test
ecorouter(config)#interface e3
ecorouter(config-if)#isis authentication mode md5
ecorouter(config-if)#isis authentication key-chain test

```

Конфигурация оставшихся маршрутизаторов будет аналогичной.

```

hostname ECO-2
key chain test2
key 2
key-string 0x8de456332b943f870ef377482f699e4c
interface e1
ip address 10.12.0.2/16
ip router isis
interface e3
ip address 10.23.0.2/16
ip router isis
port ge1
service-instance ge1/e1
encapsulation untagged
connect ip interface e1
port ge2
service-instance ge2/e2
encapsulation untagged
connect ip interface e2
router isis
net 49.0001.0000.0000.0002.00
hostname ECO-3
key chain test3
key 3
key-string 0x8de456332b943f870ef377482f699e4c
interface e1
ip address 10.13.0.3/16
ip router isis
interface e2
ip address 10.23.0.3/16
ip router isis
port ge1

```

```
service-instance ge1/e1
encapsulation untagged
connect ip interface e1
port ge2
service-instance ge2/e2
encapsulation untagged
connect ip interface e2
router isis
net 49.0001.0000.0000.0003.00
```

9.5.2 Редистрибуция, фильтрация и суммаризация маршрутов

Пользователь может запретить или разрешить передачу маршрутной информации о подсети при редистрибуции маршрутов из разных IS-IS уровней. Для этого можно сконфигурировать **policy-filter-list**, **route-map** с правилами **permit** или **deny** и применить к **distribute-list** (подробнее о листах и картах маршрутов читайте в соответствующих разделах). Команда конфигурирования: **redistribute isis <level-1/level-2> into <level-2/level-1> distribute-list <название>**.

Чтобы управлять передачей маршрутной информацией из другого протокола маршрутизации применяются только route-map. Команда конфигурирования: **redistribute <connected/static/rip/ospf/bgp> [metric <0-63>] [metric-type <internal/external>] [level-1/level-2/level-1-2] [route-map <название>]**.

Для суммаризации маршрутов используется команда: **summary-address <адрес/маска> [level-1/level-2/level-1-2] [metric <0-63>]**.

Для установки значения административной дистанции для IS-IS маршрутов можно воспользоваться командой **metric <значение> [systemID <номер policy-filter-list>]**, где **systemID** системный идентификатор соседа, от которого приходит реклама подсетей.

9.5.3 Маршруты по умолчанию и mesh-группы

Для уменьшения размера таблиц маршрутизации в IS-IS домене EcoRouterOS позволяет настраивать передачу маршрутов «по умолчанию» своим соседям. При подключении L1/L2 маршрутизатора к различным областям (area) в рекламе маршрутной информации к L1 соседу автоматически будет рассылаться дефолтный маршрут, где в качестве next-hop адреса будет указан адрес L1/L2 маршрутизатора. Для передачи маршрута «по умолчанию» в сторону L2 соседа можно воспользоваться командой **default-information originate [always] [route-map]**, где параметр **always** не учитывает наличия дефолтного маршрута в собственной таблице маршрутизации, а параметр **route-map** позволяет выделить конкретную подсеть.

Для контроля за LSP флюдингом в NBMA линках EcoRouterOS позволяет добавлять интерфейсы в разные mesh-группы, тем самым накладывая определенные правила на обработку пакетов с информацией о подсетях.

Команды конфигурирования в режиме интерфейса: **isis mesh-group <значение/blocked>**. Если LSP был принят на интерфейс, который не принадлежит mesh-группе, то он передается дальше обычным путем. Если LSP был принят на интерфейс, который принадлежит mesh-группе, то он передается во все интерфейсы, кроме тех, которые принадлежат той же группе, или указаны с параметром **blocked**.

9.5.4 Дополнительные команды конфигурирования

В таблице ниже приведены дополнительные команды конфигурирования протокола IS-IS.

Таблица 49

Команда	Режим	Описание
ignore-lsp-errors	(config-router)#	Игнорирование LSP с ошибками в контрольной сумме
ispf	(config-router)#	Включение инкрементального SPF
lsp-gen-interval	(config-router)#	Установка временного интервала регенерации LSP
lsp-mtu	(config-router)#	Размер MTU для LSP
lsp-refresh-interval	(config-router)#	Интервал обновления LSP
max-lsp-lifetime	(config-router)#	Время жизни LSP
passive-interface	(config-router)#	Задание пассивного интерфейса
prc-interval-exp	(config-router)#	Установка интервалов для PRC
restart-timer	(config-router)#	Установка сброса IS-IS таймера
set-overload-bit	(config-router)#	Установка overload бита
spf-interval-exp	(config-router)#	Установка интервалов для SPF
isis csnp-interval	(config-int)#	Установка CSNP интервала
isis hello padding	(config-int)#	Уменьшение размера Hello сообщений
isis lsp-interval	(config-int)#	Установка LSP интервала
isis priority	(config-int)#	Установка приоритета
isis retransmit-interval	(config-int)#	Установка временного интервала регенерации LSP
clear isis process	#	Сброс процесса маршрутизации

9.5.5 Команды просмотра

В таблице ниже приведены команды просмотра информации, относящиеся к протоколу. Как и другие команды **show**, они поддерживают использование модификаторов.

Таблица 50

Команда	Описание
show isis counter	Выводит количественную информацию о IS-IS сообщениях
show isis database	Выводит краткую информацию о содержимом в базе данных
show isis database detail	Выводит полную информацию о содержимом в базе данных
show isis interface	Выводит информацию о параметрах сконфигурированных на интерфейсах, включенных в процесс маршрутизации
show isis topology	Выводит информацию о содержимом в базе данных топологии

Команда	Описание
show clns neighbors	Выводит информацию о соседских отношениях
show clns protocol	Выводит общую информацию о протоколе

9.6 Настройка BGP

На сегодняшний день в качестве протокола маршрутизации, предназначенного для изучения, анонса и выбора лучшего маршрута в глобальной сети Интернет, используют Border Gateway Protocol (BGP). EcoRouterOS использует расширенную версию протокола Multiprotocol BGP (MP-BGP), что позволяет объединить различные типы адресаций (unicast, multicast) в рамках единой конфигурации и, в будущем, адресацию IPv6. Стоит заметить, что MP-BGP обратно совместим с традиционной четвертой версией протокола BGP, как результат, BGP-4 маршрутизатор может формировать соседские отношения с MP-BGP маршрутизатором и просто игнорировать любые принятые BGP сообщения, содержащие неизвестные расширения.

Приведем несколько основных концепций протокола и сравним их с логикой работы Internal Gateway Protocol (IGP) маршрутизации, в качестве примера будет выступать OSPF.

В таблице ниже приведено сравнение OSPF логики с BGP.

Таблица 51

OSPF	BGP
Для отправки маршрутной информации должны сформироваться соседские отношения между маршрутизаторами	Используется подобная логика
Соседи обнаруживаются при помощи мультикастовых сообщений в непосредственно подключенной подсети	Соседи указываются путем статической конфигурации и могут быть разных подсетях
Не используют TCP	Используется TCP соединение между соседями (порт 179)
Рекламирует prefix/length	Рекламирует prefix/length (Network Layer Reachability Information)
Рекламирует информацию о метрике	Рекламирует атрибуты пути
Приоритетна скорость переключения сети на самый эффективный и рациональный маршрут	Приоритетна масштабируемость, может выбираться не самый эффективный и рациональный маршрут

9.6.1 Базовая настройка BGP

Для обмена или получения маршрутной информации по BGP необходимо иметь заранее зарегистрированный номер автономной системы (ASN). Так же как и для открытых маршрутизуемых IP-адресов, процесс присвоения номеров регулируется ассоциацией IANA. При определенных случаях подключения к сети Интернет номера из частного диапазона автономных систем (AS) выделяются провайдером. EcoRouterOS позволяет указать номер AS в диапазоне <1-4294967295>.

В зависимости от принадлежности к локальной автономной системе или к соседней BGP определяет два класса соседств между маршрутизаторами: internal BGP (iBGP) и external BGP (eBGP) соответственно. Реализация протокола в нашем оборудовании дает возможность

гибкой настройки для обоих типов соседств. При настройке базовой конфигурации соседств можно выполнить следующие шаги:

для iBGP:

Шаг 1. Настройте IP адрес loopback интерфейса на каждом маршрутизаторе, используя команды:

```
interface loopback.<number>
ip address <address/mask>
```

Шаг 2. Запустите протокол BGP, указав нужную автономную систему, командой:

```
router bgp <number>
```

Шаг 3. Укажите BGP использовать loopback интерфейс в качестве источника, используя команду:

```
neighbor <neighbor-ip> update-source <interface-id>
```

Шаг 4. Сконфигурируйте BGP соседей на каждом маршрутизаторе, указав loopback адрес соседа и номер локальной AS, используя команду:

```
neighbor <neighbor-ip> remote-as <number>
```

Шаг 5. Убедитесь, что у каждого маршрутизатора есть маршрут до loopback адреса соседа.

```
show ip route bgp
```

для eBGP:

Шаг 1. Настройте IP адрес loopback интерфейса на каждом маршрутизаторе, используя команды:

```
interface loopback.<number>
ip address <address/mask>
```

Шаг 2. Запустите протокол BGP, указав нужную автономную систему, командой:

```
router bgp <number>
```

Шаг 3. Укажите BGP использовать loopback интерфейс в качестве источника, используя команду:

```
neighbor <neighbor-ip> update-source <interface-id>
```

Шаг 4. Сконфигурируйте BGP-соседей на каждом маршрутизаторе, указав loopback адрес соседа и номер удаленной AS, используя команду:

```
neighbor <neighbor-ip> remote-as <number>
```

Шаг 5. Убедитесь, что у каждого маршрутизатора есть маршрут до loopback адреса соседа.

```
show ip route bgp
```

Шаг 6. Сконфигурируйте eBGP multihop для увеличения значения TTL командой:

```
neighbor <neighbor-ip> ebgp-multihop <hops>
```

В данных примерах рассматривался один из способов теоретически верного (с точки зрения отказоустойчивости) конфигурирования при простейшей топологии.

9.6.2 BGP атрибуты

Для управления маршрутной информацией, маршрутами протекания трафика и, в целом, решения задач администрирования сети на основе BGP EcoRouterOS предлагает сетевым инженерам пользоваться атрибутами, представленными в таблице ниже.

Таблица 52

Атрибут	Описание	Направление трафика
Weight	Числовое значение в диапазоне от 0 до $2^{16}-1$, влияет на маршрут до префикса, переданного в сообщении update от соседа. Не рекламируется BGP соседям	Влияет на исходящий трафик
Local Preference	Числовое значение в диапазоне от 0 до $2^{32}-1$, рассылается маршрутизаторам внутри локальной AS и влияет на маршрут выхода из этой автономной системы	Влияет на исходящий трафик
AS-path (length)	Количество автономных систем. Чем меньше, тем лучше	Влияет на исходящий / входящий трафик
Origin	Показывает, каким образом маршрут был добавлен в рекламу BGP (I (IGP), E (EGP), or ? (incomplete information).)	Влияет на исходящий трафик
Multi-Exit Discriminator (MED)	Аналог метрики маршрута, числовое значение в диапазоне от 0 до $2^{32}-1$, влияет на выбор маршрута к локальной AS из другой автономной системы. Чем меньше, тем лучше	Влияет на входящий трафик

BGP-атрибуты предоставляют информацию для выбора лучшего маршрута, однако есть и такие, которые служат для других целей. Например, атрибут **Next Hop** предоставляет информацию о соседе. Для работы протокола в таблице маршрутизации должен присутствовать маршрут до этого адреса, но при этом атрибут никак не влияет на сам алгоритм выбора лучшего пути. Процесс выбора лучшего пути описан в таблице ниже. Параметры расположены в порядке убывания приоритета, начиная с наиболее предпочтаемых.

Таблица 53

Приоритет	Атрибут/свойство	Что лучше?
0	Next Hop	Если адрес недоступен, маршрутизатор не может использовать этот путь
1	Weight	Наибольшее значение
2	Local Preference	Наибольшее значение
3	Локальный маршрут (команды network/redistribution)	Локальный маршрут лучше, чем полученный через eBGP/iBGP
4	AS-path length	Наименьшее значение
5	Origin	Предпочтение I>E>?
6	MED	Наименьшее значение
7	iBGP или eBGP	Предпочтение eBGP>iBGP
8	IGP метрика до Next Hop	Наименьшее значение
9	Время жизни eBGP маршрута	Наибольшее значение
10	ID соседнего BGP-маршрутизатора	Минимальное значение
11	Длина списка кластера (cluster list) (для множественного пути)	Минимальное значение
12	IP адрес соседа	Минимальное значение

Приведем примеры конфигурационных команд для изменения значений атрибутов/свойств по умолчанию.

Команда для сохранения адреса Next Hop при iBGP соседстве (по умолчанию в iBGP адрес не передается) - **neighbor <address> next-hop-self**.

Установка значения Weight для соседа (значение по умолчанию 0 для маршрутов, полученных от соседей, 32768 для локально инжектированных маршрутов) - **neighbor <address> weight <value>**, значение может быть задано через **route-map** и применено командой **neighbor <address> route-map <name> in**.

Установка значения Local Preference (значение по умолчанию 100) - **bgp default local-preference <0-4294967295>**, значение может быть задано через **route-map** и применено командой **neighbor <address> route-map <name> in**.

9.6.3 Команды конфигурирования атрибутов через route-map

Их использование требует наличие команды **neighbor <address> soft-reconfiguration inbound** в конфигурации протокола.

Просмотр всех доступных атрибутов осуществляется на подуровне настройки BGP с помощью команды **set <атрибут>**.

```
ecorouter(config-route-map)#set ?
?corouter(config-route-map)#set
  aggregator          BGP aggregator attribute
  as-path              Prepend string for a BGP AS-path attribute
  atomic-aggregate    BGP atomic aggregate attribute
  comm-list            set BGP community list (for deletion)
  community            BGP community attribute
  dampening            Enable route-flap dampening
  extcommunity         BGP extended community attribute
  interface             Configure interface
  ip                   Internet Protocol (IP)
  level                IS-IS level to export route
  local-preference     BGP local preference path attribute
  metric               Metric value for destination routing protocol
  metric-type           Type of metric for destination routing protocol
  origin                BGP origin code
  originator-id        BGP originator ID attribute
  tag                  Tag value for destination routing protocol
  vpnv4                VPNv4 information
  weight               BGP weight for routing table
```

В таблице ниже описаны доступные для конфигурирования атрибуты.

Таблица 54

Атрибут	Описание
Aggregator	Указание на маршрутизатор, который сделал агрегацию маршрутов, соответственно, можно указать адрес маршрутизатора с указанием AS
AS-path	Указание на все AS, через который пролегает маршрут до сети назначения. С помощью set можно увеличить длину атрибута
Atomic-Aggregate	Атрибут используется при агрегировании маршрутов. Команда для суммирования маршрутов: aggregate-address <address> [summary-only] [as-set]

Атрибут	Описание
	<p>[summary-only] - ключ, который указывает передавать только суммарный маршрут (по умолчанию передаются все подсети вместе с суммарным маршрутом).</p> <p>[as-set] - ключ для объявления локальной AS.</p>
Community	<p>Атрибут позволяет выделить необходимые маршруты в логическую группу, чтобы в дальнейшем их специальным образом обработать (пустить их по другому маршруту, применить QoS политики).</p> <p>Установка значения через параметр set:</p> <pre>ecorouter(config-route-map)#set community ?<1-65535> community number AA:NN community number in aa:nn format additive Add to the existing community internet Internet (well-known community) local-AS Do not send outside local AS (well-known community) no-advertise Do not advertise to any peer (well-known community) no-export Do not export to next AS (well-known community) none No community attribute</pre> <p>Для дальнейшей рекламы маршрутов с атрибутом Community указывается команда:</p> <p>bgp config-type standart в режиме конфигурации, команда neighbor <address> send-community both добавится автоматически</p>
Comm-list	<p>Параметр позволяет задать список сообществ для удаления.</p> <p>EcoRouterOS позволяет создавать community-list для того, чтобы затем с помощью route-map обработать рекламу подсети (подробнее о route-map читайте в разделе «Карты маршрутов»). Пример настройки для установки метрики для маршрутов с community=100:</p> <pre>ip community-list 1 permit <numberAS:100>, numberAS - номер AS, которая прорекламировала маршрут route-map community permit 100 match community 1 set metric 777</pre> <p>Для дальнейшей рекламы маршрутов с атрибутом Community указывается команда:</p> <p>neighbor <address> send-community</p>
Dampening	<p>Дополнительная функциональность протокола BGP для защиты от нестабильности соединений (route flapping).</p> <p>Команда set dampening <1-45>, где <1-45> устанавливает значение Reachability Half-life time в минутах (время с момента успешного возобновления соединения до снятия штрафных очков (penalty))</p>
Extcommunity / extcommunity-list	Атрибут для использования регулярных выражений
Local Preference	<p>Атрибут указывает на выбор маршрутизатора, который будет использован для выхода из данной автономной системы.</p> <p>Команда set local-preference <0-4294967295></p>
Metric	Атрибут Multiexit_Descriinator (MED) представляется аналогом метрики маршрута, устанавливается командой set metric <1-4294967295> , по умолчанию MED равен нулю.
Origin	Атрибут указывает на то, каким образом был получен маршрут в обновлении. Значение меняется командой set origin
Originator-ID <0 1 2>	Атрибут указывает Router ID того маршрутизатора, который анонсировал маршрут внутри локальной AS. Если маршрутизатор получает обновление, в котором указан его RID, то этот маршрут не используется и не передается далее соседям. Значение устанавливается командой set originator-id .

Атрибут	Описание
	Возможные значения атрибута: 0 – IGP: NLRI получена внутри исходной автономной системы; 1 – EGP: NLRI выучена по протоколу Exterior Gateway Protocol (EGP). Предшественник BGP, не используется; 2 – Incomplete: NLRI была выучена каким-то другим образом
Vpnv4	Атрибут позволяет задать адрес следующего узла в пути для VPN. Команда set vpnv4 next-hop <address> , где <address> - адрес следующего роутера
Weight	Атрибут определяет, через какой интерфейс будет осуществляться выход из нашей AS. Чем выше вес, тем приоритетнее интерфейс. Для изменения значения используется команда set weight

Для одновременной конфигурации большого количества соседств удобнее использовать группы соседей и применять политики ко всей группе. Конфигурация потребует нескольких команд:

- **neighbor <name> peer-group**, где name - это имя группы;
- **neighbor <address> peer-group <name>** - привязка соседа к группе.

9.6.4 Пример настройки BGP

Рассмотрим пример настройки топологии:

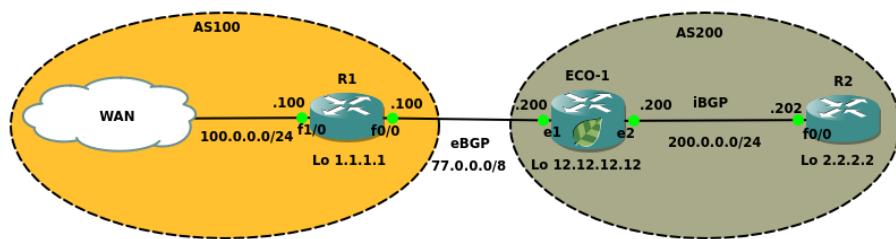


Рисунок 17

Задача: установить соседские отношения между R1-ECO1 и ECO1-R2, изменить значение атрибута MED для маршрутов, проанонсированных с R1 так, чтобы для сетей 33.0.0.0/29 метрика была равна 1000, а для 33.0.0.8/29 метрика была равна 500.

Настройка ECO1:

Шаг 1. Переход в режим конфигурации

```
ECO1>enable
ECO1#configure terminal
```

Шаг 2. Настройка интерфейсов, сервисных интерфейсов, портов.

```
ECO1(config)#interface e1
ECO1(config-if)#interface e1
ECO1(config-if)#ip address 77.0.0.200/8
ECO1(config-if)#interface e2
ECO1(config-if)#ip address 200.0.0.200/24
ECO1(config-if)#port ge1
ECO1(config-port)#service-instance ge1/e1
ECO1(config-service-instance)#encapsulation untagged
ECO1(config-service-instance)#connect ip interface e1
ECO1(config-service-instance)#exit
```

```
ECO1(config-port)#port ge2
ECO1(config-port)#service-instance ge2/e2
ECO1(config-service-instance)#encapsulation untagged
ECO1(config-service-instance)#connect ip interface e2
ECO1(config-service-instance)#exit
ECO1(config-port)#exit
```

Шаг 3. Настройка списков фильтрации

```
ECO1(config)#policy-filter-list 1 permit 33.0.0.0 0.0.0.7
ECO1(config)#policy-filter-list 2 permit 33.0.0.8 0.0.0.7
```

Шаг 4. Привязка списков фильтрации и назначение метрики для сетей

```
ECO1(config)#route-map bgp permit 1
ECO1(config-route-map)#match ip address 1
ECO1(config-route-map)#set metric 1000
ECO1(config-route-map)#route-map bgp permit 2
ECO1(config-route-map)#match ip address 2
ECO1(config-route-map)#set metric 500
```

Шаг 5. Создание пустого списка фильтрации для всех остальных маршрутов с метрикой по умолчанию

```
ECO1(config-route-map)#route-map bgp permit 3
ECO1(config-route-map)#exit
```

Шаг 6. Создание и описание групп соседей

```
ECO1(config)#router bgp 200
ECO1(config-router)#neighbor eBGP peer-group
ECO1(config-router)#neighbor eBGP remote-as 100
ECO1(config-router)#neighbor eBGP ebgp-multipath 2
ECO1(config-router)#neighbor eBGP update-source loopback.0
ECO1(config-router)#neighbor eBGP route-map bgp in
ECO1(config-router)#neighbor iBGP peer-group
ECO1(config-router)#neighbor iBGP remote-as 200
ECO1(config-router)#neighbor iBGP update-source loopback.0
ECO1(config-router)#neighbor iBGP next-hop-self
ECO1(config-router)#neighbor 1.1.1.1 peer-group eBGP
ECO1(config-router)# neighbor 2.2.2.2 peer-group iBGP
ECO1(config-router)#exit
```

Шаг 7. Создание статических маршрутов

```
ECO1(config)#ip route 1.1.1.1/32 77.0.0.100
ECO1(config)#ip route 2.2.2.2/32 200.0.0.202
```

Пример вывода информации таблицы BGP представлен на рисунке ниже.

```

ECO1#
ECO1#
ECO1#sh ip bgp
BGP table version is 2, local router ID is 12.12.12.12
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

      Network          Next Hop           Metric   LocPrf     Weight Path
*-> 33.0.0/30        1.1.1.1          1000     100       0      100 i
*-> 33.0.0.4/30      1.1.1.1          1000     100       0      100 i
*-> 33.0.0.8/30      1.1.1.1          500      100       0      100 i
*-> 33.0.0.12/30     1.1.1.1          500      100       0      100 i

Total number of prefixes 4
ECO1#

```

Рисунок 18

Для помещения маршрутов в BGP и дальнейшего анонсирования следует воспользоваться командой **network** либо сделать редистрибуцию из Interior Gateway Protocols (далее IGP) командой **redistribute**.

Таблица 55

connected	Включить в редистрибуцию маршруты к присоединенным сетям
isis	Включить в редистрибуцию маршруты, полученные через протокол IS-IS
ospf	Включить в редистрибуцию маршруты, полученные через протокол OSPF
rip	Включить в редистрибуцию маршруты, полученные через протокол RIP
static	Включить в редистрибуцию статические маршруты

Для анонса Loopback интерфейса маршрутизатора R2 используем команду **network**

```
ECO1(config-router)#network 2.2.2.2 mask 255.255.255.255
```

В реализации EcoRouterOS синхронизация выключена по умолчанию, для включения используется команда **synchronization** в режиме конфигурирования протокола.

9.6.5 Фильтрация и соседские отношения в BGP

Фильтрация маршрутов в BGP осуществляется подобно IGP протоколам, однако политики указываются конкретно для каждого соседа с отметкой направления **in** или **out**.

В таблице ниже представлены команды для фильтрации маршрутов в BGP.

Таблица 56

Команда	Список, на который ссылается команда
neighbor allowas-in	
neighbor soft-reconfiguraiton inbound	
neighbor capability dynamic	
neighbor capability orf prefix-list	
neighbor capability route-refresh	
neighbor distribute-list	policy-filter-list
neighbor prefix-list	ip prefix-list
neighbor filter-list	ip as-path access-list
neighbor route-map	route-map

Информацию по различным типам списков можно найти в соответствующих разделах, в данном разделе описаны только **AS-path** списки. **AS-path** списки позволяют фильтровать маршруты, основываясь на автономных системах, перечисленных в списке атрибута **AS-path**. Для этого используются регулярные выражения (подробнее см. Сервисные интерфейсы). Для

управления маршрутными политиками используется команда **ip as-path access-list <номер> permit/deny <регулярное выражение>**.

9.6.6 Обновление партнерских BGP отношений

В таблице ниже представлены команды для обновления партнерских BGP отношений.

Таблица 57

Команда	Тип обновления	Количество соседей, направление
clear ip bgp	Жесткий	Все, входящие/исходящие
clear ip bgp neighbor-id	Жесткий	Один, входящие/исходящие
clear ip bgp neighbor-id in/out	Мягкий	Один, входящие/исходящие
clear ip bgp neighbor-id soft in/out	Мягкий	Один, входящие/исходящие
clear ip bgp soft	Мягкий	Все, входящие/исходящие
clear ip bgp neighbor-id soft	Мягкий	Один, входящие/исходящие

Под "жестким" типом обновления подразумевается обновление соседских отношений со сбросом TCP сессии.

Под "мягким" типом обновления подразумевается обновление соседских отношений без сброса TCP сессии.

Для работы функциональности **clear ip bgp neighbor-id in** требуется наличие команды **neighbor <address> soft-reconfiguration inbound** в конфигурации протокола.

Пользователям часто приходится менять политики фильтрации маршрутов в BGP. Крупные изменения в таблицах маршрутизации и сброс TCP-сессий с BGP-соседями вызывают всплеск нагрузки на центральный процессор маршрутизатора. Чтобы уменьшить этот эффект и сделать работу с BGP-соседями и анонсами маршрутной информации более удобной и гибкой, в EcoRouterOS предусмотрен функционал отключения автообновления маршрутной информации при смене политик фильтрации. В BGP маршрутные политики могут настраиваться следующими способами:

- при помощи префикс-листов (prefix-list);
- при помощи карт маршрутов (route-map);
- при помощи специальных листов для маршрутных политик (policy-filter-list);
- при помощи листов распределения (distribute-list);
- при помощи листов фильтрации (filter-list) с листами по BGP AS (ip as-path access-list).

По умолчанию, при создании или изменении политики фильтрации в направлении к соседу маршрутизатор отправит сообщение с анонсами маршрутов (BGP Update) через 30 сек (в случае EBGP-соседства) или мгновенно (в случае iBGP-соседства).

Пример:

```
ip prefix-list 1 deny 1.1.1.1/32
neighbor 10.0.0.2 prefix-list 1 out
```

Изменить временной интервал можно командой **neighbor <IP> advertisement-interval <VALUE>**, где <VALUE> указывается в секундах. Выключить подобное поведение можно командой **neighbor <IP> disable-auto-refresh**. Тогда для отправки маршрутной информации соседу необходимо будет сбросить соседские отношения. Для этого без сброса

TCP-сессий нужно сбросить соседские отношения (мягкий сброс) - при вызове команды **clear ip bgp ...** следует добавить ключевое слово **soft**.

По умолчанию, при создании или изменении фильтрующей политики в направлении от соседа маршрутизатор мгновенно (в обоих случаях - EBGP и iBGP соседства) отправит сообщение с запросом обновления маршрутной информации от соседа (BGP Route-Refresh), но только в том случае, если сосед поддерживает эту опцию.

Пример:

```
ip prefix-list 1 deny 1.1.1.1/32
neighbor 10.0.0.2 prefix-list 1 in
```

Это поведение вызвано опцией BGP Auto-Refresh, которое в EcoRouterOS включено по умолчанию. Выключить подобное поведение можно командой **neighbor 10.0.0.2 disable-auto-refresh**, тогда для отправки запроса на обновление маршрутной информации от соседа необходимо будет сбросить соседские отношения (без сброса TCP сессий добавьте ключевое слово **soft** в командах **clear ip bgp ...**). Здесь также необходима поддержка опции BGP Route-Refresh у соседа.

Команда **no neighbor 10.0.0.2 capability route-refresh** позволит отключить поддержку опции BGP Route-Refresh и исключить возможность отправки сообщений BGP Route-Refresh соседу.

Внимание! Настоятельно рекомендуется отключать функционал auto-refresh для соседей, если те передают слишком большое количество анонсов в BGP.

Для проверки поддержки этой опции у соседа можно воспользоваться командой:

```
ecorouter#show ip bgp neighbors
BGP neighbor is 10.0.0.2, remote AS 2, local AS 1, external link
  BGP version 4, remote router ID 100.100.100.100
  BGP state = Established, up for 02:07:11
  Last read 02:07:11, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    Address family IPv4 Unicast: advertised and received
  Received 315 messages, 0 notifications, 0 in queue
```

.....Вывод сокращен.....

Фраза в выводе **«advertised and received»** говорит о включенной опции BGP Route-Refresh как на локальном маршрутизаторе, так и у соседа.

Результат отключения этой опции на локальном устройстве показан ниже:

```
ecorouter#show ip bgp neighbors
BGP neighbor is 10.0.0.2, remote AS 2, local AS 1, external link
  BGP version 4, remote router ID 100.100.100.100
  BGP state = Established, up for 02:07:11
  Last read 02:07:11, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: received (old and new)
    Address family IPv4 Unicast: advertised and received
  Received 315 messages, 0 notifications, 0 in queue
```

.....Вывод сокращен.....

9.6.7 Регулярные выражения

В реализации EcoRouterOS представлен следующий набор регулярных выражений (см. таблицу ниже):

Таблица 58

Выражение	Использование
^	Начало строки
\$	Конец строки
[]	Диапазон значений
-	Спецификация диапазона, например, [0-9]
()	Логическая группа
.	Любое значение
*	Ноль или большее количество совпадений с предыдущим символом
+	Одно или большее количество совпадений с предыдущим символом
?	Ноль или одно совпадение с предыдущим символом
_	Старт и конец строки, пробел, запятая, открытие или закрытие скобок

Приведем несколько примеров часто используемых регулярных выражений:

.* - любое значение попадает под это правило,

^\$ - маршрут из локальной AS,

^100_ - информация о маршруте получена из AS 100,

_100\$ - подсеть находится в AS 100,

100 - маршрут проходит через AS 100,

^[0-9]+\$ - маршрут из непосредственно подключенной (соседней) AS.

9.6.8 Рефлекторы маршрутов и конфедерации

Рефлектором называется маршрутизатор, который выполняет функцию отражения маршрутов. Рефлектор получает маршрут от одного соседа и рассыпает его всем другим. Это позволяет уменьшить количество связей для создания полносвязной топологии при обучении соседей всем маршрутам в AS и избежать образования петель.

При администрировании большого BGP домена требуется сконфигурировать рефлекторы. Для этого на маршрутизаторе-рефлекторе используется команда **neighbor <address> route-reflector-client**.

Рефлекторы маршрутов не оказывают влияния на пути, по которым IP пакеты проходят через сеть, но определяют порядок распространения маршрутной информации в сети.

Конфедерация - группа автономных систем, анонсируемых с общим номером AS внешним узлам BGP. Работа рефлектора рассматривается с точки зрения протокола iBGP, конфедерации функционируют на уровне автономных систем. Применение конфедераций позволяет разбить автономную систему на подсистемы, которые обмениваются маршрутной информацией с помощью протокола eBGP. При создании конфедерации необходимо на всех маршрутизаторах применить команду **bgp confederation identifier <1-65535>** с указанием номера конфедерации. Соседние AS, которые должны принадлежать конфедерации,

указываются с помощью команды **bgp confederation peers <numberAS1 numberAS2 ...>**. Номера всех нужных соседних AS указываются через пробел.

9.6.9 Команды конфигурирования BGP

Команды конфигурирования протокола BGP представлены в таблице ниже. Данные команды доступны в конфигурационном режиме и в контекстном режиме конфигурирования маршрутизатора (**config-router**)#.

Таблица 59

Команда	Режим	Описание
router bgp <номер AS>	конфигурационный	Переход в режим конфигурирования протокола BGP
address-family ipv4 { unicast multicast }	контекстный	Переход в режим конфигурирования address-family
aggregate-address <адрес>	контекстный	Создание суммарного маршрута
auto-summary	контекстный	Включение автосуммаризации
bgp always-compare-med	контекстный	Сравнение атрибутов MED для маршрута, полученного из разных AS определяет лучший путь
bgp as-local-count <2-64>	контекстный	Определяет количество значений собственной AS в атрибуте AS-path
bgp bestpath ...	контекстный	Изменение алгоритма выбора лучшего пути
bgp client-to-client reflection	контекстный	Включение роли рефлектора
bgp cluster-id <1-4294967295>	контекстный	Указание номера кластера
bgp confederation identifier <1-65535>	контекстный	Указание номера конфедерации
bgp confederation peers <1-65535>	контекстный	Указание соседей в конфедерации
bgp config-type { standard ecorouteros }	конфигурационный	Определение типа конфигурации, по умолчанию включена ecorouteros , для передачи атрибута community используется тип standard
bgp dampening ...	контекстный	Настройка подавления нестабильных маршрутов
bgp default local-preference <0-4294967295>	контекстный	Указание значения атрибута local preference
bgp deterministic-med	контекстный	Сравнение атрибутов MED для маршрута, полученного из одной AS; атрибуты weight , local preference , AS-path и origin должны быть равны
bgp enforce-first-as	контекстный	Update сообщение, которое пришло не от соседней сконфигурированной AS, будет отброшено
bgp fast-external-failover	контекстный	Моментальный сброс BGP сессии при падении интерфейса
bgp nexthop-trigger delay <1-100>	конфигурационный	Установка задержки на изменения параметров в BGP таблице при каких-либо изменениях параметров у соседнего маршрутизатора

Команда	Режим	Описание
bgp nexthop-trigger enable	конфигурационный	Включение специального мониторинга адреса соседа
bgp rfc1771-path-select	конфигурационный	Включение алгоритма выбора лучшего пути согласно RFC 1771
bgp rfc1771-strict	конфигурационный	Установка атрибута origin согласно RFC 1771
bgp router-id <адрес>	контекстный	Указание BGP идентификатора маршрутизатора
bgp scan-time <0-60>	контекстный	Значение интервала сканирования доступности маршрутов в таблице маршрутизации (по умолчанию 60 сек)
distance bgp <1-255> <1-255> <1-255>	контекстный	Установка административных расстояний для external, internal, local маршрутов
max-paths {ebgp ibgp} <2-64>	контекстный	Максимальное количество возможных маршрутов, которые считаются равными
mpls-resolution	контекстный	Автоматическое создание FTN записи для префиксов, полученных от BGP соседей
neighbor <адрес> activate	контекстный	Активация соседских отношений при конфигурировании address-family
neighbor <адрес> advertisement-interval <0-65535>	контекстный	Указание минимального интервала для отправки Update сообщений
neighbor <адрес> allowas-in <1-10>	контекстный	Указание на рекламу префиксов (маршрутов) даже если их источник в той же AS
neighbor <адрес> as-origination-interval <1-65535>	контекстный	Указание минимального интервала для отправки AS-origination Update сообщений
neighbor <адрес> attribute-unchanged [as-path next-hop med]	контекстный	При изменении значений атрибутов отправлять значения по умолчанию
neighbor <адрес> capability dynamic	контекстный	Включение динамических возможностей для определенного узла
neighbor <адрес> capability orf prefix-list	контекстный	Включение фильтрации ORF и анонсирования ORF соседям
neighbor <адрес> capability route-refresh	контекстный	Включение анонсирования поддержки возможностей обновления маршрутов
neighbor <адрес> connection-retry-time <1-65535>	контекстный	Установка таймера повторного соединения с соседом (по умолчанию 120 сек.)
neighbor <адрес> default-originate	контекстный	Отправка соседу маршрута по умолчанию
neighbor <адрес> description	контекстный	Описание для соседнего маршрутизатора (максимум 80 символов)
neighbor <адрес> disable-infinite-holdtime	контекстный	Невозможность задать бесконечный hold-time

Команда	Режим	Описание
neighbor <адрес> disable-capability-negotiate	контекстный	Отключение проверки на совместимость версий протокола (отключена по умолчанию)
neighbor <адрес> ebgp-multipath <1-255>	контекстный	Установка значения TTL в BGP пакетах при eBGP сессии
neighbor <адрес> enforce-multipath	контекстный	Включение принудительного непрямого соседства
neighbor <адрес> local-as <1-4294967295>	контекстный	Указание номера локальной AS
neighbor <адрес> maximum-prefix <1-4294967295>	контекстный	Контроль количества принимаемых маршрутов от соседа
neighbor <адрес> next-hop-self	контекстный	Отправка информации о Next-Hop соседям iBGP
neighbor <адрес> passive	контекстный	Включение пассивного режима
neighbor <адрес> password	контекстный	Задание MD5 пароля для аутентификации (максимум 80 символов)
neighbor {имя адрес} peer-group <имя>	контекстный	Создание группы соседей/добавление в группу
neighbor <адрес> port <0-65535>	контекстный	Указание BGP порта для соседа
neighbor <адрес> remote-as	контекстный	Задание номера AS соседа
neighbor <адрес> remove-private-AS	контекстный	Удаление AS из приватного диапазона при отправке обновлений
neighbor <адрес> route-reflector-client	контекстный	Включение роли рефлектора и указание соседа в роли клиента
neighbor <адрес> route-server-client	контекстный	Указание соседа в роли сервер-клиента
neighbor <адрес> send-community <both/extended/standard>	контекстный	Отправка атрибута community
neighbor <адрес> shutdown	контекстный	Административное выключение BGP отношений
neighbor <адрес> soft-reconfiguration inbound	контекстный	Включение локальной базы данных для принятых маршрутов
neighbor <адрес> timers <0-65535> <0-65535> [connect <1-65535>]	контекстный	Задание keepalive, hold и connect таймеров
neighbor <адрес> transparent-as	контекстный	Включение режима прозрачной AS, не включать значение собственной AS в атрибут AS-path
neighbor <адрес> transparent-nexthop	контекстный	Включение режима прозрачной AS, не указывать себя в качестве Next-Hop к маршруту
neighbor <адрес> unsuppress-map <имя группы>	контекстный	Реклама более специфических маршрутов для соседа при созданном суммарном маршруте
neighbor <адрес> update-source <адрес>	контекстный	Указание интерфейса для создания TCP-сессии
neighbor <адрес> weight <0-65535>	контекстный	Задание атрибута weight

Команда	Режим	Описание
network <адрес>	контекстный	Указание подсетей для рекламы
redistribute {connected isis rip static}	контекстный	Редистрибуция в BGP
synchronization	контекстный	Включение режима синхронизации
timers bgp <0-65535> <0-65535>	контекстный	Задание keepalive и hold таймеров

9.6.10BGP. Команды просмотра

Команды просмотра информации о настройках и статистике протокола BGP представлены в таблице ниже.

Таблица 60

Команда	Описание
show bgp statistics	Вывод статистических данных
show ip bgp	Выводит BGP таблицу
show ip bgp <адрес сети>	Выводит информацию о конкретном маршруте
show ip bgp attribute-info	Отображает информацию обо всех внутренних атрибутах
show ip bgp community	Выводит список маршрутов, принадлежащих к конкретному сообществу
show ip bgp community-info	Выводит информацию о сообществах
show ip bgp dampening {dampened-paths flap-statistics parameters} vrf {<vrf-name> all default}	Выводит информацию о подавлении нестабильных маршрутов
show ip bgp filter-list	Выводит список маршрутов, соответствующий AS-path списку
show ip bgp ipv4 <unicast/multicast> ...	Вывод информации для address-family
show ip bgp neighbors	Выводит информацию обо всех сконфигурированных соседях
show ip bgp neighbors <address> advertised-routes	Выводит информацию о рекламируемых маршрутах, которые прошли исходящую фильтрацию
show ip bgp neighbors <address> routes	Выводит информацию о получаемых маршрутах, которые прошли входящую фильтрацию
show ip bgp neighbors <address> received-routes*	Выводит информацию о получаемых маршрутах до каких-либо входных фильтров
show ip bgp paths	Отображает информацию о маршрутах для локального маршрутизатора
show ip bgp prefix-list	Выводит список маршрутов, соответствующий списку префиксов
show ip bgp regexp	Выводит список маршрутов, соответствующий регулярному выражению
show ip bgp route-map	Выводит список маршрутов, соответствующий карте маршрутов
show ip bgp summary	Отображает состояние всех соединений BGP

9.6.11Dampening

Подавление переключающихся маршрутов (dampening) - это инструмент управления, предназначенный для уменьшения нестабильности и нежелательных колебаний в сети. Нежелательные переключения маршрутов возникают в случае, когда маршруты то появляются в таблице маршрутизации, то пропадают. Это может быть вызвано обрывами

линков, ошибками в работе устройств, неправильной настройкой оборудования и т.п. Переключающиеся маршруты в таблице маршрутизации повышают нагрузку на процессоры сетевых устройств, что может привести к более серьезным проблемам в сети. Использование технологии подавления переключающихся маршрутов является хорошей инженерной практикой, которую можно встретить в сетях у многих провайдеров.

Переключающийся маршрут за каждое переключение получает штрафные баллы. Эти штрафные баллы суммируются в реальном времени. Когда превышается установленный "предел для подавления", нестабильный маршрут исключается из анонсирования. Накопленный маршрутом штраф автоматически уменьшается со временем на основании заданного "времени уменьшения штрафа вдвое" (Half-life time). Когда значение штрафа станет ниже "предела для повторного использования" подавление будет снято, и маршрут снова станет анонсироваться.

После того как значение штрафа для маршрута станет меньше половины "предела для повторного использования", информация о подавлении маршрута удаляется из маршрутизатора.

Для задания параметров отключения переключающихся маршрутов в контекстном режиме конфигурирования bgp-маршрутизатора используется команда **bgp dampening {route-map <ROUTE-MAP-NAME> | <REACHIBILITY-HALF-LIFE-TIME> <REUSE-VALUE> <SUPPRESS-VALUE> <MAX-SUPPRESS-VALUE> <UN-REACHABILITY-HALF-LIFE-TIME>}**. Команда также позволяет в явном виде указать карту маршрутов для подавления.

Таблица 61

Параметр	Описание
<ROUTE-MAP-NAME>	Имя карты маршрутов для подавления
<REACHIBILITY-HALF-LIFE-TIME>	Время доступности в минутах, за которое штраф уменьшается вдвое (default reachability half-life time). Допустимый диапазон 1-45. Значение по умолчанию - 15
<REUSE-VALUE>	Значение предела для повторного использования маршрута. Когда значение штрафа опускается ниже этого значения, маршрут перестает подавляться. Допустимый диапазон 1-20000. Значение по умолчанию 750
<SUPPRESS-VALUE>	Значение предела для подавления маршрута. Когда значение штрафа превышает это значение, маршрут подавляется. Допустимый диапазон 1-20000. Значение по умолчанию 2000
<MAX-SUPPRESS-VALUE>	Максимальная продолжительность подавления стабильного маршрута в минутах. Допустимый диапазон 1-255. Значение по умолчанию в 4 раза больше времени доступности, за которое штраф уменьшается вдвое, или 60 минут
<UN-REACHABILITY-HALF-LIFE-TIME>	Время недоступности в минутах, за которое штраф уменьшается вдвое. Допустимый диапазон 1-45. Значение по умолчанию - 15

Пример:

```
#configure terminal
(config)#router bgp 11
(config-router)#bgp dampening 20 800 2500 80 25
```

9.6.12 Background BGP scanners

Данные параметры отвечают за сканирование таблиц BGP RIB и IP RIB маршрутизатора, а также за сортировку, отправку и удаление записей из них. Как известно, BGP использует

только маршруты с доступным next-hop и в случае его исчезновения удаляет подсети из таблиц. Эти действия определяются значением таймера **background bgp next-hops**, по умолчанию все маршруты проверяются 1 раз в 60 секунд.

Изменить значения данного таймера можно в контекстном режиме конфигурирования bgp командой **bgp scan-time next-hops <0-60>**. При указании значения 0 сканирование будет отключено.

Помимо доступности next-hop BGP сканирует таблицы маршрутизатора на предмет наличия новых статических записей и маршрута 0.0.0.0. Эти действия определяются значением таймера **background bgp networks**, по умолчанию все маршруты проверяются 1 раз в 15 секунд.

Изменить значения данного таймера можно в контекстном режиме конфигурирования bgp командой **bgp scan-time networks <15-60>**.

Для снижения нагрузки на CPU устройства сетевой инженер может выставить максимальные значения таймеров сканирования, но при этом будет увеличено время сходимости сети.

9.6.13 Команды clear

Для очистки информации о подавлении нестабильных маршрутов для протокола BGP для выбранной сети или VRF предназначена команда **clear ip bgp dampening**, запускаемая в административном режиме. Синтаксис команды следующий: **clear ip bgp dampening [<ADDRESS>/<MASK>] [vrf {<VRF-NAME>} | default | all]**.

Таблица 62

Параметр	Описание
<ADDRESS>/<MASK>	Очистить информацию для подсети с указанным IP и маской, например, 35.0.0.0/8
vrf {<VRF-NAME>} default all	Очистить информацию о сущности VRF - с указанным именем VRF-NAME, выбранной по умолчанию (default) или для всех VRF-сущностей (all)

Пример:

```
#clear ip bgp dampening 35.0.0.0/8
```

Для очистки информации о протоколе BGP (статистики и данных по IPv4) предназначена группа команд **clear bgp**, запускаемая в административном режиме.

Для очистки статистики синтаксис команды следующий: **clear bgp statistics**.

Для очистки данных по IPv4 синтаксис команды следующий: **clear bgp ipv4 {multicast | unicast} { * | <AS-number> | <ADDRESS>/<MASK> } | flap-statistics { <ADDRESS>/<MASK> } | vrf {<VRF-NAME>} | all | default }**.

Таблица 63

Параметр	Описание
<ADDRESS>/<MASK>	IP-адрес подсети и маска, например, 35.0.0.0/8
multicast unicast	Выбрать режим multicast или unicast
<AS-number>	Номер автономной системы в диапазоне от 1 до 4294967295
flap-statistics	Очистить статистику по переключающимся маршрутам выбранной по адресу и маске (ADDRESS/MASK) или имени

Параметр	Описание
	(VRF-NAME) сущности, всех сущностей (all) или сущности по умолчанию (default)

Примеры:

```
#clear bgp statistics
#clear bgp ipv4 unicast flap-statistics all
```

9.6.14BGP Blackhole

В качестве одного из методов защиты от DDoS атак в EcoRouterOS предусмотрен функционал отбрасывания трафика через Null-интерфейс путем его подстановки в качестве адреса следующего узла для BGP-маршрутов. Подобные сценарии являются эффективным средством борьбы с крупными и масштабными атаками, целью которых является довести атакуемую сеть до «отказа в обслуживании». Более подробную информацию о всех преимуществах и недостатках этого функционала можно найти в Интернете.

Ниже рассматривается пример сценария и конфигурации EcoRouter.

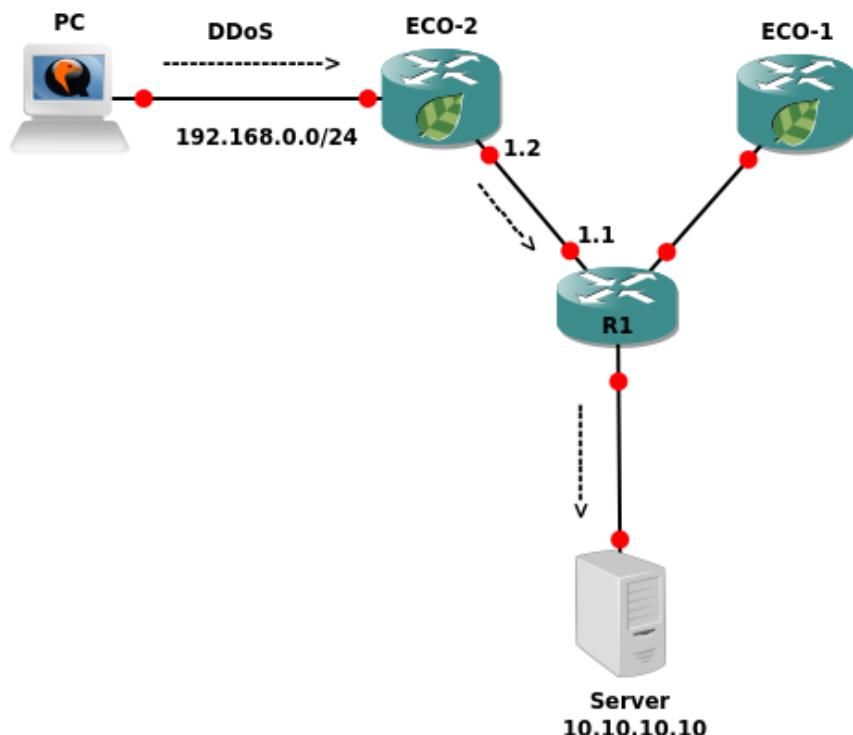


Рисунок 19

Допустим, что злоумышленник PC из сети 192.168.0.0/24 подает огромное количество трафика в BGP AS на Server 10.10.10.10/32, пытаясь вызвать неработоспособность сервера. Задача сводится к тому, что необходимо с устройства R1 отправить рекламу об адресе 10.10.10.10/32 с определенным номером атрибута community. Маршрутизатор EcoRouter ECO-2, приняв рекламу с этим маршрутом, должен обновить данные в RIB и начать отбрасывать все пакеты, приходящие с PC в сторону адреса 10.10.10.10/32. Конфигурация ECO-2 может выглядеть следующим образом:

```
ecorouter#sh running-config
```

```
!
no service password-encryption
!
hw mgmt ip 192.168.255.1/24
!
ip vrf management
!
mpls propagate-ttl
!
security default
security none vrf management
!
ip pim register-rp-reachability
!
router bgp 1
redistribute connected
neighbor 1.1.1.1 remote-as 1
neighbor 1.1.1.1 soft-reconfiguration inbound
neighbor 1.1.1.1 route-map BLACKHOLE in
!
ip route 9.9.9.9/32 Null
!
ip community-list 66 permit 1:777
!
route-map BLACKHOLE permit 10
match community 66
set ip next-hop 9.9.9.9
!
route-map BLACKHOLE permit 20
!
line con 0
line vty 0 39
!
traffic-class default
!
port te0
lacp-priority 32767
mtu 9728
service-instance 1
  encapsulation untagged
!
port tel
lacp-priority 32767
mtu 9728
service-instance 1
  encapsulation untagged
!
interface 1
ip mtu 1500
connect port tel service-instance 1
ip address 1.1.1.2/24
!
interface 2
```

```
ip mtu 1500
connect port te0 service-instance 1
ip address 192.168.0.1/24
vrf management
```

Обратите внимание на статический маршрут в Null-интерфейс и инструкцию **set ip next-hop 9.9.9.9** в карте маршрутов. Это главные условия для установки рекурсивного маршрута в RIB через Null-интерфейс. Пример вывода таблицы маршрутизации:

```
ecorouter#sh ip ro
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter
area
      * - candidate default
IP Route Table for VRF "default"
C    1.1.1.0/24 is directly connected, 1
S    9.9.9.9/32 [1/0] is a summary, Null
B    10.10.10.0/24 [200/0] via 1.1.1.1, 1, 00:08:45
B    10.10.10.10/32 [200/0] via 9.9.9.9 (recursive blackhole), 00:08:45
C    192.168.0.0/24 is directly connected, 2
Gateway of last resort is not set
```

В примере использовался протокол iBGP, при необходимости этот функционал можно использовать и в eBGP топологии, однако, для создания рекурсивного маршрута через Null потребуется указание команды **neighbor <адрес> ebgp-multipath <значение>** для соседа, который отсылает информацию о маршруте с атрибутом **community** (в примере адрес соседа 1.1.1.1) или создать loopback-интерфейс на EcoRouter с адресом из подсети используемого BGP next-hop в карте маршрутов (route-map).

9.7 Карты маршрутов

Карты маршрутов (route-map) применяются для управления формированием и изменением таблиц маршрутизации, а также процессом передачи маршрутной информации по сети. Они позволяют накладывать определенные требования на анонсируемые маршруты. Если маршрут удовлетворяет условию, указанному в конструкции **match**, то будет выполнено некоторое действие, которое сетевой администратор указывает с помощью команды **set**.

9.7.1 Настройка карт маршрутов

Создание карт маршрутов осуществляется в режиме конфигурирования маршрутизатора. В этом режиме вводится команда **route-map** и имя карты маршрута. Далее задаются условия, которым должна удовлетворять маршрутная информация, и указываются ключевые слова **permit** (разрешить) или **deny** (запретить). После чего необходимо задать номер оператора.

Синтаксис команды создания карты маршрутов: **route-map <имя> permit/deny <номер оператора>**.

После этого в контекстном режиме конфигурирования route-map можно задать условия и действия, осуществляемые при срабатывании данных условий. Эти параметры задаются в паре условие-действие.

```
EcoRouter(config)#route-map <имя> permit/deny <номер>
EcoRouter(config-route-map)#match <условие>
EcoRouter(config-route-map)#set <действие>
```

Если при создании карты маршрутов номер не был задан, то по умолчанию он будет равен 10. Для конфигурирования следующих условий и правил той же route-map номер должен быть задан администратором вручную. С помощью конструкции **match** можно проверить условия, перечисленные в таблице ниже.

Таблица 64

Условие	Описание
as-path	Наличие в BGP маршруте атрибута AS-path, который содержит данные, совпадающие с указанными в ip as-path access-list
community	Наличие в BGP маршруте атрибута community, который содержит данные, совпадающие с указанными в ip community-list
extcommunity	Наличие в BGP маршруте атрибута extcommunity, который содержит данные, совпадающие с указанными в ip extcommunity-list
interface	Совпадение с выходным интерфейсом локального маршрутизатора на основе таблицы маршрутизации
ip address <policy-filter-list>	Сопоставление префикса с policy-filter-list
ip address <prefix-list>	Сопоставление префикса с prefix-list
ip nexthop	Проверяется next-hop адрес маршрута
ip peer	Проверяется BGP сосед для определенного префикса
metric	Проверяется метрика маршрута
origin	Проверяется значение атрибута origin
route-type	Проверяет тип маршрута для OSPF и IS-IS (external, internal, type-1, type-2)
tag	Проверяется тег установленный для маршрута ранее

С помощью конструкции **set** можно выполнить следующие действия:

- установить значения BGP атрибутов (подробнее об установке атрибутов пути через параметр **set** читайте в разделе BGP);
- установить уровень маршрута для протокола IS-IS;
- изменить тип метрики в OSPF и IS-IS с помощью конструкции **metric-type**;
- протегировать маршрут с помощью конструкции **tag**.

9.7.2 Обработка записей в картах маршрутов

Записи в карте маршрутов обрабатываются по порядку, сверху вниз, как и в случае стандартных или расширенных списков доступа. Если обнаружено соответствие маршрута к какому-либо условию в списке, дальнейшая проверка списка прекращается. Нумерация записей применяется только для того, чтобы вставлять или удалять нужные записи в route-map используя параметр **no**. Если в последней записи route-map указать пустое условие с ключевым словом **permit**, то все варианты, не описанные в правилах, будут допустимыми. Если такая

строчка отсутствует в route-map, то все варианты, не описанные в правилах, по умолчанию будут запрещены (применен **deny**).

Для того, чтобы сконфигурировать route-map, которая будет устанавливать тег 7 в единственный маршрут 10.0.0.0/8 и удалять сети 11.0.0.0/8 11.0.0.0/24 из анонса потребуются следующие команды:

```
EcoRouter(config)#ip prefix-list 1 permit 10.0.0.0/8
EcoRouter(config)#ip prefix-list 2 permit 11.0.0.0/8 le 24
EcoRouter(config)#route-map TEST permit 1
EcoRouter(config-route-map) #match ip address prefix-list 1
EcoRouter(config-route-map) #set tag 7
EcoRouter(config-route-map) #route-map TEST deny 2
EcoRouter(config-route-map) #match ip address prefix-list 2
EcoRouter(config-route-map) #route-map TEST permit 3
```

Для удаления последовательности 3 можно воспользоваться командой **no route-map TEST permit 3**.

Для просмотра общей информации по картам маршрутов используется команда **show route-map <имя>**.

10 Настройка туннелирования

Туннелирование – механизм передачи пакета одного протокола внутри другого протокола, позволяющий безопасно передавать данные между двумя сетями.

Туннели являются логическими соединениями типа точка – точка, определяющиеся точкой-источником туннеля и точкой-назначением туннеля.

10.1 GRE

GRE (Generic Routing Encapsulation) – протокольный механизм, использующий IP (UDP) как транспортный протокол. GRE может быть использован для переноса различных протоколов внутри себя.

Для отправки в GRE туннель IP-пакет при прохождении через интерфейс туннеля получает сверху дополнительный заголовок GRE, в котором в качестве адреса источника и адреса назначения будут указаны ip адреса начальной и конечной точки туннеля. После прибытия пакета на интерфейс с адресом назначения туннеля служебный заголовок GRE будет отброшен и далее пакет будет обрабатываться в соответствии со своим «родным» IP заголовком.



Рисунок 20

10.1.1 MTU в протоколах туннелирования

Типичная размерность MTU для L3 интерфейса 1500 байт. В связи с добавлением служебного заголовка появляются новые требования к допустимому значению MTU при передаче пакета. Заголовок GRE имеет размерность 4 байта, 20 байт транспортный IP заголовок, заголовок IP пакета 20 байт, таким образом возникает необходимость задавать размер допустимого MTU на интерфейсах туннеля меньше стандартного значения.

10.1.2 Флаги в GRE

Реализация EcoRouterOS при инкапсулировании во внешнем заголовке устанавливает DF бит равным 1 (не фрагментировать). Если приходящий фрейм в заголовке IP содержит MF бит равным 1 (была фрагментация) или fragment offset бит равный 1 (последний фрагмент первоначального фрейма), то фрейм будет отброшен. При GRE инкапсуляции приходящие фреймы, содержащие в заголовке GRE флаги checksum, routing, key, seq number, strict source route или recursion, отличные от нуля, будут отброшены.

Команды настройки

Таблица 65

Команда	Описание
interface tunnel.<номер>	Создание интерфейса туннеля, где номер произвольное число
ip mtu <значение>	Задание значения mtu для интерфейса

Команда	Описание
ip tunnel <source IP> <destination IP> mode <gre ipip>	Задание IP-адресов начала и конца туннеля и типа туннеля

10.1.3 Пример базовой настройки туннеля GRE

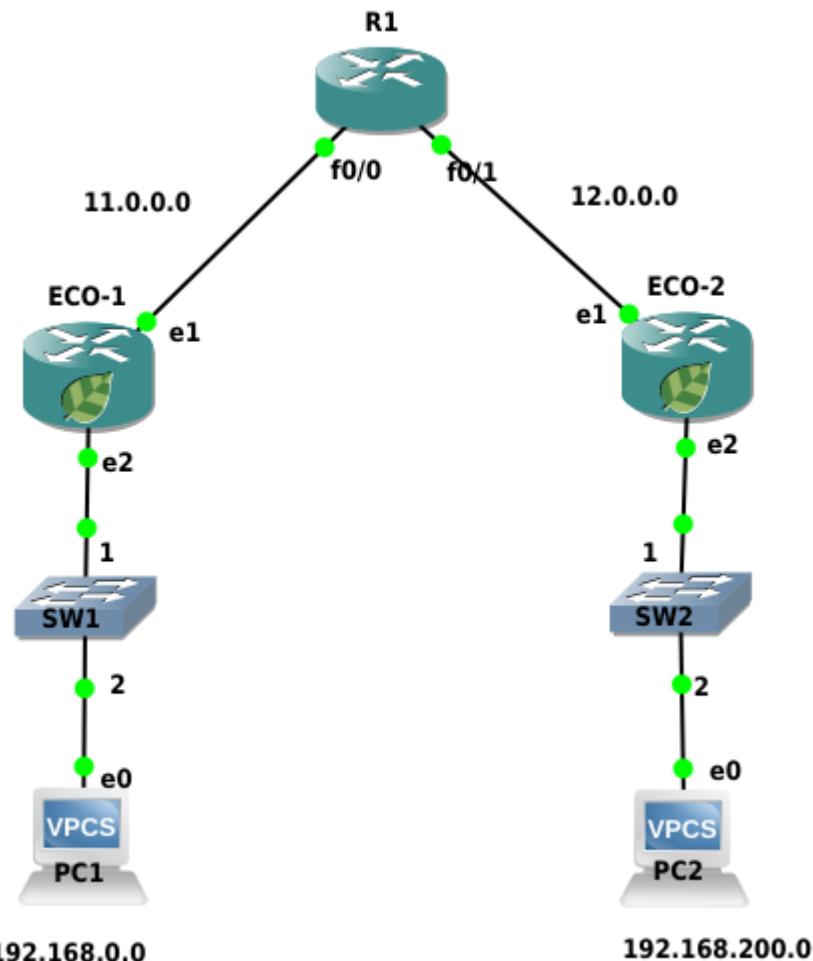


Рисунок 21

Настроим туннель GRE между устройствами ECO-1 и ECO-2. Ниже приведена настройка для устройства ECO-1.

Шаг 1. Настройка интерфейсов и портов.

```
ecorouter>en
ecorouter#conf t
ecorouter(config)#interface e1
ecorouter(config-if)ip add 11.0.0.1/16
ecorouter(config)#interface e2
ecorouter(config-if)ip add 192.168.0.1/24
ecorouter(config)#port te0
ecorouter(config-port)#service-instance te0
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface e1
ecorouter(config)#port tel
ecorouter(config-port)#service-instance tel
```

```
ecorouter(config-service-instance) #encapsulation untagged  
ecorouter(config-service-instance) #connect ip interface e2
```

Шаг 2. Создаем интерфейс туннеля с именем tunnel.0

```
ecorouter(config)#interface tunnel.0
```

Шаг 3. Назначение ip адреса

```
ecorouter(config-if)#ip add 172.16.0.1/16
```

Шаг 4. Выставление параметра MTU

```
ecorouter(config-if)#ip mtu 1400
```

Шаг 5. Задание режима работы туннеля GRE и адресов начала и конца туннеля

```
ecorouter(config-if)#ip tunnel 11.0.0.1 12.0.0.2 mode gre
```

Шаг 6. Настройка маршрутизации трафика в туннель

```
ecorouter(config)#ip route 12.0.0.0/8 11.0.0.2
```

```
ecorouter(config)#ip route 192.168.200.0/24 172.16.0.2
```

Аналогичная настройка производится на втором устройстве.

10.1.4 Команды просмотра

Для просмотра состояния туннеля используется команда **show interface tunnel.<номер туннеля>**.

Для созданной выше конфигурации команда будет отображать следующий результат:

```
ecorouter#sh int tunnel.0  
Interface tunnel.0 is up, line protocol is up  
  Ethernet address: 0000.ab27.8404  
    MTU: 1400  
    Tunnel source: 11.0.0.1  
    Tunnel destination: 12.0.0.2  
    Tunnel mode: GRE  
    ICMP redirection is on  
<UP,BROADCAST,RUNNING,NOARP,MULTICAST>  
  inet 172.16.0.1/16 broadcast 172.16.255.255/16  
    total input packets 0, bytes 0  
    total output packets 0, bytes 0
```

10.2 IP in IP

IP in IP - механизм туннелирования, который помещает один IP пакет в другой IP пакет.

Процесс туннелирования заключается в добавлении ещё одного IP заголовка к стандартному IP пакету. В верхнем заголовке будут содержаться IP адреса начала и окончания туннеля. После доставки на маршрутизатор, на котором находится окончание туннеля, верхний заголовок снимается, пакет передается с обычным, внутренним IP заголовком дальше.



Рисунок 22

10.2.1 MTU в IP in IP

Типичная размерность MTU для L3 интерфейса 1500 байт. В связи с добавлением служебного заголовка появляются новые требования к допустимому значению MTU при передаче пакета. Заголовок IP in IP имеет размерность 20 байт, заголовок IP пакета 20 байт, таким образом возникает необходимость задавать размер допустимого MTU на интерфейсах туннеля меньше стандартного значения для Ethernet.

10.2.2 Флаги в IP in IP

Реализация EcoRouterOS при инкапсулировании во внешнем заголовке устанавливает DF бит равным 1 (не фрагментировать)

Если приходящий фрейм в заголовке IP содержит MF бит равным 1 (была фрагментация) или fragment offset бит равный 1 (последний фрагмент первоначального фрема), то фрейм будет отброшен.

Команды настройки

Таблица 66

Команда	Описание
interface tunnel.<номер>	Создание интерфейса туннеля, где номер произвольное число
ip mtu <значение>	Задание значения mtu для интерфейса
ip tunnel <source IP> <destination IP> mode <gre ipip>	Задание ip-адресов начала и конца туннеля и типа туннеля

10.2.3 Пример базовой настройки туннеля IP in IP

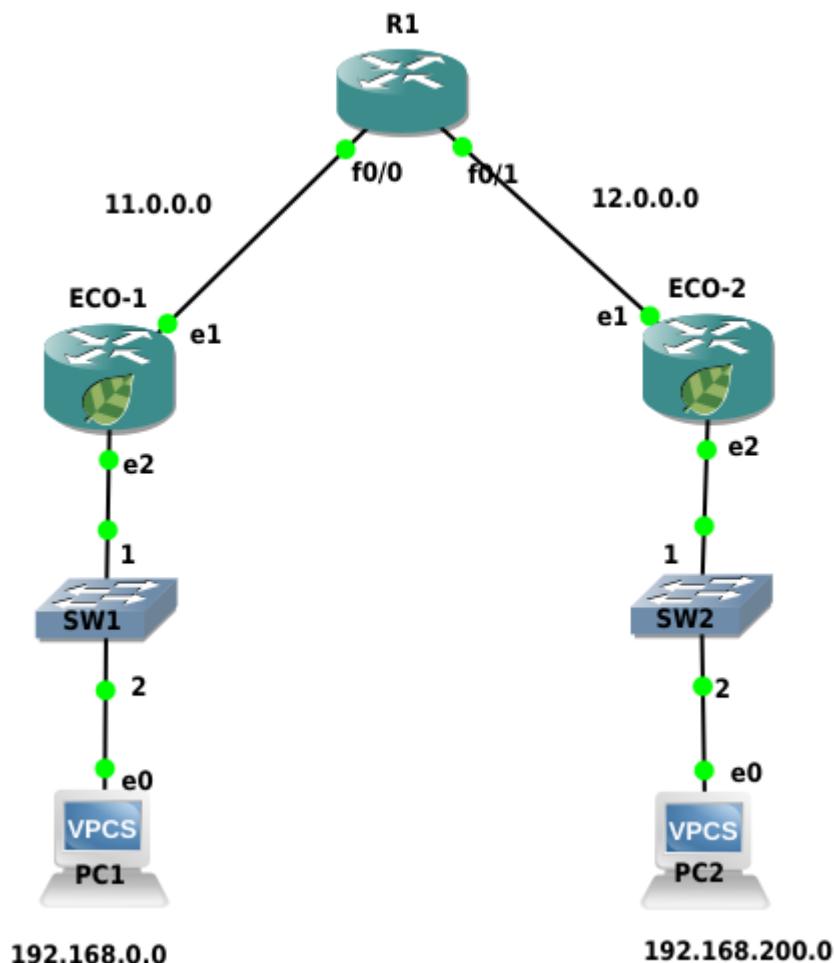


Рисунок 23

Настроим туннель IP-in-IP между устройствами ECO-1 и ECO-2. Ниже приведена настройка для устройства ECO-1

Шаг 1. Настройка интерфейсов и портов.

```

ecorouter>en
ecorouter#conf t
ecorouter(config)#interface e1
ecorouter(config-if)ip add 11.0.0.1/16
ecorouter(config)#interface e2
ecorouter(config-if)ip add 192.168.0.1/24
ecorouter(config)#port te0
ecorouter(config-port)#service-instance te0
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface e1
ecorouter(config)#port te1
ecorouter(config-port)#service-instance te1
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface e2

```

Шаг 2. Создаем интерфейс туннеля с именем tunnel.0

```
ecorouter(config)#interface tunnel.0
```

Шаг 3. Назначение ip адреса

```
ecorouter(config-if)#ip add 172.16.0.1/16
```

Шаг 4. Выставление параметра MTU

```
ecorouter(config-if)#ip mtu 1400
```

Шаг 5. Задание режима работы туннеля IP-in-IP и адресов начала и конца туннеля

```
ecorouter(config-if)#ip tunnel 11.0.0.1 12.0.0.2 mode ipip
```

Шаг 6. Настройка маршрутизации трафика в туннель

```
ecorouter(config)#ip route 12.0.0.0/8 11.0.0.2
```

```
ecorouter(config)#ip route 192.168.200.0/24 172.16.0.2
```

Аналогичная настройка производится на втором устройстве.

11 Бриджинг с поддержкой L3

Сетевой мост (бридж) – физическое или логическое устройство, разделяющее домены коллизий Ethernet и работающее на двух нижних уровнях сетевых стеков OSI и TCP/IP. Объединение двух или более сетевых сегментов называется бриджингом. В простых бриджах широковещательные пакеты рассылаются во все интерфейсы бриджа; бриджи с поддержкой VLAN могут ограничивать широковещательные домены отдельными интерфейсами. Идентификатор VLAN в таких бриджах должен быть уникален в пределах устройства. Широковещательный домен, ограниченный VLAN, получил в стандартах IEEE 802.1Q/802.1ad название VLAN бридж-домен.

С развитием провайдерских технологий появилась потребность ограничивать уникальность VLAN ID отдельным портом. Такую возможность предоставила концепция EVC (Ethernet Virtual Connection), в которой широковещательный L2 домен уже не привязан к VLAN. EVC бридж-домен объединяет виртуальные L2 интерфейсы, называемые сервисными (service instance, SI). L3 интерфейс для связи L2 и L3 доменов в традиционных бриджах называется SVI или BVI, в EVC бридж-доменах для него принято название BDI (сокр. от Bridge Domain Interface).

Диаграммы процессов, происходящих при пересылке фреймов между L2 и L3 доменами с участием BDI в обоих направлениях, приведены на рисунке ниже.

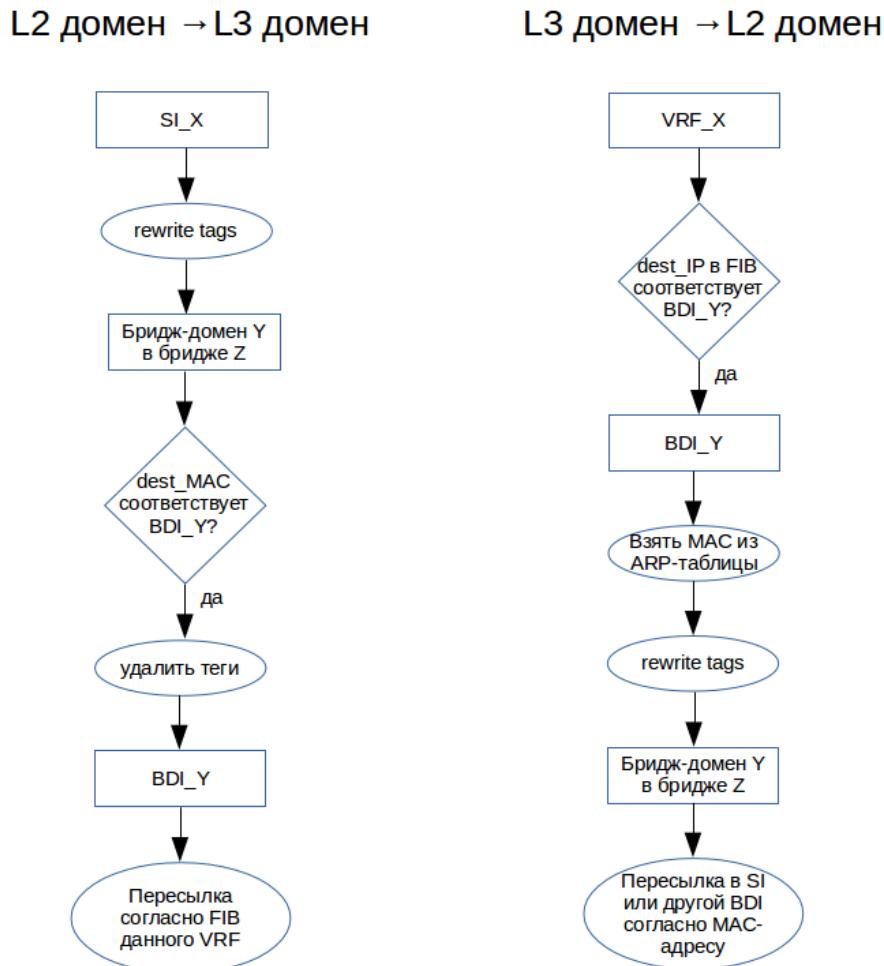


Рисунок 24

11.1 Настройка

Команда создания бриджа:

```
ecorouter(config)#bridge <NAME>
```

где <NAME> – произвольное имя, допустимое в EcoRouterOS.

Бридж-домен создаётся в контексте конфигурирования сервисного интерфейса:

```
ecorouter(config-service-instance) #
```

В таблице приведены соответствующие команды.

Таблица 67

Команда	Описание
encapsulation {default dot1q untagged}	Задание инкапсуляции (тегирования) для внешнего трафика
rewrite {pop push translate}	Преобразование инкапсуляции при отправке в бридж
connect bridge <NAME>	Подключение к созданному ранее бриджу

Тегирование (инкапсуляция) может быть произвольным (см. раздел «Операции над метками в сервисных интерфейсах»), причем, как сказано выше, VLAN ID сервисного интерфейса на одном порту может совпадать с VLAN ID сервисного интерфейса на другом

порту, и это будут разные VLAN, до тех пор, пока эти SI находятся в разных бридж-доменах. Бридж-домен на бриdge образуют подключенные к нему сервисные интерфейсы с одинаковым значением инкапсуляции на бриdge, задаваемым командами **encapsulation** и **rewrite**. Только в этом случае между ними возможен бриджинг. Например, если на одном сервисном интерфейсе задано Q-in-Q тегирование:

```
ecorouter(config-service-instance) #encapsulation dot1q 30 second-dot1q
40
```

а на другом (из того же бридж-домена) задано:

```
ecorouter(config-service-instance) #encapsulation dot1q 20
```

то для бриджинга между ними, к примеру, на первом можно дать команду:

```
ecorouter(config-service-instance) #rewrite translate 2-to-1 20
```

11.2 Создание BDI

Интерфейс BDI создается как обычный L3 интерфейс с двумя дополнительными командами в контексте конфигурирования интерфейса, описанными в таблице ниже.

Таблица 68

Команда	Описание
rewrite push	Преобразование инкапсуляции при отправке в бридж
connect bridge <NAME>	Привязка к созданному ранее бриджу

Команда **encapsulation** здесь отсутствует, т. к. в L3 домен нельзя отправлять тегированный трафик.

Пример:

```
ecorouter(config)#interface bdi0
ecorouter(config-if)#ip address 192.168.0.1/24
ecorouter(config-if)#rewrite push 20
ecorouter(config-if)#connect bridge br0
```

При такой конфигурации в L3 домен могут попадать фреймы бриджа **br0** с VLAN ID **20**. В обратном направлении пакеты будут направляться в **br0** при условии, что для IP-адреса назначения в FIB указан интерфейс **bdi0**.

11.3 Команды просмотра

Для просмотра информации о созданных бриджах используется команда административного режима **show bridge**. Если необходимо вывести на консоль информацию по какому-то конкретному бриджу, к указанной команде добавляется имя бриджа: **show bridge <BRIDGE_NAME>**.

```
ecorouter#show bridge
Bridge br1
  Connect interface bd1 symmetric
```

Для просмотра информации об интерфейсах BDI используется стандартная для всех интерфейсов команда **show interface <BDI_NAME>**.

```
ecorouter#show interface bd1
Interface bd1 is up
Ethernet address: 1c87.7640.6903
MTU: 1500
Rewrite: push 20
ICMP redirection is on
Label switching is disabled
<UP,BROADCAST,RUNNING,MULTICAST>
Connect bridge br1 symmetric
inet 1.1.1.1/24 broadcast 1.1.1.255/24
total input packets 0, bytes 0
total output packets 0, bytes 0
```

В EcoROuterOS есть возможность посмотреть таблицу mac-адресов по конкретному бриджу.

Для этого необходимо ввести команду **show bridge mac-table <BRIDGE_NAME>**. Эта команда доступна в пользовательском режиме и режиме администрирования.

Данная команда показывает все mac-адреса, которые были изучены в рамках данного бриджа.

```
ecorouter#show bridge mac-table br0
L3 BDI address: 192.168.1.1/24
BD Aging time is 300 sec

Outer   Inner      L2
Vlan    Vlan       Address     Port      Type      Age
-----  -----  -----
-        -          0050.7966.6801  te2       Dynamic    2
30       -          0050.7966.6800  te1       Dynamic    18
20       10         0050.7966.6802  te0       Dynamic    21
```

В приведенном примере показаны следующие параметры и их значения:

L3 BDI address: 192.168.1.1/24 - IP-адрес L3 интерфейса в данном бриdge;

BD Aging time - время устаревания для каждого mac-адреса в секундах;

Outer Vlan - значение внешнего vlan, с которым был подключен пользователь;

Inner Vlan - значение внутреннего vlan, с которым был подключен пользователь;

L2 address - mac-адрес устройства;

Port - название порта, с которого пришел данный mac-адрес;

Type - метод, по которому был изучен mac-адрес (статически или динамически);

Age - время в секундах, когда был зафиксирован последний пакет от данного mac-адреса.

12 Dynamic Host Configuration Protocol

Dynamic Host Configuration Protocol (DHCP)

Протокол динамической настройки адресации узлов сети, позволяющий устройствам внутри сети динамически получать IP-адреса и другие параметры для работы в сети (TCP-IP). Для автоматической конфигурации компьютер-клиент на этапе конфигурации сетевого устройства обращается к так называемому серверу DHCP и получает от него нужные параметры.

Адрес IP выделяется клиенту на определенное время (т.н время аренды). Временные параметры аренды определяются настройками сервера DHCP.

DHCP-сервер – сервер, выдающий параметры настройки TCP/IP.

DHCP-клиент – тот, кто запрашивает настройки TCP/IP.

DHCP-ретранслятор – посредник во взаимодействии клиента и сервера. Ретранслятор используется, когда у клиента нет возможности обратиться к DHCP-серверу напрямую, в частности, когда клиент и сервер располагаются в разных широковещательных доменах.

EcoRouter поддерживает 2 режима ретрансляции: DHCP-relay и DHCP-relay-proxy. В таблице ниже приведены особенности этих режимов.

Таблица 69

Действие или событие	Действие EcoRouter в режиме DHCP-relay	Действие EcoRouter в режиме DHCP-relay-proxy
Клиент послал широковещательное сообщение DISCOVER	EcoRouter перенаправляет широковещательное (multicast) сообщение DISCOVER	EcoRouter перехватывает широковещательное сообщение DISCOVER, вносит в таблицу DHCP mac-адрес и VLAN клиента, после чего перенаправляет сообщение DISCOVER в виде unicast
DHCP-серверы послали сообщения OFFER	EcoRouter перенаправляет сообщения OFFER от всех ответивших DHCP-серверов клиенту	EcoRouter подменяет в сообщении OFFER от первого ответившего DHCP-сервера клиенту адрес ответившего сервера своим адресом, добавляет в таблицу информацию о выданном ip-адресе и времени начала аренды, а остальные сообщения OFFER игнорирует
Клиент послал сообщение REQUEST	EcoRouter перенаправляет широковещательное сообщение REQUEST	EcoRouter подменяет адрес клиента на собственный и перенаправляет сообщение REQUEST выбранному клиентом DHCP-серверу
DHCP-сервер послал сообщения ACK на mac-адрес компьютера, указанного в сообщении REQUEST	EcoRouter перенаправляет сообщение ACK клиенту	EcoRouter перенаправляет сообщение ACK клиенту
Наступил момент для запроса обновления аренды адреса (RENEWING) (определяется настройками DHCP-сервера)	EcoRouter перенаправляет DHCP-серверу сообщение REQUEST от клиента с просьбой продлить срок аренды	EcoRouter самостоятельно направляет DHCP-серверу сообщение REQUEST с просьбой продлить срок аренды. EcoRouter также хранит информацию о времени последнего получения запроса обновления аренды адреса от клиента и

Действие или событие	Действие EcoRouter в режиме DHCP-relay	Действие EcoRouter в режиме DHCP-relay-proxy
		времени получения последнего пакета подтверждения от сервера
Наступил момент для запроса обновления конфигурации (REBINDING) (определяется настройками DHCP-сервера)	EcoRouter перенаправляет широковещательное сообщение REQUEST с текущим сетевым адресом клиента	EcoRouter самостоятельно направляет широковещательное сообщение REQUEST с текущим собственным сетевым адресом

Если опция 82 включена, то в режиме DHCP-relay ее параметры добавляются в запрос REQUEST клиента (описание опции 82 см. ниже).

Опция 82 – опция протокола DHCP, нужная для передачи DHCP-серверу разнообразной информации. Применяется для защиты DHCP-сервера от атак с использованием DHCP-протокола и не является обязательной для использования.

12.1 Список команд

Таблица 70

Команда	Описание
ecorouter(config)#dhcp-profile VALUE	Создание DHCP-профиля, где VALUE - любое численное значение
ecorouter(config-dhcp)#description LINE	Описание созданного профиля, где LINE - любое значение. Необязательная команда
ecorouter(config-dhcp)#mode proxy	Включение режима работы proxy для ретрансирования запросов к серверу. Задание режима работы обязательно
ecorouter(config-dhcp)#mode relay	Включение режима работы relay для ретрансирования запросов к серверу. Задание режима работы обязательно
ecorouter(config-dhcp)#server IP-address	Указание IP-адреса DHCP-сервера. Обязательная команда
ecorouter(config-dhcp)#server IP-address lease VALUE	Указание адреса сервера с возможным временем использования адреса от него в секундах. Значение по умолчанию 3600. Работает только для режима proxy
ecorouter(config-dhcp)#information-option circuit-id LINE	Опция передачи дополнительной информации серверу. Подробнее о параметрах смотри раздел 3. Необязательная команда
ecorouter(config-dhcp)#information-option install	Принудительная установка информационной опции. Необязательная команда
ecorouter(config-dhcp)#information-option remote-id LINE	Опция передачи информации с mac-адресом клиента, который отправил запрос. Необязательная команда
ecorouter(config-dhcp)#information-option rewrite	Перезапись информационной опции. Если circuit-id и remote-id не будут заданы на маршрутизаторе, то опция будет просто удалена из пакета. Необязательная команда
ecorouter(config-if)#dhcp-profile VALUE	Команда привязки созданного профиля к интерфейсу, где VALUE номер созданного профиля

12.2 Базовая настройка

Шаг 1. Создание интерфейса для привязки профиля DHCP-ретранслятора и назначение ip-адреса.

```
ecorouter(config)#interface dhcp1
ecorouter(config-if)#ip add
10.10.10.10/30
```

Шаг 2. Создание DHCP-профиля.

```
ecorouter(config)#dhcp-profile 0
```

Профиль необходим для более гибкой настройки раздачи адресов в разных сегментах сети. К одному интерфейсу можно привязать только один профиль, но один профиль можно привязать к разным интерфейсам. Количество профилей не ограничено.

Шаг 3. Указание адреса DHCP-сервера.

```
ecorouter(config-dhcp)#server 170.200.10.10
```

В одном профиле может быть указано до 8 серверов.

Шаг 4. Указание режима работы ретранслятора.

```
ecorouter(config-dhcp)#mode relay
```

Настройка разных режимов не различается. Выбор режима работы зависит от производительности модели оборудования и решаемых задач.

Шаг 5. Указание параметров опции 82.

```
ecorouter(config-dhcp)#information-option circuit-id Router: %{port} /
client: %{cmac}/%{svlan}.%{cvlan}
ecorouter(config-dhcp)#information-option remote-id Router:
%{hname}/%{vr}
```

Таблица 71

Параметр	Описание
port	номер порта, откуда запрос пришел
cmac	mac-адрес клиентского оборудования
svlan	номер сервисного VLAN'a
cvlan	номер VLAN'a клиента
hname	hostname маршрутизатора, который отправляет пакет на DHCP-сервер
vr	идентификатор виртуального маршрутизатора

На основании перечисленных в таблице данных DHCP-сервер решает, выдавать настройки или нет и может определять, из какого пула адресов выдавать адрес.

Вместо такой записи можно использовать произвольную строку, например:

```
ecorouter(config-dhcp)#information-option circuit-id randomstring
```

которую также необходимо задать на сервере. При успешном сравнении строк сервер примет положительное решение о выдаче адреса.

Можно указывать и параметры, и произвольную строку совместно, например:

```
ecorouter(config-dhcp)#information-option circuit-id Router: %{port} /
client: %{cmac}/%{svlan}
ecorouter(config-dhcp)#information-option remote-id randomstring
```

Задавать **remote-id** возможно только при задании **circuit-id**.

Шаг 6. Привязка созданного профиля к интерфейсу.

```
ecorouter(config)# interface dhcp1
ecorouter(config-if)#dhcp-profile 0
```

12.3 Команды просмотра состояния DHCP

Команда **show dhcp-profile** выводит список всех существующих профилей DHCP и основные их настройки:

```
ecorouter#show
dhcp-profile
DHCP profile 0 is in relay mode
Relay information option (82) is on
Circuit-ID: randomstring
DHCP profile 2 is in proxy mode
Relay information option (82) is on
Circuit-ID: 78
Server 1.1.1.1
Server 4.4.4.4
Server 4.4.4.5
Server 4.4.4.6
Server 4.4.4.7
DHCP profile 3 is in relay mode
Relay information option (82) is on
Circuit-ID: Router: %{hname}, client: %{cmac}/{svlan}.%{cvlan}
```

Для просмотра определенного профиля та же команда даётся с номером профиля, который нужно посмотреть.

```
show dhcp-profile 0
DHCP profile 0 is in relay mode
Relay information option (82) is off
```

Команда **show interface dhcp clients <NAME>** работает только для DHCP-relay-proxy, где <NAME> - имя интерфейса, к которому привязан DHCP-профиль.

Данной командой выводится на экран таблица, содержащая список всех DHCP-клиентов. В таблице содержатся записи с выданными IP-адресами, mac-адресами клиентов, адрес DHCP-сервера, выдавшего настройку, время подтверждения, время, на которое адрес был выдан.

```
ecorouter#sh interface dhcp clients demux.0
IP Address MAC Address DHCP Server ACK Time Lease Time
-----
192.168.1.3 c403.130f.0000 20.0.0.1 296 86400
```

13 ARP

ARP (Address Resolution Protocol – протокол определения адреса) – протокол в компьютерных сетях, предназначенный для определения MAC-адреса по известному IP-адресу.

В маршрутизаторе данный протокол включен по умолчанию и дополнительных настроек не требует. Реализация протокола в EcoRouterOS позволяет хранить как динамические записи, полученные при помощи широковещательных запросов, так и статические записи.

Функция проксирования, известная как Proxy ARP, не поддерживается.

Функционал протокола настраивается в конфигурационном режиме при помощи команд, представленных в таблице ниже.

Таблица 72

Команда	Описание
arp <IP ADDRESS> <MAC ADDRESS>	Создание статической записи для конкретного IP-адреса
arp expiration-period <0-300>	Настройка времени хранения динамической записи в ARP-таблице в минутах. Значение по умолчанию - 5 минут
arp incomplete-time <5-300>	Настройка времени хранения incomplete записи в ARP-таблице в секундах. Значение по умолчанию - 60 секунд
arp request-interval <0-100>	Задание интервала времени отправки ARP-запросов в секундах в случае отсутствия ARP-ответов. Значение по умолчанию - 1 секунда
arp request-number <0-100>	Задание количества отправляемых ARP-запросов при отсутствии ARP-ответов. Значение по умолчанию - 3

Для просмотра таблицы ARP-записей следует воспользоваться командой административного режима **show arp**. В качестве аргументов можно передать различные параметры, перечисленные ниже.

Таблица 73

Команда	Описание
show arp	Вывод полной ARP-таблицы
show arp interface <INTERFACE NAME>	Вывод ARP-таблицы для записей, полученных с определенного интерфейса
show arp ip <IP ADDRESS>	Вывод ARP-записи для определенного IP-адреса
show arp mac <MAC ADDRESS>	Вывод ARP-записей для определенного MAC-адреса

Пример создания статической ARP-записи и вывода ARP-таблицы (стрелки около названий интерфейсов указывают на локально созданные интерфейсы маршрутизатора).

```
ecorouter(config)#arp 10.12.0.100 ca0b.3b18.001d
ecorouter(config)#exit
ecorouter#show arp
Interface  IP Address      MAC Address      Type      Age
-----
>eth2      200.22.0.200    1c87.7640.0507   -----   -----
>eth1      100.24.0.200    1c87.7640.0506   -----   -----
>eth3      10.12.0.200     1c87.7640.0505   -----   -----
eth3       10.12.0.100     ca0b.3b18.001d   static   -----
```

```
|eth3      10.12.0.1      ca0b.3b18.001c dynamic  3
```

Сконфигурированные настройки можно посмотреть командой **show arp settings**.

14 Настройка IP Demux

Технология демультиплексирования входящего потока данных со стороны глобальной сети на один или более выходных потоков в направлении локальных сетей. Выбор желаемого выхода осуществляется на основании сконфигурированных сервисных интерфейсов на портах устройства. Для полноценного функционирования технология предполагает наличие сформированной таблицы с информацией о расположении клиентов в сети. Подобная информация может быть получена динамическим и статическим путем. Под динамикой, в данном контексте, подразумевается способность маршрутизатора получать всю необходимую информацию о клиентах при DHCP перенаправлениях к серверу. Подобный метод не подразумевает статическую настройку IP адреса на клиентских машинах. Однако, для полноценного контроля, доступности сетевых элементов и полной независимости от удаленных серверов, в арсенале сетевого администратора присутствует способ создания статической записи о клиенте.

- Интерфейс IP demux является интерфейсом третьего уровня
- К интерфейсу IP demux может быть присоединено несколько сервисных интерфейсов на одном или нескольких физических портах
- IP demux имеет таблицу соответствия клиентских IP адресов, VLAN'ов и портов. Таблица может формироваться динамическим и статическим путем
- При передаче кадра с меткой VLAN на интерфейс IP demux метка снимается автоматически и дополнительных операций над меткой не требуется

Интерфейс IP demux – это виртуальный L3 интерфейс, на который может быть назначен IP-адрес из маршрутизуемой подсети. Пересылка пакетов в другие подсети будет осуществляться за счёт привязки к определенному порту с набором service instance.

Базовая настройка интерфейса IP demux:

Таблица 74

Команда	Описание
interface demux.<NAME>	Создание интерфейса demux. Где <NAME> – произвольное число
ip address <IP>/<MASK>	Назначение IP-адреса с префиксом

Пример:

```
ecorouter(config)#interface demux.0  
ecorouter(config-if-demux)#ip address 10.10.10.1/24
```

Работа динамической версии IP demux реализуется при наличии DHCP-сервера в сети. Таблица соответствия IP-адресов, VLAN'ов и портов формируется на основании сетевых настроек, которые запрашивают конечные устройства у DHCP-сервера. На интерфейсе IP demux необходимо указать созданный профиль DHCP-ретрансляции. При такой организации конечные устройства за интерфейсом demux будут иметь доступ к шлюзу и глобальной сети соответственно, но возможность общения между VLAN исключена.

14.1 Пример настройки IP Demux

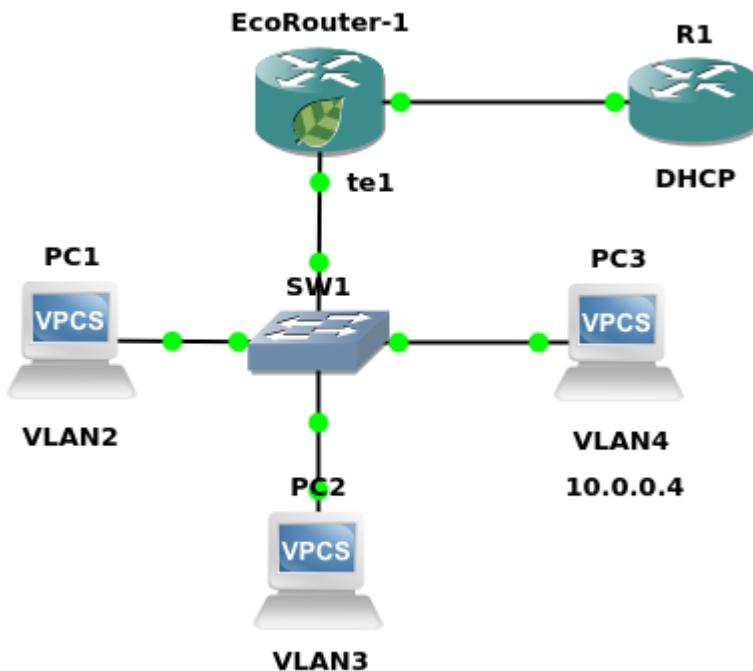


Рисунок 25

Шаг 1. Создание интерфейса demux и назначение адреса

```
ecorouter(config)#interface demux.0
ecorouter(config-demux)#ip add 10.0.0.254/30
```

Шаг 2. Создание DHCP-профиля, указание режима работы и адреса DHCP-сервера

```
ecorouter(config)#dhcp-profile 0
ecorouter(config-dhcp)#mode proxy
ecorouter(config-dhcp)#server 1.100.100.1
```

Подробнее о настройке DHCP см. статью DHCP-ретрансляция.

Шаг 3. Присоединение DHCP-профиля к интерфейсу demux

```
ecorouter(config)#interface demux.0
ecorouter(config-demux)#set dhcp 0
```

На одном demux интерфейсе может быть привязка к одному профилю DHCP.

Шаг 4. Создание сервисного интерфейса на порту (см. статью Сервисные интерфейсы)

```
ecorouter(config)#port tel
ecorouter(config-port)#service-instance 1
```

Шаг 5. Указание номеров или диапазона обрабатываемых VLAN

```
ecorouter(config-service-instance)#encapsulation dot1q 1-3 exact
```

Шаг 6. Привязка сервисного интерфейса к интерфейсу demux

```
ecorouter(config-service-instance)#connect ip interface demux.0
```

Также в этой же схеме реализована работа статической версии IP demux для конечного устройства PC3, которое работает со статическим адресом.

Шаг 7. Создание сервисного интерфейса для операций с VLAN конечного устройства.

```
ecorouter(config)#port tel  
ecorouter(config-port)#service-instance 1.4  
ecorouter(config-service-instance)#encapsulation dot1q 4 exact
```

Шаг 8. Присоединение к интерфейсу demux.

```
ecorouter(config-service-instance)# connect ip interface demux.0
```

Шаг 9. Добавление записи в таблицу интерфейса demux.

```
ecorouter(config-if)#ip demux 10.0.0.4/32 port tel service-instance 1.4  
push 4
```

Данной записью добавили клиента со статическим адресом в таблицу интерфейса demux. В команде **ip demux** на первом месте стоит аргумент с ip адресом конечного устройства, далее следует указание на порт, на котором настроен сервисный интерфейс, обрабатывающий данный VLAN. На последнем месте указывается метка VLAN, которую необходимо поместить в пакет.

14.2 Команды просмотра IP Demux

Просмотр содержимого таблицы интерфейса выполняется с помощью команды **show interface demux clients demux.NAME**.

Пример выполнения показан ниже.

```
ecorouter#sh interface demux clients demux.0  
IP Address MAC Address Port C-tag S-tag WAN packets LAN packets WAN  
bytes LAN bytes  
-----  
-----  
10.0.0.1 c403.130f.0000 <4> ----- 0  
0 0 0
```

15 Мультикаст

Без мультикастового вещания для успешной передачи данных различным пользователям трафик в сети должен дублироваться на каждом узловом участке. Такое дублирование приводит к неэффективному использованию ресурсов сети. Multicast-приложения являются гораздо более эффективными, так как передают только один экземпляр трафика. Его дублирование обычно происходит только в L3-устройствах, расположенных ближе к потребителям. Для решения задач доставки/приема мультикастовых данных EcoRouterOS поддерживает работу следующих протоколов:

- IGMPv1/v2/v3,
- PIM-SM,
- PIM-SSM.

Инструкции по настройке протоколов доступны в документации. В данном документе содержатся краткие описания нескольких специфичных технологий, которые поддерживаются маршрутизатором для более тонкой настройки мультикастового домена при отсутствии нужной функциональности в оборудовании других производителей:

- IGMP SSM Mapping для возможности доставки/приема мультикастовых потоков с определенного сервера при IGMPv2;
- IGMP proxy для создания IGMP домена между L2/L3 устройствами и работы маршрутизатора в качестве клиента мультикастовой группы;
- PIM-DM поддержка более раннего протокола мультикастовой маршрутизации;
- PIM-SDM смешанный режим работы.

Описание по настройке этих расширений можно найти в соответствующих главах.

15.1 IGMP

IGMP (Internet Group Management Protocol) – протокол управления групповой передачей данных в IP-сетях. IGMP используется клиентским компьютером и локальным маршрутизатором, осуществляющим групповую передачу. В EcoRouter поддерживаются 1-3 версии протокола.

Список команд, использующихся для настройки протокола IGMP в EcoRouter, представлен в таблице ниже.

Таблица 75

Команда	Режим	Описание
ip igmp access-group <номер списка доступа>	(config-if)#	Фильтрация доступа к определенным мультикаст-группам с помощью списков доступа
ip igmp immediate-leave group-list <номер списка фильтров>	(config-if)#	Команда сокращения времени отписки последнего клиента от группы/групп, заданных в списке фильтрации
ip igmp join-group <ip-адрес>	(config-if)#	Команда добавления интерфейса маршрутизатора в мультикаст-группу

Команда	Режим	Описание
ip igmp last-member-query-count <2-7>	(config-if)#	Настройка количества IGMP query сообщений, отправляемых в ответ на сообщение типа leave. По умолчанию 2
ip igmp last-member-query-interval <1000-25500>	(config-if)#	Настройка интервала отправки IGMP query сообщений. По умолчанию 1000 мс
ip igmp limit <1-2097152>	(config)#	Настройка ограничений количества мультикаст-маршрутов
ip igmp mroute-proxy <имя интерфейса>	(config-if)#	Включение проксирования для мультикаст маршрутов на другой интерфейс
ip igmp proxy unsolicited-report-interval <1000-25500>	(config-if)#	Задание значения задержки между двумя IGMP join сообщениями. По умолчанию 1000 мс
ip igmp proxy-service	(config-if)#	Включение режима IGMP proxy
ip igmp querier-timeout <60-300>	(config-if)#	Задание времени до перевыборов querier маршрутизатора в сегменте в секундах
ip igmp query-interval <1-18000>	(config-if)#	Задание частоты отправки General Query. По умолчанию 125 с
ip igmp query-max-response-time <1-240>	(config-if)#	Задание максимального значения времени ответа на IGMP query в секундах. По умолчанию 10 с
ip igmp robustness-variable <2-7>	(config-if)#	Задание числа для тонкой настройки IGMP сообщений. По умолчанию 2
ip igmp startup-query-count <2-10>	(config-if)#	Задание количества query сообщений. По умолчанию 2
ip igmp startup-query-interval <1-18000>	(config-if)#	Настройка интервала отправки IGMP query сообщений. По умолчанию 31 с
ip igmp static-group <ip-адрес>	(config-if)#	Назначение интерфейса устройства на прослушивания определенной мультикаст-группы
ip igmp version <1-3>	(config-if)#	Выставление версии IGMP
ip igmp ssm-map {enable static <номер списка доступа>}	(config)#	Включение SSM-маппирования. Задание статического SSM с помощью списка доступа
ip igmp tos-check	(config)#	Проверка значения поля TOS. Включена по умолчанию
ip igmp vrf <имя виртуального маршрутизатора> {limit <1-2097152> ssm-map enable ssm-map static <номер списка доступа>}	(config)#	Команды настройки для выполнения в виртуальном маршрутизаторе
ip igmp ra-option	(config-if)#	Включает проверку опции во входящих IGMP-пакетах

Настройка IGMP в сегменте с настроенным PIM сводится к включению IGMP на интерфейсе маршрутизатора, ближайшего к пользователю. Включение осуществляется с помощью команды на настроенном нисходящем интерфейсе **ip igmp version <1-3>**.

Шаг 1. Включение глобальной поддержки мультикаста.

```
ecorouter(config)#ip multicast-routing
```

Шаг 2. Настройка интерфейсов устройства.

```
ecorouter(config)#interface e10
ecorouter(config-if)#ip address 10.10.10.1/24
ecorouter(config)#port te0
ecorouter(config-port)#service-instance 10
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface e10
```

Шаг 3. Включение IGMP на нынешнем интерфейсе.

```
ecorouter(config-if)#ip igmp version 2
```

При включении PIM на интерфейсе IGMPv3 включается автоматически.

Шаг 4. Настройка таймеров протокола: частоты рассылки запросов устройством и времени ожидания ответов.

```
ecorouter(config-if)#ip igmp query-interval 100
```

```
ecorouter(config-if)#

ip igmp query-max-response-time 20
```

Шаг 5. Для корректной работы со всем спектром ОС необходимо отключать проверку значения поля ToS в сообщениях IGMP report.

```
ecorouter(config)#no ip igmp tos-check
```

15.2 IGMP SSM Mapping

Для поддержки SSM необходима функциональность IGMPv3, однако не все оборудование в сети поддерживает все версии этого протокола. EcoRouterOS позволяет выполнить маршрутизацию мультикастового трафика от специфичного источника до клиентов, которые поддерживают только вторую версию IGMP протокола. Ниже приведен пример настройки:

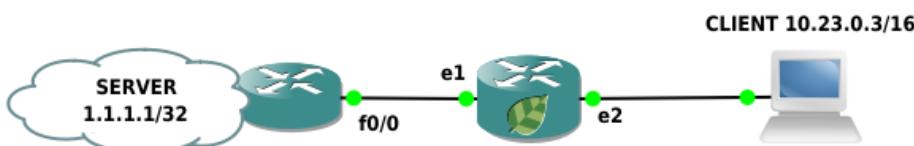


Рисунок 26

Шаг 1. Настройка портов, интерфейсов и сервисных интерфейсов.

```
ecorouter(config)#interface e1
ecorouter(config-if)#ip address 10.12.0.2/16
ecorouter(config)#interface e2
ecorouter(config-if)#ip address 10.23.0.2/16
ecorouter(config)#port ge1
ecorouter(config-port)#service-instance ge1/e1
```

```
ecorouter(config-service-instance)#connect ip interface e1
ecorouter(config)#port ge2
ecorouter(config-port)#service-instance ge2/e2
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface e2
ecorouter(config)#port ge2
ecorouter(config-port)#service-instance ge2/e2
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface e2
```

Шаг 2. Задание policy-filter-list для определенной группы.

```
ecorouter(config)#policy-filter-list 2 permit 235.7.7.7
```

Шаг 3. Включение SSM-mapping для определенной группы.

```
ecorouter(config)#ip igmp ssm-map enable
ecorouter(config)#ip igmp ssm-map static 2 1.1.1.1
ecorouter(config)#ip pim ssm default
```

Шаг 4. Настройка PIM-SM.

```
ecorouter(config)#ip pim rp-address 10.12.0.2
ecorouter(config)#interface e1
ecorouter(config-if)#ip pim sparse-mode
ecorouter(config-if)#interface e2
ecorouter(config-if)#ip pim sparse-mode
```

На интерфейсе fa0/0 другого маршрутизатора настроен IP адрес 10.12.0.1/16. Теперь если клиент запросит группу 235.7.7.7 одновременно с отправкой мультикастового трафика с сервера и с маршрутизатора на эту группу, то на маршрутизаторе можно наблюдать следующую картину:

```
Ecorouter#show ip mroute
IP Multicast Routing Table
Flags: I - Immediate Stat, T - Timed Stat, F - Forwarder installed
B - BIDIR
Timers: Uptime/Stat Expiry Interface State: Interface (TTL)
(1.1.1.1, 235.7.7.7), uptime 00:04:24, stat expires 00:03:29
Owner PIM, Flags: TF
    Incoming interface: e1
    Outgoing interface list:
        e2 (1)
(10.12.0.1, 235.7.7.7), uptime 00:04:24, stat expires 00:00:09
Owner PIM, Flags: TF
    Incoming interface: e1
    Outgoing interface list:
```

Как видно, интерфейсов в списке outgoing для сервера 10.12.0.1 нет. При включении на интерфейсе протокола PIM командой **ip pim sparse-mode**, IGMPv3 включается по умолчанию. Можно было просто включить IGMPv3 отдельно от PIM командой **ip igmp version 3**. Полезная команда для просмотра информации по статическому маппингу **show ip igmp ssm-map <ip-адрес>**:

```
ecorouter#show ip igmp ssm-map 235.7.7.7
Group address: 235.7.7.7
Database      : Static
Source list   : 1.1.1.1
```

15.3 Proxy-IGMP

Использование этой технологии позволит избежать зависимости от используемого протокола мультикастовой маршрутизации и уменьшить размер служебного трафика в сети. Маршрутизатор выступает в роли клиента и передает информацию в виде сообщений IGMP Report в сторону PIM-домена. PIM-соседи в таком случае не нужны. Устройство хранит информацию о запрошенных группах, полученную через нисходящие интерфейсы, в базе данных. Сам прокси-сервис работает на восходящих интерфейсах, передавая запросы от клиентов. Ниже приведен пример топологии и конфигурирования IGMP Proxy сервиса в EcoRouterOS.

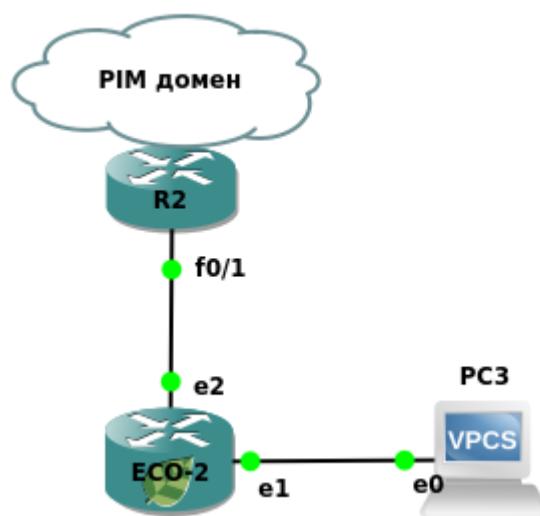


Рисунок 27

15.3.1 Настройка

Шаг 1. Задание имени устройства и включение мультикастовой маршрутизации.

```
(config) #hostname ECO-2
(config) #ip multicast-routing
```

Шаг 2. Настройка портов, интерфейсов и сервисных интерфейсов.

```
(config) #interface e1
(config-if) #ip address 10.23.0.2/16
(config-if) #ip igmp version 2
(config) #interface e2
(config-if) #ip address 10.24.0.2/16
(config-if) #ip igmp version 2
(config) #port ge1
(config-port) #service-instance ge1/e1
(config-service-instance) #encapsulation untagged
(config-service-instance) #connect ip interface e1
(config) #port ge2
(config-port) #service-instance ge2/e2
(config-service-instance) #encapsulation untagged
(config-service-instance) #connect ip interface e2
```

Шаг 3. Включение IGMP Proxy.

```
(config) #interface e2
(config-if) #ip igmp proxy-service
(config) #interface e1
(config-if) #ip igmp mrouter-proxy e2
```

Прокси-сервис работает с любой версией IGMP. Для проверки статуса сервиса и просмотра запрошенных групп используются команды **show ip igmp proxy** и **show ip igmp proxy groups**. Если сервис запущен и работает, то статус группы должен быть «Active».

15.4 PIM-SM/SSM

Тонкая настройка протоколов мультикастовой маршрутизации довольно сложна и не рассматривается в данном документе. Для базовой настройки необходимо выполнить следующие действия.

Шаг 1. Включение мультикастовой маршрутизации командой конфигурационного режима **ip multicast-routing**.

Шаг 2. Включение протокола мультикастовой маршрутизации на нужных интерфейсах контекстной командой **ip pim sparse-mode**. При введении этой команды на интерфейсе автоматически включается протокол IGMPv3.

Шаг 3. Статическое задание точки встречи деревьев от источника и клиентов (Rendezvous Point, далее – RP) командой **ip pim rp-address <IP> [<POLICY-FILTER-LIST>] [override]**. Здесь с помощью номера **<POLICY-FILTER-LIST>** можно привязать RP к определенной мультикастовой группе, а параметр **override** повышает приоритет статической записи о RP по сравнению с полученной динамическим путем. Динамический путь описан ниже.

Шаг 4. Добавление возможности переключения на более короткий маршрут до источника при помощи команды **ip pim spt-threshold [group-list <POLICY-FILTER-LIST>]**, где номер **policy-filter-list** указывает конкретные мультикастные группы.

Этих шагов достаточно для успешной доставки мультикаст-трафика от сервера до клиентов, однако при выходе из строя RP все клиенты перестанут получать запрашиваемые данные.

Поэтому предпочтение отдается протоколу bootstrap, который динамически информирует участников мультикастового домена о RP.

Таким образом, на 4 шаге для информирования PIM-соседей о RP необходимо сконфигурировать кандидата на эту роль командой конфигурационного режима **ip pim rp-candidate <название интерфейса> [priority <0-255>] [group-list <POLICY-FILTER-LIST>] [interval <1-16383>]**. Параметры команды описаны в таблице ниже.

Таблица 76

Параметр	Описание
<название интерфейса>	Интерфейс, назначаемый кандидатом. Интерфейс должен быть предварительно создан в системе
priority	Приоритет, при задании нескольких кандидатов. Чем меньше значение данного параметра, тем выше приоритет кандидата. Допустимые значения от 0 до 255. Значение по умолчанию 192
group-list <POLICY-FILTER-LIST>	Группы, которым рассыпается реклама о кандидате

Параметр	Описание
interval	Интервал рассылки сообщений в секундах. Допустимые значения от 1 до 16383

Далее необходимо сконфигурировать рекламных агентов, которые будут рассыпать информацию о RP, так называемых BSR, командой конфигурационного режима **ip pim bsr-candidate <название интерфейса> [<0-32>][<0-255>]**. Параметры команды описаны в таблице ниже.

Таблица 77

Параметр	Описание
<название интерфейса>	Интерфейс, назначаемый рекламным агентом (BSR). Интерфейс должен быть предварительно создан в системе
<0-32>	Длина хэш-маски для расчета хэш-значения RP. Допустимые значения от 0 до 32. Значение по умолчанию 10
<0-255>	Приоритет BSR, при наличии нескольких агентов в сети. Чем больше значение данного параметра, тем выше приоритет кандидата. Допустимые значения от 0 до 255. Значение по умолчанию 64

Ниже приведен пример схемы и конфигурирования маршрутизаторов. При мультикаст-вещании со стороны сервера Multicast-1 маршрут протекания трафика будет ECO-3 – ECO-2 – ECO-4 – PC1, а после того, как ближайший к клиенту маршрутизатор получит информацию о сервере, произойдет SPT switchover – маршрут поменяется на ECO-3 – ECO-4 – PC1.

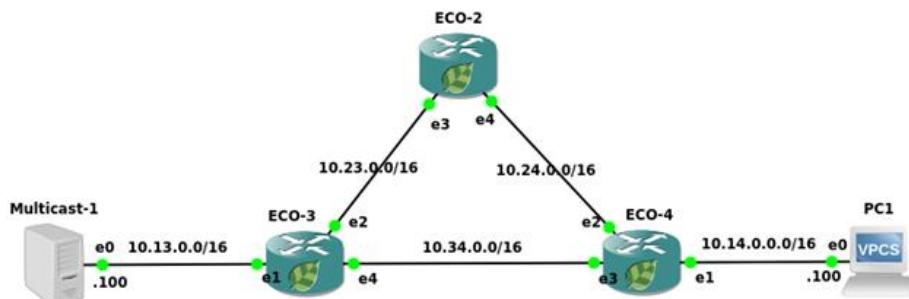


Рисунок 28

Шаг 1. Задание имени устройства и включение мультикастовой маршрутизации.

```
ecorouter(config)#hostname ECO-2
ecorouter(config)#ip multicast-routing
```

Шаг 2. Настройка портов, интерфейсов и сервисных интерфейсов.

```
ecorouter(config)#interface e3
ecorouter(config-if)#ip address 10.23.0.3/16
ecorouter(config-if)#ip pim sparse-mode
ecorouter(config)#interface e4
ecorouter(config-if)#ip address 10.24.0.2/16
ecorouter(config-if)#ip pim sparse-mode
ecorouter(config)#port ge3
ecorouter(config-port)#service-instance ge3/e3
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface e3
ecorouter(config)#port ge4
ecorouter(config-port)#service-instance ge4/e4
ecorouter(config-service-instance)#encapsulation untagged
```

```
|ecorouter(config-service-instance)#connect ip interface e4
```

Шаг 3. Включение маршрутизации.

```
ecorouter(config)#router isis
ecorouter(config-router)#net 49.0001.0000.0000.0003.00
ecorouter(config-router)#exit
ecorouter(config)#interface e3
ecorouter(config-int)#ip router isis
ecorouter(config-int)#interface e4
ecorouter(config-int)#ip router isis
ecorouter(config-int)#exit
```

Шаг 4. Задание информации о RP и включение возможности SPT-switchover.

```
ecorouter(config)#ip pim bsr-candidate e3
ecorouter(config)#ip pim rp-candidate e3 priority 20
ecorouter(config)#ip pim spt-threshold
```

Конфигурация оставшихся маршрутизаторов будет аналогичной.

```
ecorouter(config)#hostname ECO-3
ecorouter(config)#ip multicast-routing
ecorouter(config)#interface e1
ecorouter(config-if)#ip address 10.13.0.3/16
ecorouter(config-if)#ip router isis
ecorouter(config-if)#ip pim sparse-mode
ecorouter(config)#interface e2
ecorouter(config-if)#ip address 10.23.0.3/16
ecorouter(config-if)#ip router isis
ecorouter(config-if)#ip pim sparse-mode
ecorouter(config)#interface e4
ecorouter(config-if)#ip address 10.34.0.3/16
ecorouter(config-if)#ip router isis
ecorouter(config-if)#ip pim sparse-mode
ecorouter(config)#port ge1
ecorouter(config-port)#service-instance ge1/e1
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface e1
ecorouter(config)#port ge2
ecorouter(config-port)#service-instance ge2/e2
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface e2
ecorouter(config)#port ge4
ecorouter(config-port)#service-instance ge4/e4
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface e4
ecorouter(config)#router isis
ecorouter(config-router)#net 49.0001.0000.0000.0003.00
ecorouter(config)#hostname ECO-4
ecorouter(config)#ip multicast-routing
ecorouter(config)#ip pim spt-threshold
ecorouter(config)#ip pim bsr-candidate e3
ecorouter(config)#ip pim rp-candidate e3 priority 40
ecorouter(config)#interface e1
ecorouter(config-if)#ip address 10.14.0.4/16
ecorouter(config-if)#ip router isis
```

```

ecorouter(config-if)#ip pim sparse-mode
ecorouter(config-if)#ip igmp version 2
ecorouter(config)#interface e2
ecorouter(config-if)#ip address 10.24.0.4/16
ecorouter(config-if)#ip router isis
ecorouter(config-if)#ip pim sparse-mode
ecorouter(config)#interface e3
ecorouter(config-if)#ip address 10.34.0.4/16
ecorouter(config-if)#ip router isis
ecorouter(config-if)#ip pim sparse-mode
ecorouter(config)#port ge2
ecorouter(config-port)#service-instance ge2/e2
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface e2
ecorouter(config)#port ge4
ecorouter(config-port)#service-instance ge4/e4
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface e4
ecorouter(config)#router isis
ecorouter(config-router)#net 49.0001.0000.0000.0003.00

```

Подробнее о IGMP можно прочитать в соответствующем разделе.

Для включения Source-Specific-Multicast требуется ввести дополнительную команду **ip pim ssm {default | range} <номер policy-filter-list>**, где **default** означает применить ко всем группам, а **range** и **номер policy-filter-list** позволяют выделить конкретные группы, для которых будет использоваться SSM. Подробнее о настройке SSM-mapping и policy-filter-list читайте в соответствующих разделах.

Дополнительные команды конфигурирования

Таблица 78

Команда	Режим	Описание
ip pim accept-register <policy-filter-list>	(conf)#	Указывает RP принимать Register сообщения от определенных источников
ip pim cisco-register-checksum	(conf)#	Опция для расчета checksum в Register сообщениях. Для совместимости с более старыми версиями Cisco IOS
ip pim ignore-rp-set-priority	(conf)#	Используется для игнорирования приоритета RP, чтобы полагаться только на хеш-алгоритм
ip pim jp-timer <1-65535>	(conf)#	Тайминг для отправки сообщений Join и Prune
ip pim register-rate-limit <1-65535>	(conf)#	Управление количеством отправляемых Register сообщений
ip pim register-gr-reachability	(conf)#	Включение проверки RP доступности на маршрутизаторе (по умолчанию в конфигурации)
ip pim register-source <адрес>	(conf)#	Задание адреса в Register сообщениях
ip pim register-suppression <1-65535>	(conf)#	Изменение RP-keepalive-timer, если команда ip pim rp-register-kat не задана
ip pim rp-register-kat <1-65535>	(conf)#	Изменение таймеров для мониторинга Register сообщений
ip pim dr-priority	(conf-int)#	Приоритет маршрутизатора для выбора DR

Команда	Режим	Описание
ip pim bsr-border	(conf-int)#	Пометить интерфейс как граничный, для отмены передачи/приема bootstrap
ip pim exclude-genid	(conf-int)#	Исключение опции generated ID
ip pim hello-holdtime <1-65535>	(conf-int)#	Установка таймера holdtime для сообщений hello
ip pim hello-interval <1-18724>	(conf-int)#	Установка таймера interval для сообщений hello
ip pim neighbor-filter <policy-filter-list>	(conf-int)#	Установка соседств с конкретными маршрутизаторами
ip pim propagation-delay <1000-5000>	(conf-int)#	Установка задержки распространения сообщений
ip pim unicast-bsm	(conf-int)#	Включение unicast bootstrap сообщений. Для совместимости с более старыми версиями Cisco IOS
ip pim sparse-mode passive	(conf-int)#	Включение пассивного режима
ip multicast ttl-threshold <1-255>	(conf-int)#	Включение TTL-scope мультикастового домена
ip mroute <адрес подсети> <grpf сосед>	(conf)#	Статическая запись о подсети, в которой находится источник мультикаста

15.4.1 Команды просмотра

Таблица 79

Команда	Описание
show ip mroute	Таблица мультикастовой маршрутизации
show ip mvif	Информация о созданных виртуальных интерфейсах, которые поддерживают мультикаст
show ip rpf <адрес источника>	Отображение RPF информации о источнике
show ip pim bsr-router	Информация о BSR маршрутизаторах в домене
show ip pim interface	Информация об интерфейсах, на которых включена мультикастовая маршрутизация
show ip pim local-members	Локальная информация о запрошенных группах
show ip pim mroute [detail]	Детальная информация по мультикастовой маршрутизации
show ip pim neighbor	Информация о соседских отношениях
show ip pim nexthop	Информация о RP, источниках многоадресной рассылки, интерфейсах через которые получены данные
show ip pim rp mapping	Информация о RP в домене
show ip pim rp-hash <адрес группы>	Информация о RP для конкретной группы
show ip mroute count	Вывод статистической информации

15.4.2 Команды сброса данных

```
clear ip mroute statistics </адрес группы>
clear ip mroute </адрес группы>
clear ip pim sparse-mode bsr rp-set *
```

15.5 PIM-DM and mixed Sparse-Dense mode

EcoRouterOS поддерживает более ранний протокол мультикастовой маршрутизации PIM-DM. Механизм его работы подразумевает излишнее заполнение домена мультикастовым трафиком, поэтому сетевым инженерам необходимо тщательно продумать пути протекания пакетов по сети. Возможно, потребуется отделить домены юникастовой маршрутизации от мультикастовой. В данном случае следует воспользоваться статической записью о маршруте до источника. Для включения функционала на маршрутизаторе достаточно одной команды в режиме конфигурирования интерфейсов - **ip pim dense-mod**.

В EcoRouterOS существует расширение, которое позволяет задать смешанный Sparse-Dense режим на интерфейсе. В этом режиме трафик для группы, идущий по Dense-режиму, будет обработан по правилам PIM-DM, а трафик для группы, идущий по Sparse-режиму, будет обработан по правилам PIM-SM. Для того чтобы включить смешанный режим работы, необходимо в режиме конфигурирования интерфейсов ввести команду **ip pim sparse-dense-mode**.

Для определенных групп можно настроить обработку трафика исключительно PIM-DM логикой. Для этого используется команда **ip pim dense-group <адрес группы>**.

16 Multiprotocol Label Switching

MPLS (multiprotocol label switching – многопротокольная коммутация по меткам) – механизм, осуществляющий передачу данных от одного узла сети к другому с помощью меток.

Каждому пакету, проходящему через MPLS-сеть, независимо от типа этого пакета, назначается определенная метка, на основе которой принимается решение о маршрутизации. Содержимое пакетов при этом не изучается.

Маршрутизаторы в сети MPLS разделяются по своему функционалу на граничные (Label Edge Router, LER) и промежуточные (Label Switch Router, LSR) маршрутизаторы, на которых происходит смена меток.

В таблице ниже представлены основные команды, необходимые для настройки MPLS на EcoRouter.

Таблица 80

Команда	Описание
mpls ac-group <имя> <номер>	Создание новой группы каналов доступа
mpls bandwidth-class	
mpls disable-all-interfaces	Отключение MPLS на всех интерфейсах
mpls egress-ttl <0-255>	Задание значения TTL для выхода с маршрутизатора
mpls enable-all-interfaces	Включение MPLS на всех интерфейсах
mpls ftn-entry <ip- префикс> <метка> <ip-адрес ожидающего интерфейса> <имя исходящего интерфейса>	Настройка метки для FEC при входе в MPLS облако
mpls ilm-entry <приходящая метка> <имя входного интерфейса> swap <исходящая метка> <имя исходящего интерфейса> <ip-адрес ожидающего интерфейса> <ip- префикс>	Настройка замены метки для FEC при транзите через LSR
mpls ingress-ttl <0-255>	Задание значения TTL при входе на маршрутизатор
mpls ldp <max-label-value min-label-value>	Задание диапазона значений выдаваемых меток. Возможные значения от 16 до 1048575
mpls lsp-tunneling <имя входного интерфейса> <приходящая метка> <исходящая метка> <ip- префикс>	
mpls map-route <ip- префикс ip- префикс/маска> <ip- префикс>	
mpls propagate-ttl	Управление переносом значения TTL из IP в MPLS
mpls l2-circuit <имя> <ID> <ip- префикс>	Создание l2-circuit 5 типа
mpls l2-circuit <имя> <ID> <ip- префикс> mode tagged svlan <VLAN> tpid <TPID>	Создание l2-circuit 4 типа

16.1 Настройка статического MPLS

Статический MPLS позволяет вручную настроить все операции с метками на маршрутизаторе. Для хранения используются таблицы ILM и FTN. Настройки правила ILM применяются для проведения операций замены метки внутри домена MPLS. Настройки

правила FTN применяются для навешивания или срезания метки на граничном маршрутизаторе домена MPLS.

Задание правила ILM. Где 1111 – метка, которая ожидается на интерфейсе e1; 2222 – новое значение метки и отправка ее через интерфейс e2; 10.0.0.1 – адрес следующего маршрутизатора(nexthop), а 2.2.2.2/32 – FEC.

```
ecorouter(config)#mpls ilm-entry 1111 e1 swap 2222 e2 10.0.0.1  
2.2.2.2/32
```

Для explicit-null и implicit-null выходящие метки должны быть 0 и 3, соответственно.

Задание правила FTN. Где 2.2.2.2/32 – FEC; 2222 – метка, которая будет навешана; 10.0.0.2 – адрес следующего маршрутизатора(nexthop); e1 – интерфейс для отправки.

```
ecorouter(config)#mpls ftn-entry 2.2.2.2/32 2222 10.0.0.2 e1
```

16.2 LDP

LDP (Label Distribution Protocol) – протокол распределения меток. Метки генерируются для всех маршрутов в таблице маршрутизации. Все локальные метки хранятся в LIB. Метки распространяются в направлении от Egress LER к Ingress LER. В зависимости от настроек распространение меток может происходить либо в режиме Downstream Unsolicited – распространение меток сразу всем соседним маршрутизаторам, либо Downstream-on-Demand – распространение меток по запросу. Соответствие между меткой и сетью отправляется всем соседям LDP.

Настройка LDP

Для начала обмена метками между маршрутизаторами необходимо настроить работу протокола LDP и включить функцию работы с метками на интерфейсах в сторону соседнего MPLS маршрутизатора.

Переход в режим настройки и активация протокола LDP.

```
ecorouter(config)#router ldp
```

При изменении у FEC (Forwarding equivalence class) адреса next-hop (адрес следующего маршрутизатора) маршрутизатор генерирует для этого FEC новую метку и сообщает ее своим соседям. Для того чтобы маршрутизатор использовал одну и ту же метку для одного FEC при изменении адреса next-hop, необходимо включить данную опцию в режиме конфигурации протокола LDP.

```
ecorouter(config)#ldp label preserve
```

Метка сохраняется 30 секунд. Поэтому для корректной работы данной опции смена next-hop должна быть произведена за меньшее время.

Определение транспортного адреса маршрутизатора (необязательный параметр).

```
ecorouter(config-router)#transport-address ipv4 <ip-address>
```

Включение LDP и функции работы с метками на интерфейсах.

```
ecorouter(config-if)#ldp enable ipv4  
ecorouter(config-if)#label-switching
```

Просмотр информации о LDP-соседстве.

```
ecorouter#sh mpls ldp neighbor
```

Команды просмотра

Для просмотра конфигурации и статуса протокола LDP используются команды, представленные в таблице ниже.

Таблица 81

Команда	Описание
show ldp adjacency	Список LDP-связности
show ldp advertise-labels	Просмотр информации о метках
show ldp downstream	Просмотр распространение меток по методу downstream
show ldp upstream	Просмотр распространение меток по методу upstream
show ldp fec	Информация о Forwarding Equivalence Class
show ldp fec-ipv4	Информация о Forwarding Equivalence Class
show ldp graceful-restart	Статус механизма Graceful Restart
show ldp igp	Параметры IGP
show ldp interface	Статус интерфейсов с функцией LDP
show ldp lsp	Просмотр пути прохождения пакета на основе протоколов LDP
show ldp mpls-l2-circuit	Просмотр конфигурации l2-circuit
show ldp ms-pw	Multi-Segment PW information
show ldp routes	Таблица NSM маршрутов LDP
show ldp session	Информация о сессии LDP
show ldp statistics	Просмотр статистики LDP
show ldp targeted-peer	Информация о пограничном MPLS маршрутизаторе
show ldp targeted-peers	List of targeted peers defined

16.3 Pseudowire

Pseudowire (pseudo-wire) или L2-circuit – это сервис виртуальной частной сети для связи между собой двух сегментов сети по типу точка-точка. Любому поступающему трафику на PE маршрутизаторе назначается метка MPLS по которой происходит маршрутизация.

16.3.1 Настройка L2-circuit

Базовая настройка pseudowire включает в себя настройку граничных (Label Edge Router, LER) и промежуточных (Label Switch Router, LSR) маршрутизаторов сети.

Пример настройки LSR.

Создание loopback интерфейса.

```
ecorouter(config)#interface loopback.<number>
ecorouter(config-if)#ip address <address/mask>
```

Переход в режим настройки протокола LDP.

```
ecorouter(config)#router ldp
```

Определение транспортного адреса маршрутизатора.

```
ecorouter(config-router)#transport-address ipv4 <ip-address>
```

Включение LDP и функции работы с метками на интерфейсах.

```
ecorouter(config-if)#enable-ldp ipv4
ecorouter(config-if)#label-switching
```

Пример настройки LER.

Создание loopback интерфейса.

```
ecorouter(config)#interface loopback.<number>
ecorouter(config-if)#ip address <address/mask>
```

Переход в режим настройки протокола LDP.

```
ecorouter(config)#router ldp
```

Определение транспортного адреса маршрутизатора.

```
ecorouter(config-router)#transport-address ipv4 <ip-address>
```

Определение целевого маршрутизатора. Где в качестве <ip-address> указывается сетевой адрес пограничного маршрутизатора, до которого будет построен l2-circuit.

```
ecorouter(config-router)#targeted-peer ipv4 <ip-address>
```

Включение ldp и функции работы с метками на интерфейсах.

```
ecorouter(config-if)#enable-ldp ipv4
ecorouter(config-if)#label-switching
```

L2-circuit конфигурируется в зависимости от типа создаваемой схемы.

Создание l2-circuit type 5.

```
mpls l2-circuit <name> <Identifying value> <ip-address for end-point>
```

Где в качестве <name> задается идентификационное имя соединения, <Identifying value> - номер l2-circuit, <ip-address for end-point> - адрес граничного маршрутизатора.

Создание l2-circuit type 4.

```
mpls l2-circuit <name> <Identifying value> <ip-address for end-point>
mode tagged svlan <vlan Identifier>
```

Где в качестве <name> задается идентификационное имя соединения, <Identifying value> - номер l2-circuit, <ip-address for end-point> - адрес граничного маршрутизатора, <vlan Identifier> - номер виртуальной сети.

Привязка созданной l2-circuit к порту.

```
ecorouter(config)#port ge2
ecorouter(config-port)#service-instance ge2/e2
ecorouter(config-service-instance)#encapsulation <tag/untag>
ecorouter(config-service-instance)#mpls-l2-circuit <name>
```

Где в зависимости от типа l2-circuit указывается тегированный или нетегированный трафик, параметр <name> - имя ранее созданного l2-circuit.

Просмотр состояния l2-circuit. Где <name> - имя ранее созданного l2-circuit.

```
ecorouter#show mpls l2-circuit <name>
```

Гибкая настройка различных операций с VLAN-тегами на service-instance позволяет передавать пакет через l2-circuit, предварительно проделав эти операции с VLAN-тегами. При этом используется тип инкапсуляции 5 (ethernet).

Поддерживаются следующие операции:

Снять внешнюю метку с пакета с двумя метками, перед отправкой в MPLS-туннель:

```
mpls l2-circuit pop_sv_any_cv 20 2.2.2.2
!
port tel
  service-instance pop_sv_any_cv
  encapsulation dot1q 40 second-dot1q any
```

```
rewrite pop 1
mpls-l2-circuit pop_sv_any_cv primary
```

Внутренняя метка может быть любой (second-dot1q any) или жестко заданной (second-dot1q 100). Во втором случае, все пакеты должны иметь внешнюю метку 40 и внутреннюю метку 100. В противном случае пакет будет отброшен.

Снять обе метки с пакета перед отправкой в MPLS-туннель:

```
mpls l2-circuit pop_pop 30 2.2.2.2
!
port tel
  service-instance pop_pop
  encapsulation dot1q 40 second-dot1q 90
  rewrite pop 2
mpls-l2-circuit pop_pop primary
```

Снять внешнюю метку и заменить внутреннюю на произвольную перед отправкой в MPLS-туннель:

```
mpls l2-circuit pop_swap 40 2.2.2.2
!
port tel
  service-instance pop_swap
  encapsulation dot1q 40 second-dot1q 90
  rewrite translate 2-to-1 77
mpls-l2-circuit pop_swap primary
```

Добавить внешнюю метку перед отправкой в MPLS-туннель:

```
mpls l2-circuit push_sv 50 2.2.2.2
!
port tel
  service-instance push_sv
  encapsulation dot1q 60 exact
  rewrite push 77
mpls-l2-circuit push_sv primary
```

Добавить две метки перед отправкой в MPLS-туннель:

```
mpls l2-circuit push_two 60 2.2.2.2
!
port tel
  service-instance push_two
  encapsulation untagged
  rewrite push 77 88
mpls-l2-circuit push_two primary
```

Заменить внешнюю метку перед отправкой в MPLS-туннель:

```
mpls l2-circuit swap_sv 70 2.2.2.2
!
port tel
  service-instance swap_sv
  encapsulation dot1q 40 second-dot1q 90
  rewrite translate 1-to-1 77
mpls-l2-circuit push_two primary
```

Заменить обе метки перед отправкой в MPLS-туннель:

```
mpls l2-circuit swap_swap 80 2.2.2.2
```

```
!
port tel
service-instance swap_swap
encapsulation dot1q 40 second-dot1q 90
rewrite translate 2-to-2 77 88
mpls-l2-circuit swap_swap primary
```

Заменить внутреннюю метку и добавить внешнюю перед отправкой в MPLS-туннель:

```
mpls l2-circuit swap_push 90 2.2.2.2
!
port tel
service-instance swap_push
encapsulation dot1q 60 exact
rewrite translate 1-to-2 77 88
mpls-l2-circuit swap_push primary
```

16.3.2 Backup Pseudowire

Pseudowire Redundancy (backup pseudowire) позволяет настроить один из граничных маршрутизаторов сети MPLS для обнаружения сбоя в сети и перенаправить трафик к другой конечной точке. Функция обеспечивает возможность восстановления после сбоя одного из удаленных граничных маршрутизаторов.

Для аварийного переключения на резервный pseudowire в конфигурации EcoRouter должно быть настроено два L2 туннеля, один из которых будет выполнять роль backup pseudowire. При передаче трафика по основному L2 туннелю backup pseudowire будет находиться в состоянии standby.

Для настройки backup pseudowire необходимо произвести описанные ниже действия.

Создать loopback интерфейс loopback.0 с сетевым адресом 1.1.1.1 и маской 32.

```
ecorouter(config)#interface loopback.0
ecorouter(config-if)#ip address 1.1.1.1/32
```

Перейти в режим настройки протокола LDP.

```
ecorouter(config)#router ldp
```

Определить транспортный адрес маршрутизатора.

```
ecorouter(config-router)#transport-address ipv4 1.1.1.1
```

Определить целевой маршрутизатор, например, сетевой адрес конечного маршрутизатора будет 2.2.2.2 с маской 32.

```
ecorouter(config-router)#targeted-peer ipv4 2.2.2.2
```

Включить режим распространения меток по всей таблице маршрутизации.

```
ecorouter(config-router)#pw-status-tlv
```

Включить LDP и функцию работы с метками на интерфейсе в сторону MPLS сети.

```
ecorouter(config-if)#enable-ldp ipv4
ecorouter(config-if)#label-switching
```

Далее необходимо настроить основной L2 туннель. Например, создать l2-circuit type 5 с именем vc1, Identifying value – 1111.

Для этого нужно создать l2-circuit type 5.

```
mpls 12-circuit vc1 1111 2.2.2.2
```

Настроить резервный L2 туннель, с именем vc2, Identifying value – 2222.

```
mpls 12-circuit vc2 2222 2.2.2.2
```

Привязать созданный l2-circuit к порту ge2, включить функцию переключения на основной l2-circuit при его доступности.

```
ecorouter(config)#port ge2
ecorouter(config-port)#service-instance ge2/e2
ecorouter(config-service-instance)#encapsulation untag
ecorouter(config-service-instance)#mpls-12-circuit vc1
ecorouter(config-service-instance)#mpls-12-circuit vc2
ecorouter(config-service-instance)#vc-mode revertive
```

16.4 Совместная работа BGP и MPLS

В данном разделе рассматривается реализация совместной работы протоколов BGP и MPLS на базе EcoRouterOS.

Главным отличием протокола BGP от IGP при работе с MPLS является то, что для BGP-маршрутов метки не создаются. Когда маршрутизатор LSR получает маршрут по BGP, то дальше он передает пакеты в сторону BGP-соседа, который указан, как next-hop в анонсе этого маршрута, используя созданную для next-hop метку. Поэтому нет необходимости настраивать BGP на каждом маршрутизаторе в автономной системе, его конфигурируют только на пограничных маршрутизаторах, к которым подключены клиенты или другие провайдеры.

16.4.1 Топология

Приведенная ниже схема реализует классический сценарий совместной работы протоколов BGP и MPLS, который явно демонстрирует все плюсы коммутации по меткам.

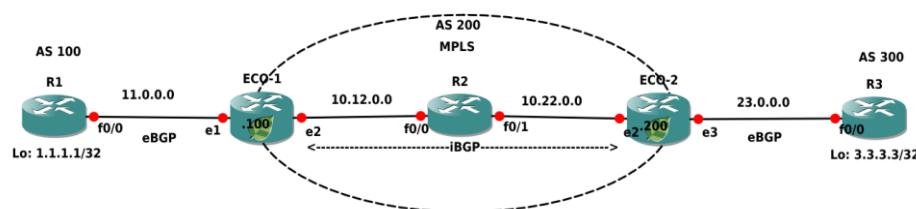


Рисунок 29

На схеме маршрутизаторы ECO-1, ECO-2 и R2 находятся в MPLS-облаче, и между ECO-1 и ECO-2 настроен iBGP. Маршрутизаторы R1 и R3 подключены к MPLS-облачу через eBGP. Локальные сети маршрутизаторов R1 и R3 представлены в виде loopback-интерфейсов. Необходимо создать связность между локальными сетями маршрутизаторов R1 и R3.

16.4.2 Конфигурация маршрутизаторов

Ниже приведена конфигурация маршрутизаторов для реализации данной схемы.

ECO-1

```
ECO-1#sh running-config
!
router ldp
```

```

transport-address ipv4 100.100.100.100
!
mpls map-route 3.3.3.3/32 200.200.200.200/32
!
router ospf 1
network 10.0.0.0 0.255.255.255 area 0.0.0.0
network 100.100.100.100 0.0.0.0 area 0.0.0.0
!
router bgp 200
neighbor 11.0.0.1 remote-as 100
neighbor 200.200.200.200 remote-as 200
neighbor 200.200.200.200 update-source loopback.0
neighbor 200.200.200.200 next-hop-self
!
port te0
lacp-priority 32767
mtu 9728
service-instance te0/e1
  encapsulation untagged
!
port te1
lacp-priority 32767
mtu 9728
service-instance te1/e2
  encapsulation untagged
!
interface loopback.0
ip mtu 1500
ip address 100.100.100.100/32
!
interface e2
ip mtu 1500
label-switching
connect port te1 service-instance te1/e2
ip address 10.12.0.100/16
ldp enable ipv4
!
interface e1
ip mtu 1500
connect port te0 service-instance te0/e1
ip address 11.0.0.100/16
!
end

```

ECO-2

```

ECO-2#sh running-config
!
router ldp
transport-address ipv4 200.200.200.200
!
mpls map-route 1.1.1.1/32 100.100.100.100/32
!
router ospf 1
network 10.0.0.0 0.255.255.255 area 0.0.0.0

```

```

network 200.200.200.200 0.0.0.0 area 0.0.0.0
!
router bgp 200
neighbor 23.0.0.3 remote-as 300
neighbor 100.100.100.100 remote-as 200
neighbor 100.100.100.100 update-source loopback.0
neighbor 100.100.100.100 next-hop-self
!
port te1
lacp-priority 32767
mtu 9728
service-instance te1/e2
  encapsulation untagged
!
port te2
lacp-priority 32767
mtu 9728
service-instance te2/e3
  encapsulation untagged
!
interface loopback.0
ip mtu 1500
ip address 200.200.200.200/32
!
interface e3
ip mtu 1500
connect port te2 service-instance te2/e3
ip address 23.0.0.200/16
!
interface e2
ip mtu 1500
label-switching
connect port te1 service-instance te1/e2
ip address 10.22.0.200/16
ldp enable ipv4
!
end

```

R1

```

R1#sh running-config
!
router bgp 100
neighbor 11.0.0.100 remote-as 200
network 1.1.1.1 mask 255.255.255.255
!
port te0
lacp-priority 32767
mtu 9728
service-instance te0/FastEthernet0/0
  encapsulation untagged
!
interface loopback.0
ip mtu 1500
ip address 1.1.1.1/32

```

```
!
interface FastEthernet0/0
ip mtu 1500
connect port te0 service-instance te0/FastEthernet0/0
ip address 11.0.0.1/16
!
end
```

R3

```
R3#sh running-config
!
router bgp 300
neighbor 23.0.0.200 remote-as 200
network 3.3.3.3 mask 255.255.255.255
!
port te0
lacp-priority 32767
mtu 9728
service-instance te0/FastEthernet0/0
  encapsulation untagged
!
interface loopback.0
ip mtu 1500
ip address 3.3.3.3/32
!
interface FastEthernet0/0
ip mtu 1500
connect port te0 service-instance te0/FastEthernet0/0
ip address 23.0.0.3/16
!
end
```

R2

```
R2#sh running-config
!
router ldp
transport-address ipv4 22.22.22.22
!
mpls map-route 3.3.3.3/32 200.200.200.200/32
!
router ospf 1
  network 10.0.0.0 0.255.255.255 area 0.0.0.0
  network 22.22.22.22 0.0.0.0 area 0.0.0.0
!
port te0
lacp-priority 32767
mtu 9728
service-instance te0/FastEthernet0/1
  encapsulation untagged
!
port tel
lacp-priority 32767
mtu 9728
service-instance tel1/FastEthernet0/0
```

```

encapsulation untagged
!
interface loopback.0
ip mtu 1500
ip address 22.22.22.22/32
!
interface FastEthernet0/0
ip mtu 1500
label-switching
connect port tel service-instance tel/FastEthernet0/0
ip address 10.12.0.2/16
ldp enable ipv4
!
interface FastEthernet0/1
ip mtu 1500
label-switching
connect port te0 service-instance te0/FastEthernet0/1
ip address 10.22.0.2/16
ldp enable ipv4
!
end

```

Для связности между loopback-интерфейсами маршрутизаторов R1 и R3 не требуется, чтобы на маршрутизаторе R2 был настроен BGP и присутствовали все маршруты в таблице маршрутизации. При увеличении MPLS-облака в размерах это становится заметным преимуществом использования технологии коммутации по меткам.

Ниже представлен вывод на консоль таблицы маршрутизации ECO-1.

```

ECO-1#sh ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter
area
      * - candidate default
IP Route Table for VRF "default"
B    1.1.1.1/32 [20/0] via 11.0.0.1, e1, 19:33:53
B    3.3.3.3/32 [200/0] via 200.200.200.200 (recursive via 10.12.0.2 ), 19:33:40
C    10.12.0.0/16 is directly connected, e2
O    10.22.0.0/16 [110/20] via 10.12.0.2, e2, 19:34:09
C    11.0.0.0/16 is directly connected, e1
C    100.100.100.100/32 is directly connected, loopback.0
O    200.200.200.200/32 [110/30] via 10.12.0.2, e2, 19:33:56

```

16.4.3MPLS карта

Маршрут до адреса 3.3.3.3/32, полученный от BGP-соседа ECO-2, пролегает по MPLS-облаку через устройство с адресом 10.12.0.2. Такие маршруты называются рекурсивными. Для того чтобы при передаче пакетов в сторону адреса 3.3.3.3 добавлялась MPLS-метка, предназначенная для адреса next-hop BGP-соседа, в EcoRouterOS требуется явно указать «MPLS карту».

Для этого необходимо ввести команду конфигурационного режима **mpls map-route <IP подсеть/маска подсети> <FEC подсеть/маска подсети>**, где подсети задаются статически. Первый параметр в команде - IP-подсеть, для которой необходимо составить MPLS-карту. Второй параметр - FEC для этой подсети. FEC (Forwarding Equivalence Class) представляет собой класс трафика. В простейшем случае идентификатором класса является адресный префикс назначения (другими словами, IP-адрес или подсеть назначения).

В приведенной выше конфигурации маршрутизатора ECO-1 этому действию соответствует строка:

```
mpls map-route 3.3.3.3/32 200.200.200.200/32
```

Эта строка конфигурации означает, что при отправке пакета в сторону подсети 3.3.3.3/32 для него необходимо использовать метку для подсети 200.200.200.200/32.

Подобные статические карты более полно описывают топологию и операции над фреймами, что позволяет уменьшить время поиска проблем в сети.

17 MPLS L3 VPN

Технология L3-VPN позволяет организовывать изолированные виртуальные частные сети с индивидуальными таблицами маршрутизации (VRF) на базе MPLS сети оператора. Пользовательская информация о маршрутах импортируется в VRF, используя цель маршрута (Route Target, RT). Данная информация идентифицируется по различителю маршрута (Route Distinguisher, RD) и распространяется между PE-маршрутизаторами, используя расширенную версию протокола MP-BGP.

17.1 Требования

Для того чтобы данная технология полностью работала, необходимо задействовать поддержку следующих протоколов:

- MP-BGP,
- LDP,
- MPLS,
- OSPFv2,
- RIP.

17.2 MPLS VPN терминология

На рисунке ниже показана сеть оператора Connector с частными виртуальными сетями клиентов ComA и ComB.

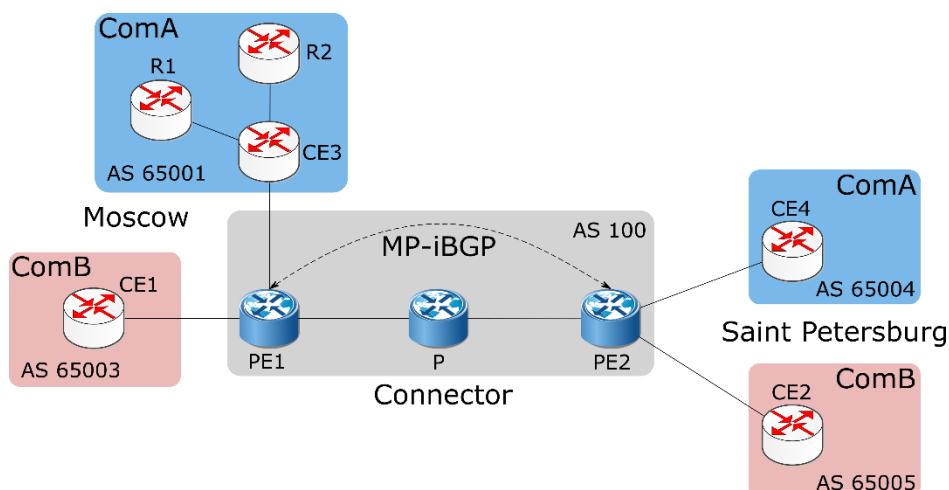


Рисунок 30

Пограничное устройство клиента (Customer Edge Router, CE) - маршрутизатор на стороне клиента, который подсоединен к сети оператора связи (к PE-маршрутизатору). На рисунке это CE1, CE2, CE3 и CE4.

Пограничное устройство оператора (Provider Edge Router, PE) - операторский маршрутизатор, к которому подключен CE-маршрутизатор. На рисунке это маршрутизаторы PE1 и PE2, которые соединяют клиентское оборудование с сетью оператора Connector.

Маршрутизаторы сети оператора (Provider Core Router, P) - устройства внутри сети оператора, не являющиеся пограничными. На рисунке это маршрутизатор P, не соединенный с клиентскими устройствами, и принадлежащий сети оператора Connector.

Клиентские маршрутизаторы (Customer Router, R) - устройства внутри клиентской сети, не подключенные напрямую к сети оператора. На рисунке выше R1 и R2 - это клиентские маршрутизаторы.

17.3 Процесс маршрутизации сетей VPN

Процесс маршрутизации MPLS-VPN включает следующие этапы:

1. Оператор предоставляет услугу VPN через PE-маршрутизаторы, которые подсоединены напрямую к клиентским CE-маршрутизаторам по Ethernet.
2. Каждый PE-маршрутизатор содержит таблицу маршрутизации (VRF) для каждого клиента. Это гарантирует изоляцию клиентских сетей и позволяет использовать частные адреса независимо от адресации сети оператора и других клиентов. Когда приходит пакет от CE, используется таблица VRF, которая назначена для данной сети, и по ней определяется маршрут передачи данных. Если PE-маршрутизатор связан с сетью несколькими линками, то для всех этих подключений используется одна таблица VRF.
3. После того как PE-маршрутизатор определил IP- префикс, он конвертирует его в VPN-IPv4 префикс, предваряя его 8-байтовым (64 бит) различителем маршрута (RD). RD гарантирует, что даже если у двух клиентов одинаковые адреса, к ним будут установлены два разных маршрута. Эти VPN-IPv4-адреса анонсируются среди PE-маршрутизаторов по MP-BGP.
4. Для определения MPLS-метки и передачи VPN-пакета через сеть оператора используется уникальный идентификатор маршрутизатора (как правило, его loopback-адрес).
5. Пакеты передаются в точку назначения по MPLS, ориентируясь на информацию из таблицы маршрутизации. Каждый PE-маршрутизатор определяет уникальную метку для каждого маршрута в таблицах маршрутизации (даже если у них один и тот же next hop) и объявляет эту метку вместе с 12-байтовым VPN-IPv4 адресом по MP-BGP.
6. PE-маршрутизаторы на входе (ingress) прикрепляют к VPN-пакету, отправляемому по сети оператора, стек из двух меток. В него входят: сервисная метка - BGP-метка, определенная из таблицы маршрутизации (ассоциированной с входящим интерфейсом), которая указывает на BGP next hop; транспортная метка - LDP-метка из глобальной FTN таблицы, определяющая IP next hop.
7. Операторский (P) маршрутизатор в сети перекидывает VPN-пакет в зависимости от транспортной метки. Эта метка используется как ключ для поиска входного интерфейса в таблице Incoming Labels Mapping (ILM). Если у пакета две метки, верхняя меняется, и пакет отправляется на следующий узел. Если нет, то маршрутизатор является предпоследним в цепочке, и он снимает транспортную метку и отправляет пакет только с сервисной меткой на PE-маршрутизатор на выходе. Каждый раз, когда пакет проходит очередной маршрутизатор Р вдоль туннеля, транспортная метка анализируется и заменяется новым значением. Предпоследний маршрутизатор в цепочке снимает транспортную метку и на конечную точку туннеля

- маршрутизатор PE2 - пакет приходит с одной меткой. В случае если включена опция **mpls explicit-null**, предпоследний маршрутизатор отправляет пакет с двумя метками, где значение верхней метки - 0.
- 8. Выходной PE-маршрутизатор снимает BGP-метку, производит поиск по ней на исходящих интерфейсах и отправляет пакет соответствующему клиентскому CE-маршрутизатору.

17.4 Конфигурирование MPLS Layer-3 VPN

Процесс конфигурирования MPLS Layer-3 VPN можно разделить на следующие этапы:

1. Установка соединения между PE-маршрутизаторами.
2. Настройка iBGP соседства между PE1 и PE2.
3. Создание VRF.
4. Подключение интерфейсов к VRF.
5. Настройка для таблиц VRF различителей маршрутов (RD) и целей маршрутов (RT).
6. Настройка соседей CE для VPN.
7. Проверка конфигурации перехода от MPLS к VPN.

17.4.1 Топология

В приведенном примере к опорной MPLS-VPN сети оператора Connector подключены для клиента: ComA и ComB. Сайты обоих клиентов находятся в Москве и Санкт-Петербурге. На рисунке ниже приведена топология сети, показывающая распределение BGP4-адресов между PE и CE маршрутизаторами. Далее описана последовательность действий по настройке клиентских виртуальных сетей поверх опорной MPLS-VPN.

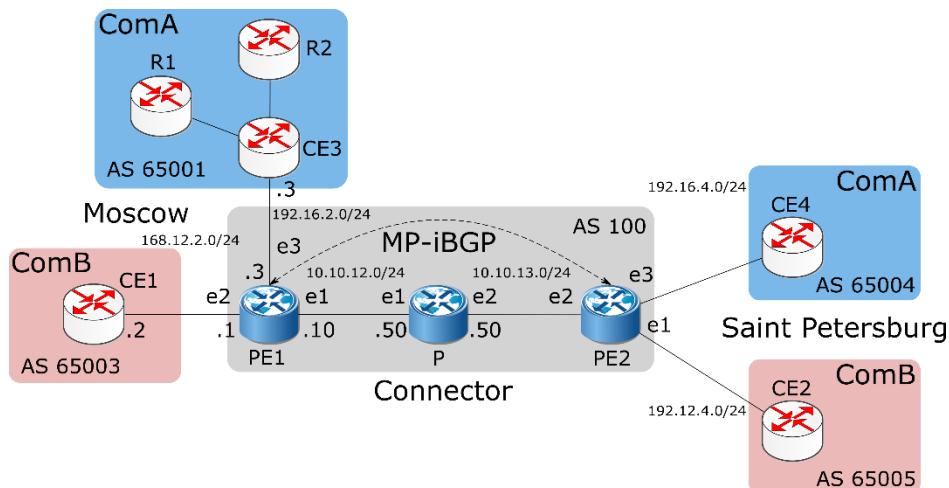


Рисунок 31

Для установки соединения между маршрутизаторами требуется осуществить действия, описанные ниже.

17.4.2 Включение коммутации по меткам

Ниже приведена примерная конфигурация для включения коммутации по меткам (Labeled Switched Path, LSP) между маршрутизаторами PE1 и PE2.

PE1

```
PE1(config)#interface e1
PE1(config-if)#ip address 10.10.12.10/24
PE1(config-if)#label-switching
PE1(config-if)#ex
PE1(config)#port tel
PE1(config-port)#service-instance sel
PE1(config-service-instance)#encapsulation untagged
PE1(config-service-instance)#connect ip interface e1
```

P

```
P(config)#interface e1
P(config-if)#ip address 10.10.12.50/24
P(config-if)#label-switching
P(config-if)#ex
P(config)#port tel
P(config-port)#service-instance sel
P(config-service-instance)#encapsulation untagged
P(config-service-instance)#connect ip interface e1
P(config-service-instance)#ex
P(config-port)#ex
P(config)#interface e2
P(config-if)#ip address 10.10.13.50/24
P(config-if)#label-switching
P(config-if)#ex
P(config)#port te2
P(config-port)#service-instance se2
P(config-service-instance)#encapsulation untagged
P(config-service-instance)#connect ip interface e2
```

PE2

```
PE2(config)#interface e2
PE2(config-if)#ip address 10.10.13.10/24
PE2(config-if)#label-switching
PE2(config-if)#ex
PE2(config)#port te2
PE2(config-port)#service-instance se2
PE2(config-service-instance)#encapsulation untagged
PE2(config-service-instance)#connect ip interface e2
```

17.4.3 Включение IGP

Ниже приведен пример конфигурации для установки соединения между двумя PE-маршрутизаторами PE1 и PE2.

Подробнее о настройке OSPF можно прочитать в соответствующем разделе "Open Shortest Path First".

PE1

```
PE1(config)#router ospf 100
PE1(config-router)#network 10.10.12.0/24 area 0
```

P

```
P(config)#router ospf 100
```

```
P(config-router)#network 10.10.12.0/24 area 0
P(config-router)#network 10.10.13.0/24 area 0
```

PE2

```
PE2(config)#router ospf 100
PE2(config-router)#network 10.10.13.0/24 area 0
```

17.4.4 Включение протокола коммутации меток

Данный протокол используется для построения путей коммутации по меткам (LSP) между PE-маршрутизаторами. В EcoRouterOS поддерживается протокол LDP.

Ниже приведен пример конфигурации для включения LDP на всем пути между PE1 и PE2. В конфигурации PE-маршрутизаторов присутствует настройка loopback-интерфейса, необходимая для работы LDP и BGP (см. ниже).

Подробнее о настройке LDP можно прочитать в соответствующем разделе "Label Distribution Protocol".

PE1

```
PE1(config)#interface loopback.0
PE1(config-lo)#ip address 2.2.2.2/32
PE1(config-lo)#ex
PE1(config)#router ldp
PE1(config-router)#exit
PE1(config)#interface e1
PE1(config-if)#ldp enable ipv4
PE1(config-if)#ex
PE1(config)#router ldp
PE1(config-router)#advertisement-mode downstream-on-demand
PE1(config-router)#multicast-hellos
```

P

```
P(config)#interface e1
P(config-if)#ldp enable ipv4
P(config-if)#ex
P(config)#interface e2
P(config-if)#ldp enable ipv4
P(config-if)#ex
P(config)#router ldp
P(config-router)#advertisement-mode downstream-on-demand
P(config-router)#multicast-hellos
```

PE2

```
PE2(config)#interface loopback.0
PE2(config-lo)#ip address 3.3.3.3/32
PE2(config-lo)#ex
PE2(config)#router ldp
PE2(config-router)#exit
PE2(config)#interface e2
PE2(config-if)#ldp enable ipv4
PE2(config-if)#ex
PE2(config)#router ldp
PE2(config-router)#advertisement-mode downstream-on-demand
PE2(config-router)#multicast-hellos
```

17.4.5 Настройка BGP-соседства между PE-маршрутизаторами

Для передачи маршрутной информации частных сетей через сеть оператора используется протокол BGP и его многопротокольное расширение MP-BGP. Это позволяет обмениваться информацией между опосредованно соединенными маршрутизаторами, а также передавать маршрутную информацию сетей VPN, минуя маршрутизаторы опорной сети оператора (P). Через P-маршрутизаторы информация передается прозрачно, как дополнительный BGP атрибут. В MPLS-VPN модели нет необходимости в том, чтобы P-маршрутизаторы принимали решения о маршрутах, основываясь на внутренней адресации сетей VPN. Они просто передают пакеты в соответствии со значениями прикрепленных меток. Таким образом, на P-маршрутизаторы не требуется добавлять конфигурацию сетей VPN.

Подробнее о настройке BGP можно прочитать в соответствующем разделе "Border Gateway Protocol".

PE1

```
PE1(config)#router bgp 100
PE1(config-router)#neighbor 3.3.3.3 remote-as 100
PE1(config-router)#neighbor 3.3.3.3 update-source 2.2.2.2
PE1(config-router)#address-family vpnv4 unicast
PE1(config-router-af)#neighbor 3.3.3.3 activate
```

PE2

```
P2(config)#router bgp 100
P2(config-router)#neighbor 2.2.2.2 remote-as 100
P2(config-router)#neighbor 2.2.2.2 update-source 3.3.3.3
P2(config-router)#address-family vpnv4 unicast
P2(config-router-af)#neighbor 2.2.2.2 activate
```

17.4.6 Создание VRF

Каждый PE-маршрутизатор в опорной сети MPLS-VPN подсоединен к сайтам, входящим в виртуальные частные сети клиентов. Для каждого сайта действуют маршруты соответствующей сети VPN. Поэтому на PE-маршрутизаторе должны содержаться таблицы VRF для тех сетей VPN, к сайтам которых он подключен. В приведенном примере - это обе сети VPN.

Для создания таблицы VRF введите команду конфигурационного режима `ip vrf <VRF_NAME>`. На каждом PE-маршрутизаторе должны быть созданы таблицы VRF с именами ComA и ComB. При вводе данной команды создается таблица маршрутизации VRF RIB (Routing Information Base), назначается VRF-ID, и консоль переключается в контекстный режим конфигурирования VRF.

```
PE1(config)#ip vrf ComB
PE1(config-vrf) #
```

17.4.7 Подключение интерфейсов к VRF

После того как на каждом PE-маршрутизаторе определены таблицы VRF, необходимо указать, какой интерфейс маршрутизатора принадлежит к какой таблице VRF. VRF заполняются маршрутами с подсоединенными сайтов. К одной таблице VRF могут быть подключены несколько интерфейсов. Для подключения интерфейса (подсоединенного к СЕ-

маршрутизатору) используется команда контекстного режима конфигурации интерфейса **ip vrf forwarding <VRF_NAME>**.

В приведенном ниже примере интерфейс e2 маршрутизатора PE1 подключается к созданной ранее таблице VRF ComB.

```
PE1(config)#interface e2
PE1(config-if)#ip vrf forwarding ComB
```

17.4.8 Настройка VRF-RD и целевых маршрутов

После того как таблицы VRF созданы, настраиваются различители маршрутов и цели маршрутов.

Настройка различителей маршрутов

Различители маршрутов (Route Distinguishers, RDs) обеспечивают уникальность каждого маршрута. Таким образом, в случае одинаковых маршрутов в разных сетях VPN, MP-BGP будет воспринимать их как уникальные. Для этого к каждому IPv4-адресу из виртуальной сети добавляется префикс длиной 64 бит (RD), преобразуя его в формат VPN-IPv4. BGP считает два IPv4-адреса с разными RD уникальными (несравнимыми), даже если у них совпадают и адрес, и маска.

RD состоит из номера автономной системы и присвоенного номера (ASN:nn) или IP-адреса и присвоенного номера (IP:nn), записанных через двоеточие ':'.

Для того чтобы назначить RD каждой таблице VRF на PE-маршрутизаторе используется команда контекстного режима конфигурирования VRF **rd <ASN:nn | IP:nn>**.

В приведенном ниже примере назначается RD для VRF ComB на маршрутизаторе PE1.

```
PE1(config)#ip vrf ComB
PE1(config-vrf)#rd 168.12.2.1:1
```

Для просмотра таблицы маршрутизации данной таблицы VRF используется команда административного режима **show ip route vrf <VRF_NAME>** или команда административного режима **show ip route vrf all** для всех VRF.

Настройка целевых маршрутов

Все полученные от клиентов маршруты анонсируются по всей сети по протоколу MP-BGP. Все маршруты, узнанные по MP-BGP, добавляются в соответствующую таблицу VRF. Цель маршрута (RT) помогает PE-маршрутизаторам идентифицировать, к какой таблице VRF относится маршрут.

Для того чтобы назначить RT каждой таблице VRF на PE-маршрутизаторе, используется команда контекстного режима конфигурирования VRF **route-target {both | export | import} <ASN:nn | IP:nn>**.

Команда **route-target** создает списки импорта и экспорта расширенных атрибутов сообщества (в том числе, RT) для VRF. RT идентифицирует целевую сеть VPN. Данную команду необходимо вводить отдельно для каждого сообщества. Все маршруты с указанными расширенными атрибутами сообщества импортируются во все VRF, относящиеся к тем же сообществам в качестве целевого маршрута импорта.

В команде **route-target** также задается политика экспорта маршрутных объявлений:

- **export** - добавить RT к экспортируемой маршрутной информации VRF;
- **import** - импортировать маршрутную информацию с указанным RT;
- **both** - указать сразу и импорт, и экспорт.

Указанные политики задаются в зависимости от планируемой топологии сети. Например, задание одного и того же значения для политики экспорта и импорта для всех таблиц VRF определенной сети VPN приводит к полносвязной топологии – каждый сайт может посылать пакеты непосредственно тому сайту, в котором находится сеть назначения.

В приведенном ниже примере назначается RT для VRF ComB на маршрутизаторе PE1. Для остальных маршрутизаторов и сетей в рассматриваемой топологии задается то же значение политики экспорта.

```
PE1 (config) #ip vrf ComB  
PE1 (config-vrf) #route-target both 100:1
```

17.4.9 Конфигурация СЕ-соседей для VPN (с использованием BGP / OSPF / RIP)

Для предоставления услуги VPN, PE-маршрутизаторы должны быть сконфигурированы таким образом, чтобы любая маршрутная информация, приходящая с интерфейса клиентской сети VPN могла быть соотнесена с соответствующей таблицей VRF. Это достигается за счет распространения по сети маршрутной информации протоколами маршрутизации, такими как BGP, OSPF, IS-IS, RIP. Для настройки СЕ-соседства используются приведенные ниже действия, в зависимости от используемого протокола (BGP, OSPF или RIP).

BGP

BGP-сессия между PE и СЕ-маршрутизаторами может включать разные типы маршрутов (VPN-IPv4, IPv4 маршруты). Соответственно, от используемого семейства адресов зависит тип BGP-сессии. Таким образом, необходимо настроить семейство адресов BGP для каждой таблицы VRF на PE-маршрутизаторах и отдельно адресное семейство для VPN-IPv4-маршрутов между PE-маршрутизаторами. Все не-VPN BGP-соседи определяются при помощи режима IPv4-адресов. Каждое VPN BGP-соседство определяется связанным с ним режимом семейства адресов. Для того чтобы задать семейство адресов, используется команда режима конфигурации маршрутизации BGP **address-family ipv4 vrf <VRF_NAME>**.

Отдельная запись о семействе адресов должна быть в каждой таблице VRF, в каждой записи о семействе адресов может значиться несколько СЕ-маршрутизаторов с VRF.

PE и СЕ-маршрутизаторы должны быть напрямую подключены для BGP4-сессий; BGP multihop между ними не поддерживается.

В приведенном ниже примере маршрутизатор переключается в режим семейства адресов и указываются имена компаний-клиентов ComA и ComB в качестве названий VRF, для того чтобы проассоциировать их с подмножеством команд, соответствующим IPv4 семейству адресов. Подобная конфигурация используется, когда между PE и СЕ-маршрутизаторами настроен BGP.

PE1

```
PE1 (config) #router bgp 100
```

```
PE1 (config-router) #address-family ipv4 vrf ComA
PE1 (config-router-af) #neighbor 192.16.3.3 remote-as 65001
PE1 (config-router-af) #exit
PE1 (config-router) #address-family ipv4 vrf ComB
PE1 (config-router-af) #neighbor 168.12.0.2 remote-as 65003
```

OSPF

В отличие от BGP и RIP, OSPF не поддерживает разные контексты маршрутизации в одном процессе. Для запуска OSPF между PE и CE-маршрутизаторами настраивается отдельный OSPF-процесс для каждой VRF, который получает маршруты сети VPN по OSPF. PE-маршрутизатор различает принадлежность маршрутизаторов к определенной VRF, связывая конкретный клиентский интерфейс с таблицей VRF и с определенным процессом OSPF.

Чтобы распространить OSPF-маршруты таблицы VRF в BGP, необходимо включить редистрибуцию OSPF в контексте конфигурирования маршрутизации BGP для семейства адресов, связанного с VRF.

PE1

```
PE1 (config) #router ospf 101 ComA
PE1 (config-router) #network 192.16.3.0/24 area 0
PE1 (config-router) #redistribute bgp
PE1 (config-router) #ex
PE1 (config) #router ospf 102 ComB
PE1 (config-router) #network 192.12.0.0/24 area 0
PE1 (config-router) #redistribute bgp
```

PE1

```
PE1 (config) #router bgp 100
PE1 (config-router) #address-family ipv4 vrf ComA
PE1 (config-router-af) #redistribute ospf
PE1 (config-router-af) #ex
PE1 (config-router) #address-family ipv4 vrf ComB
PE1 (config-router-af) #redistribute ospf
```

17.4.10 Проверка настройки MPLS-VPN

Для того чтобы проверить соседство между CE и PE-маршрутизаторами используется команда административного режима **show ip bgp neighbor**. Для просмотра всех созданных VRF и маршрутов в них используется команда **show ip bgp vpnv4 all**. Ниже приведен пример вывода команды **show running-config** для маршрутизаторов PE1, CE1 и P, сконфигурированных в соответствии с топологией рассматриваемого примера. Для связи PE с CE используется OSPF.

PE1

```
PE1#show running-config
!
hostname PE1
!
ip vrf management
!
ip vrf ComA
```

```

rd 168.12.2.1:1
route-target both 100:1
!
ip vrf ComB
rd 192.16.2.1:1
route-target both 100:1
!
mpls propagate-ttl
!
!
ip pim register-rp-reachability
!
router ldp
targeted-peer ipv4 10.10.21.50
exit-targeted-peer-mode
advertisement-mode downstream-on-demand
!
router ospf 100
network 10.10.12.0/24 area 0.0.0.0
!
router ospf 101 ComA
redistribute bgp
network 192.16.3.0/24 area 0.0.0.0
!
router ospf 102 ComB
redistribute bgp
network 192.12.0.0/24 area 0.0.0.0
!
router bgp 100
neighbor 3.3.3.3 remote-as 100
neighbor 3.3.3.3 update-source 2.2.2.2
address-family vpnv4 unicast
neighbor 3.3.3.3 activate
exit-address-family
!
address-family ipv4 vrf ComA
redistribute ospf
exit-address-family
!
address-family ipv4 vrf ComB
redistribute ospf
exit-address-family
!
interface loopback.0
ip mtu 1500
ip address 2.2.2.2/32
!
interface e1
ip mtu 1500
label-switching connect port tel service-instance sel
ip address 10.10.21.10/24
ldp enable ipv4
!

```

```
interface e2
 ip mtu 1500
 ip vrf forwarding ComB
!
interface e3
 ip mtu 1500
 ip vrf forwarding ComA
!
!
P
!
hostname P
!
ip vrf management
!
mpls propagate-ttl
!
!
ip pim register-rp-reachability
!
router ldp
pw-status-tlv
advertisement-mode downstream-on-demand
!
interface e1
ip mtu 1500
label-switching
connect port tel service-instance sel
ip address 10.10.21.50/24
enable-ldp ipv4
!
interface e2
ip mtu 1500
label-switching
connect port tel service-instance sel
ip address 10.10.13.50/24
enable-ldp ipv4
!
end
```

17.5 MPLS Layer-3 eBGP VPN Configuration

В данном разделе приведены примеры конфигурации для организации сети VPN при помощи eBGP в случае, когда PE-маршрутизаторы находятся в разных автономных системах (AS).

Возможности сети VPN расширены для того чтобы была возможна реализация сценариев, когда PE-маршрутизаторы находятся в разных AS. Во всех рассмотренных случаях соединение между PE-маршрутизаторами устанавливается по eBGP. По умолчанию EBGP-VPN не разрешены.

17.5.1 Настройка eBGP между PE и ASBR

В этом примере eBGP сконфигурирован между CE и PE-маршрутизаторами. PE-маршрутизаторы по iBGP соединены с пограничными маршрутизаторами автономной системы (Autonomous System Border Router, ASBR). ASBR соединены между собой по eBGP.

Топология

На рисунке ниже приведена топология сети для данного примера.

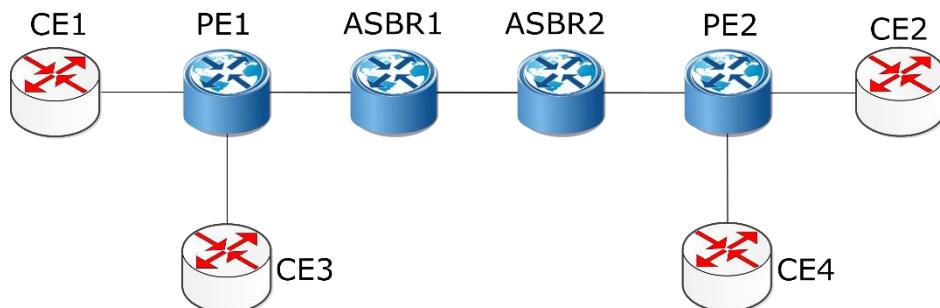


Рисунок 32

В таблицах ниже представлены команды конфигурирования маршрутизаторов CE, PE и ASBR в соответствии с топологией сети.

Настройка CE-маршрутизаторов

Таблица 82

Команда	Описание
#configure terminal	Вход в конфигурационный режим
(config)#interface e1	Вход в режим конфигурирования интерфейса
(config-if)#ip address 172.6.7.117/24	Назначение IP-адреса
(config-if)#exit	Выход из режима конфигурирования интерфейса
(config)#router bgp 65001	Определение процесса BGP маршрутизации для AS 65001
(config-router)#neighbor 172.6.7.116 remote-as 1	Определение PE-маршрутизатора как соседа. Где 172.6.7.116 - IP-адрес PE-маршрутизатора, 1 - номер AS

Для проверки настроенной конфигурации используются команды административного режима **show ip bgp neighbors**, **show ip bgp**.

Настройка PE-маршрутизаторов

Таблица 83

Команда	Описание
#configure terminal	Вход в конфигурационный режим
(config)#ip vrf IPI	Создание VRF под названием IPI
(config-vrf)#rd 1:100	Назначение RD 1:100
(config-vrf)#route-target both 100:200	Настройка импорта маршрутов между RT расширенных сообществ 100 и 200
(config-vrf)#exit	Выход из конфигурирования VRF
(config)#interface e3	Вход в режим конфигурирования интерфейса
(config-if)#ip vrf forwarding IPI	Привязка VRF под названием IPI к интерфейсу, к которому подключен CE-маршрутизатор
(config-if)#ip address 172.6.7.116/24	Назначение IP-адреса

Команда	Описание
(config-if)#exit	Выход из режима конфигурирования интерфейса
(config)#router bgp 1	Определение процесса BGP маршрутизации для AS 1
(config-router)#neighbor 172.5.6.115 remote-as 1	Добавление ASBR в качестве однорангового iBGP устройства с IP-адресом 172.5.6.115 и AS 1
(config-router)#address-family vpnv4 unicast	Вход в режим конфигурирования семейства адресов VPNv4
(config-router-af)#neighbor 172.5.6.115 activate	Активация ASBR-соседства, чтобы ASBR мог принимать маршруты сети VPN
(config-router-af)#exit-address-family	Выход из режима конфигурирования семейства адресов VPNv4
(config-router)#address-family ipv4 vrf IPI	Вход в режим конфигурирования семейства адресов IPv4 для VRF IPI
(config-router-af)#neighbor 172.6.7.117 remote-as 65001	Добавление CE-маршрутизатора в качестве однорангового eBGP устройства с IP-адресом 172.6.7.117 и AS 65001
(config-router-af)#exit-address-family	Выход из режима конфигурирования семейства адресов IPv4
(config-router)#exit	Выход из режима конфигурирования маршрутизации

Для проверки настроенной конфигурации используются команды административного режима **show ip bgp neighbors**, **show ip bgp vpnv4 all**.

Настройка ASBR1 и ASBR2

Таблица 84

Команда	Описание
#configure terminal	Вход в конфигурационный режим
(config)#ip vrf IPI	Создание VRF под названием IPI
(config-vrf)#rd 1:100	Назначение RD 1:100
(config-vrf)#route-target both 100:200	Настройка импорта маршрутов между RT расширенных сообществ 100 и 200
(config-vrf)#exit	Выход из конфигурирования VRF
(config)#interface e1	Вход в режим конфигурирования интерфейса
(config-if)#ip address 172.5.6.115/24	Назначение IP-адреса
(config-if)#exit	Выход из режима конфигурирования интерфейса
(config)#router bgp 1	Определение процесса BGP маршрутизации для AS 1
(config-router)#neighbor 172.5.6.116 remote-as 1	Добавление ASBR в качестве однорангового iBGP устройства с IP-адресом 172.5.6.116 и AS 1
(config-router)#neighbor 172.4.5.114 remote-as 2	Добавление удаленного ASBR в качестве однорангового eBGP устройства с IP-адресом 172.4.5.114 и AS 2
(config-router)#address-family vpnv4 unicast	Вход в режим конфигурирования семейства адресов VPNv4
(config-router-af)#neighbor 172.5.6.116 activate	Активация ASBR-соседства, чтобы ASBR мог принимать маршруты сети VPN
(config-router-af)#neighbor 172.4.5.114 allow-ebgp-vpn	Включение в CLI возможности установления eBGP сети VPN между двумя ASBR
(config-router-af)#neighbor 172.4.5.114	Активация eBGP ASBR для обработки маршрутов сети VPN

Команда	Описание
activate	
(config-router-af)#exit-address-family	Выход из режима конфигурирования семейства адресов VPNv4
(config-router)#exit	Выход из режима конфигурирования маршрутизации

Для проверки настроенной конфигурации используются команды административного режима **show ip bgp neighbors**, **show ip bgp vpnv4 all**.

17.5.2 Настройка eBGP между PE и RR и между ASBR

В данном примере PE-маршрутизатор соединен с Route-Reflector (RR), одним из клиентов которого является ASBR, соединенный с другими ASBR по eBGP. Конфигурация аналогична предыдущему примеру "Настройка eBGP между PE и ASBR", кроме конфигурации PE-маршрутизаторов и клиентов RR, одним из которых является ASBR. Между собой ASBR соединены по eBGP.

Топология

На рисунке ниже приведена топология сети для данного примера.

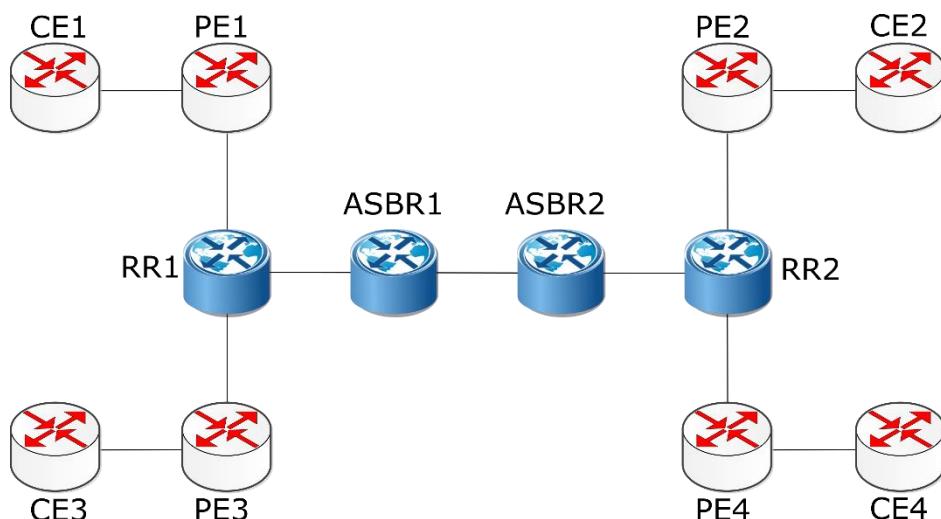


Рисунок 33

Ниже представлены команды конфигурирования маршрутизаторов CE, PE, RR и ASBR в соответствии с топологией сети.

Настройка CE-маршрутизаторов

Используются те же команды, что и в примере "Настройка eBGP между PE и ASBR".

Настройка PE-маршрутизаторов

Используются те же команды, что и в примере "Настройка eBGP между PE и ASBR", кроме того, что RR конфигурируется как одноранговое iBGP устройство, вместо ASBR.

Настройка Route Reflectors

Таблица 85

Команда	Описание
#configure terminal	Вход в конфигурационный режим
(config)#ip vrf IPI	Создание VRF под названием IPI
(config-vrf)#rd 1:100	Назначение RD 1:100

Команда	Описание
(config-vrf)#route-target both 100:200	Настройка импорта маршрутов между RT расширенных сообществ 100 и 200
(config-vrf)#exit	Выход из конфигурирования VRF
(config)#interface eth1	Вход в режим конфигурирования интерфейса
(config-if)#ip address 172.4.5.114/24	Назначение IP-адреса
(config-if)#exit	Выход из режима конфигурирования интерфейса
(config)#router bgp 1	Определение процесса BGP маршрутизации для AS 1
(config-router)#neighbor 172.5.6.116 remote-as 1	Добавление ASBR в качестве однорангового iBGP устройства с IP-адресом 172.5.6.116 и AS 1
(config-router)#neighbor 172.4.5.114 remote-as 1	Добавление ASBR в качестве однорангового iBGP устройства с IP-адресом 172.4.5.114 и AS 1
(config-router)#address-family vpnv4 unicast	Вход в режим конфигурирования семейства адресов VPNv4
(config-router-af)#neighbor 172.5.6.116 activate	Активировать PE-маршрутизатор для обработки маршрутов сети VPN
(config-router-af)#neighbor 172.5.6.116 route-reflector-client	Добавить PE-маршрутизатор, как route-reflector-client
(config-router-af)#neighbor 172.4.5.114 activate	Активация ASBR-соседства, чтобы ASBR мог принимать маршруты сети VPN
(config-router-af)#neighbor 172.4.5.114 route-reflector-client	Добавить ASBR, как route-reflector-client
(config-router-af)#exit-address-family	Выход из режима конфигурирования семейства адресов VPNv4
(config-router)#exit	Выход из режима конфигурирования маршрутизации

Для проверки настроенной конфигурации используются команды административного режима **show ip bgp neighbors**, **show ip bgp vpnv4 all**.

Настройка ASBR

Используются те же команды, что и в примере "Настройка eBGP между PE и ASBR", кроме того, что ASBR конфигурируется как одноранговое iBGP устройство, вместо RR.

17.5.3 Соединение PE-маршрутизаторов с использованием eBGP Multi-hop

В данном примере PE-маршрутизаторы подключены друг к другу напрямую с использованием eBGP multi-hop.

Между СЕ и PE-маршрутизаторами настроен eBGP. PE-маршрутизаторы настроены таким образом, чтобы между ними было соединение eBGP multi-hop. Для того чтобы соединение multi-hop работало, между PE1, Р и PE2 должен быть запущен протокол IGP.

Топология

На рисунке ниже приведена топология сети для данного примера.

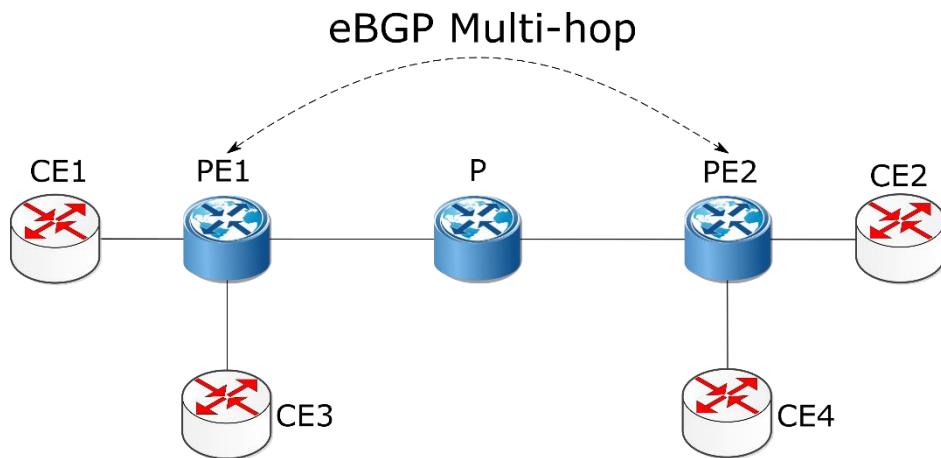


Рисунок 34

Ниже представлены команды конфигурирования маршрутизаторов СЕ и РЕ в соответствии с топологией сети.

На Р-маршрутизаторах должен быть настроен только протокол IGP (в данном примере OSPF).

Настройка СЕ-маршрутизаторов

Таблица 86

Команда	Описание
#configure terminal	Вход в конфигурационный режим
(config)#interface eth1	Вход в режим конфигурирования интерфейса
(config-if)#ip address 172.6.7.117/24	Назначение IP-адреса
(config-if)#exit	Выход из режима конфигурирования интерфейса
(config)#router bgp 65001	Определение процесса BGP маршрутизации для AS 65001
(config-router)#neighbor 172.6.7.116 remote-as 1	Определение РЕ-маршрутизатора как соседа. Где 172.6.7.116 - IP-адрес РЕ-маршрутизатора, 1 - номер AS

Для проверки настроенной конфигурации используются команды административного режима **show ip bgp neighbors**, **show ip bgp**.

Настройка РЕ-маршрутизаторов

Таблица 87

Команда	Описание
#configure terminal	Вход в конфигурационный режим
(config)#ip vrf IPI	Создание VRF под названием IPI
(config-vrf)#rd 1:100	Назначение RD 1:100
(config-vrf)#route-target both 100:200	Настройка импорта маршрутов между RT расширенных сообществ 100 и 200
(config-vrf)#exit	Выход из конфигурирования VRF
(config)#interface eth3	Вход в режим конфигурирования интерфейса
(config-if)#ip vrf forwarding IPI	Bind the interface connected to the CE router with VRF IPI.
(config-if)#ip address 172.6.7.116/24	Назначение IP-адреса
(config-if)#exit	Выход из режима конфигурирования интерфейса
(config)#router ospf 1	Определение процесса маршрутизации OSPF

Команда	Описание
(config-router)#network 172.5.6.0/24 area 0	Рекламировать сеть между PE и P-маршрутизатором для того, чтобы обеспечить multi-hop
(config-router)#exit	Выход из режима конфигурации маршрутизации OSPF
(config)#router bgp 1	Определение процесса BGP маршрутизации для AS 1
(config-router)#neighbor 172.4.5.114 remote-as 2	Определение PE-маршрутизатора, как соседа. Здесь 172.4.5.114 - IP-адрес удаленного PE-маршрутизатора, 2 - номер AS
(config-router)#neighbor 172.4.5.114 ebgp-multi-hop 255	Установление PE-маршрутизатора в качестве однорангового устройства eBGP
(config-router)#address-family vpnv4 unicast	Вход в режим конфигурирования семейства адресов VPNv4
(config-router-af)#neighbor 172.4.5.114 allow-ebgp-vpn	Настройка удаленного PE-маршрутизатора для разрешения eBGP сетей VPN
(config-router-af)#neighbor 172.4.5.114 activate	Активация удаленного PE-маршрутизатора, чтобы он мог получать маршруты сети VPN
(config-router-af)#exit-address-family	Выход из режима конфигурирования семейства адресов VPNv4
(config-router)#address-family ipv4 vrf IPI	Вход в режим конфигурирования семейства адресов IPv4 для VRF IPI
(config-router-af)#neighbor 172.6.7.117 remote-as 65001	Определение CE-маршрутизатора, как соседа с IP-адресом 172.6.7.117 и номером AS 65001
(config-router-af)#exit-address-family	Выход из режима конфигурирования семейства адресов IPv4
(config-router)#exit	Выход из режима конфигурирования маршрутизации

Для проверки настроенной конфигурации используются команды административного режима **show ip bgp neighbors**, **show ip bgp vpnv4 all**.

17.5.4 Соединение PE-маршрутизаторов с RR через RR, используя eBGP multi-hop

В данном примере PE-маршрутизаторы подсоединены к Route-Reflector (RR), которые подсоединенны к другим RR, используя eBGP-multi-hop.

Конфигурация аналогична предыдущему примеру "Соединение PE-маршрутизаторов с использованием eBGP Multi-hop", кроме того, что PE-маршрутизаторы подсоединенны к RR по iBGP. EBGP multi-hop соединения остаются только между RR.

Топология

На рисунке ниже приведена топология сети для данного примера.

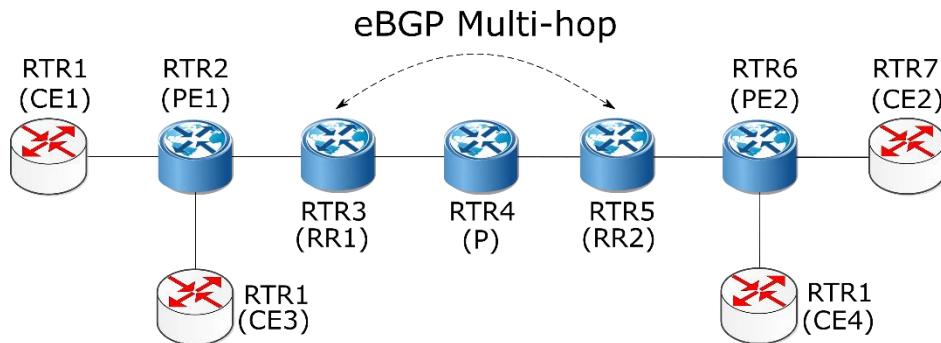


Рисунок 35

Ниже представлены команды конфигурирования маршрутизаторов CE, PE и RR в соответствии с топологией сети.

На Р-маршрутизаторах должен быть настроен только протокол IGP (в данном примере OSPF).

Настройка CE-маршрутизаторов

Настраивается аналогично примеру "Соединение PE-маршрутизаторов с использованием eBGP Multi-hop".

Настройка PE-маршрутизаторов

Настраивается аналогично примеру "Соединение PE-маршрутизаторов с использованием eBGP Multi-hop", кроме того, что у PE-маршрутизаторов есть только одно iBGP соединение с RR.

Настройка Route Reflectors

Таблица 88

Команда	Описание
#configure terminal	Вход в конфигурационный режим
(config)#ip vrf IPI	Создание VRF под названием IPI
(config-vrf)#rd 1:100	Назначение RD 1:100
(config-vrf)#route-target both 100:200	Настройка импорта маршрутов между RT расширенных сообществ 100 и 200
(config-vrf)#exit	Выход из конфигурирования VRF
(config)#interface eth1	Вход в режим конфигурирования интерфейса
(config-if)#ip address 172.5.6.115/24	Назначение IP-адреса
(config-if)#exit	Выход из режима конфигурирования интерфейса
(config)#router bgp 1	Определение процесса BGP маршрутизации для AS 1
(config-router)#neighbor 172.5.6.116 remote-as 1	Добавление ASBR в качестве однорангового iBGP устройства с IP-адресом 172.5.6.116 и AS 1
(config-router)#neighbor 172.3.4.113 remote-as 2	Добавление удаленного RR в качестве однорангового iBGP устройства с IP-адресом 172.3.4.113 и AS 2
(config-router)#neighbor 172.3.4.113 ebgp-multi-hop 255	Назначение удаленного RR-маршрутизатора в качестве однорангового устройства eBGP-multi-hop
(config-router)#address-family vpnv4 unicast	Вход в режим конфигурирования семейства адресов VPNv4

Команда	Описание
(config-router-af)#neighbor 172.3.4.113 allow-ebgp-vpn	Настройка удаленного RR, чтобы разрешить EBGP сети VPN
(config-router-af)#neighbor 72.3.4.113 activate	Активация соседства, чтобы удаленный RR мог принимать маршруты сети VPN
(config-router-af)#neighbor 172.5.6.116 activate	Активация PE-маршрутизатора для обработки маршрутов сети VPN
(config-router-af)#neighbor 172.5.6.116 route-reflector-client	Добавление PE-маршрутизатора в качестве route-reflector-client
(config-router-af)#exit-address-family	Выход из режима конфигурирования семейства адресов VPNv4
(config-router)#exit	Выход из режима конфигурирования маршрутизации
(config)#router ospf 1	Определение процесса маршрутизации OSPF
(config-router)#network 172.4.5.0/24 area 0	Рекламировать сеть между PE и P-маршрутизатором для того, чтобы обеспечить multi-hop
(config-router)#exit	Выход из режима конфигурирования маршрутизации

Для проверки настроенной конфигурации используются команды административного режима **show ip bgp neighbors**, **show ip bgp vpng4 all**.

18 Virtual Private LAN Service

Функционал VPLS L2VPN позволяет создавать распределённые LAN-сети поверх IP/MPLS-сети. В отличие от сервиса VPWS (Virtual Private Wire Service), сервис VPLS позволяет создавать не только сети типа точка-точка, но и полносвязные L2-сети. Маршрутизаторы EcoRouter также поддерживают тип сервиса H-VPLS, позволяющий терминировать на пограничном устройстве VPLS-сети не только физический канал, но и pseudowire, представляя собой объединение сервисов VPWS (L2-circuit) и VPLS.

В терминологии VPLS существует несколько типов устройств, каналов и интерфейсов:

- PW (Pseudowire) – виртуальный канал между двумя PE-устройствами или устройством MTU и PE;
- PE (Provider Edge) – граничный маршрутизатор сети провайдера, на котором терминируется сервис VPLS;
- MTU-r (Multi-Tenant Unit router) – маршрутизатор, термирующий VPWS-каналы в сторону сети провайдера и физические каналы (или VLAN) в сторону клиентов;
- CE (Customer Edge) – оборудование клиента, подключающееся к оборудованию провайдера – PE или MTU;
- AC (Access circuit) – интерфейс PE в сторону клиента. Может терминировать физический канал или L2-circuit. В EcoRouterOS под физическим каналом следует понимать порт, с привязанным service-instance и инкапсуляцией untagged или dot1q;
- VC (Virtual circuit) – интерфейс PE в сторону другого PE сети. Представляет собой односторонний виртуальный канал;
- VSI (Virtual Switch Instance) – виртуальный Ethernet-bridge, термирующий AC со стороны клиентов и VC со стороны сети провайдера. VPLS-instance – синоним VSI.

На схеме ниже изображены основные устройства и каналы VPLS-сети.

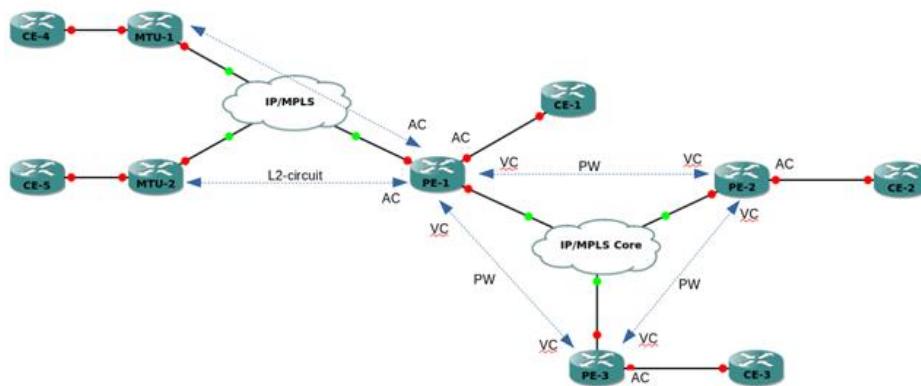


Рисунок 36

В реализации сервиса VPLS в EcoRouterOS используется сигнализация LDP (Martini). Сигнализация BGP (Kompella) не поддерживается.

18.1 Общие требования для работы VPLS (Martini)

Сервис VPLS работает поверх IP/MPLS-сети, соответственно, для организации его работы необходимо, чтобы между устройствами PE была IP-связность, а также функционировал MPLS-транспорта на основе LDP. Между устройствами PE должна быть установлена tLDP-сессия, используемая для обмена сервисными MPLS-метками.

Аналогичные требования существуют для связности устройств PE и MTU-r. Сами устройства MTU-r могут быть в разных сетях и не иметь IP-связности друг с другом.

18.2 Схема с одним PE, терминирующим L2-circuit

Простейшая схема использования сервиса VPLS выглядит следующим образом (см. рисунок ниже).

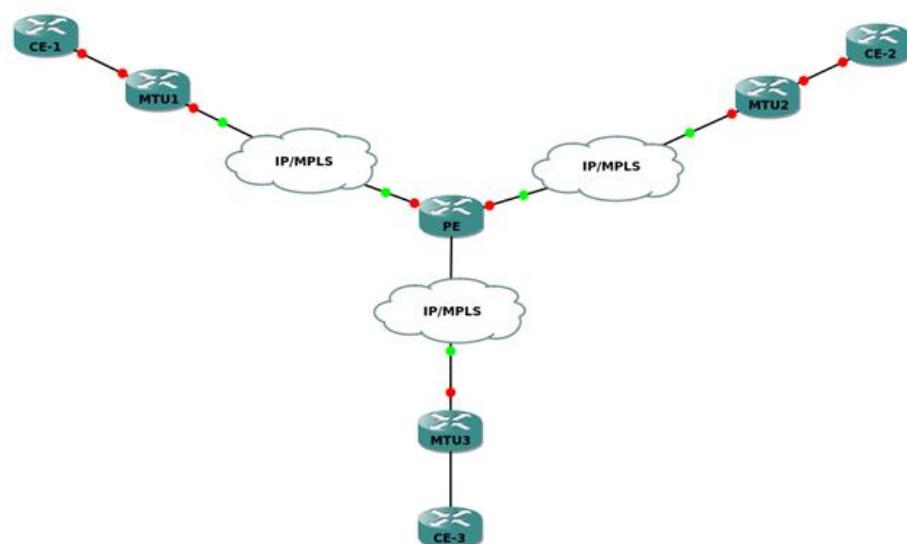


Рисунок 37

Устройство PE термирует в одном VPLS-домене несколько каналов L2-circuit, в результате чего устройства CE находятся в одной LAN-сети.

Настройка MTU-r

На устройствах MTU-r настраивается сервис L2-circuit. Эти устройства ничего не знают о VPLS и в принципе не обязаны его поддерживать. Пример настройки L2-circuit можно посмотреть в соответствующем разделе.

Настройка PE

На PE должны быть предварительно настроены:

IP-интерфейсы (см. раздел Виды интерфейсов),

loopback.0 (см. раздел Виды интерфейсов),

IGP-протокол,

LDP (см. раздел Multiprotocol Label Switching),

tLDP с MTU-r-устройствами.

Для создания L2-circuit используются команды конфигурационного режима:

```
ecorouter(config)#mpls l2-circuit vc10 10 11.11.11.11
ecorouter(config)#mpls l2-circuit vc20 20 22.22.22.22
ecorouter(config)#mpls l2-circuit vc30 30 33.33.33.33
```

Где 11.11.11.11, 22.22.22.22 и 33.33.33.33 – это loopback.0 адреса устройств MTU-r.

VSI создаётся командой конфигурационного режима:

```
ecorouter(config)#vpls-instance test100 100
```

Где 100 – это ID VSI. После ввода команды, выполняется переход в контекст VPLS-instance **ecorouter(config-vpls)#[/]**, где выполняются настройки VPLS-instance.

Для добавления L2-circuit в VSI используются команды в контексте vpls-instance:

```
ecorouter(config-vpls)#member vpls-vc vc10 ethernet
ecorouter(config-vpls)#member vpls-vc vc20 ethernet
ecorouter(config-vpls)#member vpls-vc vc30 ethernet
```

18.3 Схема с тремя PE, L2-circuit и Service-instance

Данная схема предполагает полную связность между PE-устройствами, объединяющими клиентов в одну LAN-сеть. Клиенты подключаются к сети физическим каналом (CE2, CE3) и по L2-circuit (CE1).

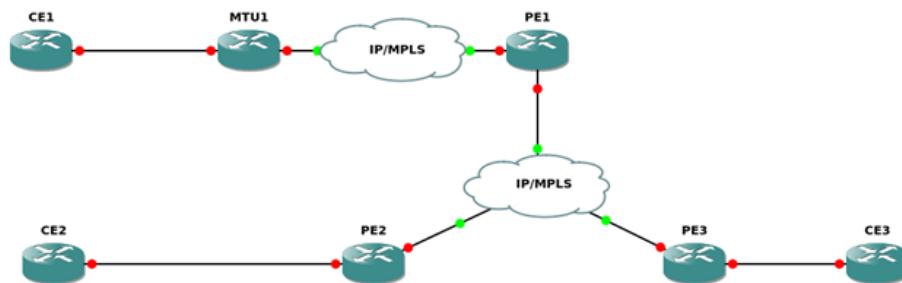


Рисунок 38

Настройка MTU-r

На устройствах MTU-r настраивается сервис L2-circuit. К данным устройствам нет требований по поддержке VPLS. Пример настройки L2-circuit можно посмотреть в разделе Multiprotocol Label Switching.

Настройка PE1

На PE1 должны быть предварительно настроены:

IP-интерфейсы (см. раздел Виды интерфейсов),

loopback.0 (см. раздел Виды интерфейсов),

IGP-протокол,

LDP (см. раздел Multiprotocol Label Switching),

tLDP с MTU-r, PE2 и PE3.

Для создания L2-circuit используется команда конфигурационного режима **mpls l2-circuit vc10 10 11.11.11.11**. Где 11.11.11.11 – это loopback.0 адрес устройства MTU-r.

VSI создаётся командой конфигурационного режима **vpls-instance test100 100**, где 100 – это ID VSI (должно совпадать у всех PE).

После ввода команды, выполняется переход в контекст VPLS-instance **ecorouter(config-vpls)#**, где выполняются настройки VPLS-instance.

Для добавления L2-circuit в VSI используется команда в контексте VPLS-instance **member vpls-vc vc10 ethernet**.

Для добавления VPLS-соседей PE2 и PE3 используются следующие команды контекста VPLS-instance.

```
PE1 (config-vpls) # signaling ldp
PE1 (config-vpls-sig) #vpls-peer 2.2.2.2
PE1 (config-vpls-sig) #vpls-peer 3.3.3.3
```

Где 2.2.2.2, 3.3.3.3 – это loopback.0 адреса устройств PE2 и PE3 соответственно.

Настройка PE2

На PE2 должны быть предварительно настроены:

IP-интерфейсы (см. раздел Виды интерфейсов),

loopback.0 (см. раздел Виды интерфейсов),

IGP-протокол,

LDP (см. раздел Multiprotocol Label Switching),

tLDP с PE1 и PE3.

VSI создаётся командой **vpls-instance test100 100**, где 100 – это ID VSI, значение которого должны совпадать у всех PE.

После ввода команды, выполняется переход в контекст VPLS-instance **ecorouter(config-vpls)#**, где выполняются настройки vpls-instance.

Для добавления сервисных интерфейсов в VSI используются команды в контексте VPLS-instance **member port te2 service-instance vpls**, где te2 – это номер порта, а vpls – это имя сервисного интерфейса, который должен быть создан на соответствующем порту.

Для добавления VPLS-соседей PE2 и PE3 используются следующие команды контекста VPLS-instance.

```
PE1 (config-vpls) # signaling ldp
PE1 (config-vpls-sig) #vpls-peer 1.1.1.1
PE1 (config-vpls-sig) #vpls-peer 3.3.3.3
```

Где 2.2.2.2, 3.3.3.3 – это loopback.0 адреса устройств PE2 и PE3 соответственно.

18.4 Команды просмотра VPLS

Для просмотра состояния VPLS-instance используются команды режима администрирования, перечисленные ниже.

Команда **show vpls-instance** показывает основные параметры VSI.

ecorouter#show vpls-instance					
Name	VPLS-ID	Type	MPeers	SPeers	SIG-Protocol

test100	100	Ethernet	0	3	N/A
---------	-----	----------	---	---	-----

Команда **show vpls-instance detail** показывает более подробную информацию о VPLS-instance.

```
ecorouter#show vpls-instance detail
Virtual Private LAN Service Instance: test100, ID: 100
SIG-Protocol: LDP
Learning: Enabled
Group ID: 0, VPLS Type: Ethernet, Configured MTU: 9714
Description: none
Operating mode: Raw
Configured interfaces:
  Interface: vi-100
Mesh Peers: 2.2.2.2 (Up)
            3.3.3.3 (Up)
Spoke Peers: vc10 (Up)
```

Для просмотра таблицы MAC-адресов в VSI используется команда **show vpls mac-table <NAME>**, где **NAME** – это имя VPLS-instance.

```
ecorouter#show vpls mac-table test100
VPLS Aging time is 60 sec
  L2
  Address      Port      Type      Age
  -----
  0050.7966.6801  te2    Dynamic    11
  0050.7966.6800  te0    Dynamic    11
```

18.5 Дополнительные настройки VPLS

Aging time

По умолчанию запись в таблице коммутации хранится 60 секунд. Время хранения записи можно настраивать для каждого VPLS-instance. Для этого используется команда контекста VPLS-instance **aging-time <NUM>**, где **NUM** - время хранения в секундах.

```
ecorouter(config)#vpls-instance test200 200
ecorouter(config-vpls)#aging-time 300
<60-86400> Time in seconds
```

MTU

По умолчанию MTU (maximum transmission unit) на VPLS-instance - 9710 байт. MTU настраивается для каждого VPLS-instance. Для этого используется команда контекста VPLS-instance **vpls-mtu <NUM>**, где **NUM** - максимальный размер data unit в байтах.

```
ecorouter(config)#vpls-instance test200 200
ecorouter(config-vpls)#vpls-mtu 9000
<576-65535> Allowed MTU range
```

Для согласования peer-соседства между двумя маршрутизаторами, MTU каждого из них на VPLS-instance должен совпадать. Для корректной работы l2circuit (в случае привязки к VPLS-instance), MTU на устройствах PE и MTU-r должны совпадать.

19 VRRP

VRRP – Virtual Router Redundancy Protocol, протокол резервирования L3 устройств в сетях IPv4/6.

Протокол VRRP решает задачу по резервированию L3-интерфейса, выполняющего роль next-hop'a для IPv4 маршрутов. Принцип работы протокола подразумевает наличие в сегменте некоторого множества маршрутизаторов, один из которых исполняет роль владельца общего виртуального IP-адреса. Остальные маршрутизаторы являются резервными и принимают на себя роль мастера только в случае, если первоначальный мастер вышел из строя. При этом все устройства прослушивают входящий трафик на предмет служебных VRRP сообщений и сравнивают значение собственного приоритета с соответствующими значениями в сообщениях соседей.

Маршрутизатор, имеющий наибольшее значение приоритета, принимает роль мастера.

Только маршрутизатор, выполняющий роль мастера, имеет право на обработку транзитного трафика, отправляемого на общий виртуальный MAC-адрес, а также только он имеет право отвечать на ARP запросы, адресованные владельцу виртуального IP-адреса.

19.1 Базовая настройка

Для настройки протокола VRRP необходимо выполнить следующие шаги.

Шаг 1. Перейдите из конфигурационного режима в контекстный режим конфигурирования протокола с помощью команды **router vrrp <VRRP-ID> <NAME>**, где VRRP-ID - это номер группы, имеющий значение в пределах от 1 до 255, а NAME - имя интерфейса, участвующего в группе.

Шаг 2. Укажите IP-адрес, который будет использован в качестве виртуального. Для этого введите команду **virtual-ip <IPv4>**. В случае если роль мастера необходимо назначить конкретному маршрутизатору, например, наиболее производительному в сегменте, удобно назначать виртуальный IP равным реальному транспортному адресу. При этом значение приоритета автоматически становится равным 255, что означает безусловное принятие роли мастера при корректной работе устройства.

Шаг 3. Если это необходимо, то сконфигурируйте явный приоритет маршрутизатора. Значение приоритета устанавливается с помощью команды **priority <значение>**. При этом значение имеет вид числа в диапазоне от 1 до 254 и по умолчанию равно 100.

Шаг 4. Активируйте работу протокола командой **enable**.

После активации протокола при каждом внесении изменений необходимо останавливать его работу командой **disable**.

19.2 Дополнительные функции

В реализации EcoRouterOS VRRP также поддерживает ряд функций, описанных ниже.

19.2.1Функция preempt-mode

Если необходимо, чтобы вышедший из строя мастер по возвращению в работу игнорировал тот факт, что назначенное ему значение приоритета выше, чем у текущего мастера, необходимо отключить режим вытеснения командой **preempt-mode false**. В этом режиме вернувшийся к работе маршрутизатор с более высоким заданным приоритетом не будет анонсировать служебные сообщения, что в противном случае привело бы к вытеснению текущего мастера. Для возвращения режима вытеснения применяется команда **preempt-mode true**.

19.2.2Функция switch-back-delay

Для задания времени ожидания, в течение которого вернувшийся к работе маршрутизатор с более высоким приоритетом не будет анонсировать служебные сообщения, применяется команда **switch-back-delay <1-500000>**, где единственный аргумент - продолжительность ожидания, выраженная в ms. Данная функция не является дополнением к вышеописанной, а применяется в качестве альтернативного поведения в случаях, когда необходимо избежать частую смену ролей в нестабильной топологии.

19.2.3Функция circuit-failover

Для отслеживания состояния какого-либо сетевого соединения маршрутизатора, при выходе из строя которого потребуется смена роли устройства, используется команда **circuit-failover <Имя наблюдаемого интерфейса> <декремент приоритета>**, где последний аргумент - число, на которое уменьшается значение приоритета маршрутизатора. Пример использования данной функции - отслеживание состояния соединений с маршрутизаторами, которые находятся выше в иерархии. В случае VRRP-мастера потеря соединения с таким маршрутизатором приводит к тому, что устройство не может обслуживать трафик и вынуждено передать свою роль соседу.

19.2.4Функция accept-mode

Согласно RFC 5798, по умолчанию мастер отбрасывает трафик, адресованный виртуальному IP-адресу непосредственно. Однако в ряде случаев необходимо, чтобы такой трафик обрабатывался. Для изменения поведения по умолчанию необходимо воспользоваться командой **accept-mode {false | true}**. Использование аргумента **true** приводит к переходу в режим обработки трафика, адресованного виртуальному IP. Аргумент **false** применяется для отключения этого режима.

19.2.5Функция advertisement-interval

Для изменения интервала отправки VRRP-сообщений необходимо воспользоваться командой **advertisement-interval <5-4096>**, где в качестве единственного аргумента указывается продолжительность интервала, выраженная в сенитисекундах (1 cs = 0.01 s).

19.2.6Функция vrrp vmac

Согласно RFC 5798, по умолчанию виртуальный MAC-адрес указывается в Ethernet-заголовке служебных VRRP-сообщений в поле Source MAC Address. Для повышения эффективности диагностики в указанное поле можно устанавливать значение реального MAC-

адреса сетевого устройства, сформировавшего служебный пакет. Для этого в конфигурационном режиме следует использовать команду **vrrp vmac {enable | disable}**.

19.3 Поддерживаемые версии протокола

На данный момент существует 3 версии протокола VRRP, из которых реально используются только v2 и v3, при этом, по ряду причин, наиболее актуальной является v2. EcoRouterOS поддерживает обе версии протокола, при этом по умолчанию используется только v3.

Если требуется использовать EcoRouter в одном VRRP-домене с маршрутизаторами, которые не поддерживают VRRP v3, в EcoRouterOS необходимо включить поддержку v2. Для этого следует выполнить два действия:

- в конфигурационном режиме ввести команду: **ecorouter(config)#vrrp compatible-v2;**
- в контекстном режиме конфигурации работы протокола в контексте конкретного интерфейса применить команду: **ecorouter(config-router)#v2-compatible.**

При этом EcoRouter будет передавать VRRP-анонсы в формате v2 и v3 одновременно, то есть по два сообщения один раз в интервал. Аналогично анонсированию маршрутизатор будет обрабатывать и учитывать все служебные сообщения от соседей, в том числе и сообщения в формате v3. Во избежание ошибок дизайна необходимо применять только одну версию протокола на всех маршрутизаторах других производителей, находящихся в одном VRRP-домене с EcoRouter. Под VRRP-доменом здесь подразумевается множество маршрутизаторов, обслуживающих общий виртуальный IP-адрес в конкретном локальном сегменте и анонсирующих общее значение VRRP-ID.

19.4 Пример конфигурации

VRRP протокол часто применяется для резервирования шлюза по умолчанию в пользовательском сегменте сети. При этом хосты, выполняющие роль пользователей, имеют минимальную конфигурацию протокола IP, предполагающую наличие незначительного количества сетей, подключенных непосредственно, и маршрутизатора в качестве узла, обслуживающего передачу трафика в направлении всех остальных пунктов назначения. Если сегмент обслуживается только одним маршрутизатором его выход из строя в отношении конечных узлов означает, что трафик за пределы сегмента перестанет отправляться. Применение двух маршрутизаторов с одинаковым значением IP-адреса приводит к конфликту в отсутствии дополнительных средств контроля. В качестве такого средства применяется VRRP протокол.

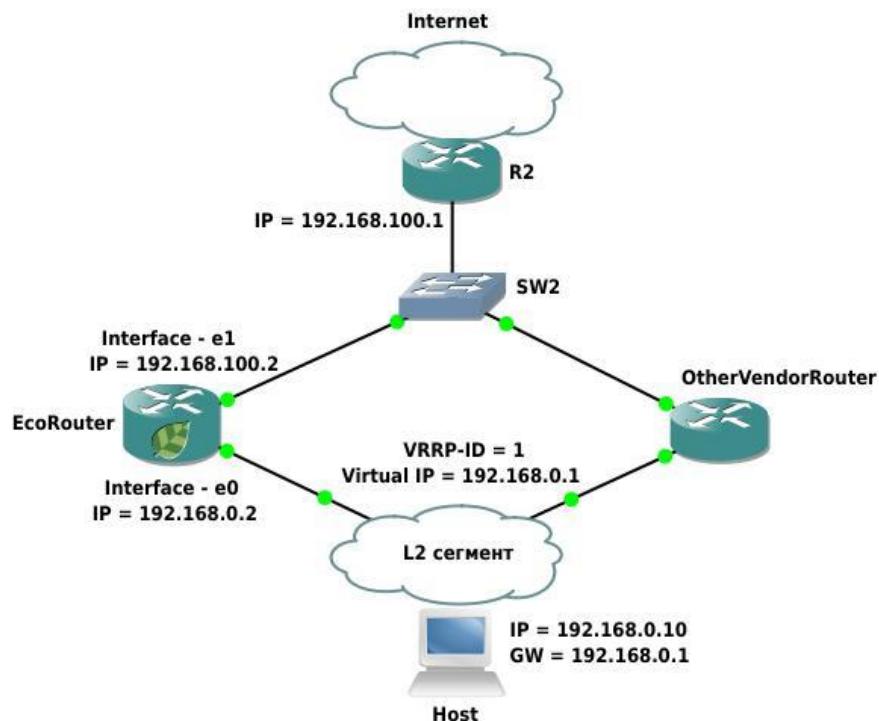


Рисунок 39

В приведенной схеме топологии в подсети для VRRP-протокола задействованы 2 маршрутизатора: EcoRouter и маршрутизатор другого производителя (OtherVendorRouter). Маршрутизатор R2 фигурирует в качестве пограничного для AS узла и является шлюзом по умолчанию для обоих маршрутизаторов, реализующих VRRP протокол. Его настройка не предполагает использование VRRP протокола и по этой причине выходит за рамки данной статьи. Оба VRRP-маршрутизатора подключены к L2-сегменту, обслуживающему подсеть 192.168.0.0/24. В данном сегменте присутствует конечный хост, имеющий две маршрутных записи: маршрут в непосредственно подключенную сеть 192.168.0.0/24, а также маршрут по умолчанию, где в качестве шлюза выступает устройство с адресом 192.168.0.1. На маршрутизаторе другого производителя выполнена минимальная настройка, обеспечивающая работу VRRP-протокола v2, при которой значение приоритета маршрутизатора оставлено равным значению по умолчанию (100), значение обслуживаемого виртуального IP - 192.168.0.1, а ID сегмента - 1. Его собственный IP-адрес имеет значение 192.168.0.3. EcoRouter также выступает в качестве VRRP-маршрутизатора, однако имеет более сложную настройку, которая предполагает работу протокола v2, выставленный пользователем более высокий приоритет, временную задержку при возвращении, а также отслеживание состояния интерфейса e1.

Конфигурация EcoRouter:

Задание имени устройства.

```
ecorouter(config):hostname EcoRouter
```

Включение VRRP

```
ecorouter(config)#vrrp compatible-v2 enable
```

Включение протокола, задание группы и имени интерфейса.

```
ecorouter(config)#router vrrp 1 e0
```

Задание виртуального адреса.

```
virtual-ip 192.168.0.1
```

Задание приоритета для этого маршрутизатора.

```
ecorouter(config-router) #priority 150
```

Включение отслеживания интерфейса.

```
ecorouter(config-router) #circuit-failover e1 100
```

Задание времени ожидания, после которого восстановятся анонсы.

```
ecorouter(config-router) #switch-back-delay 5000
```

Включение поддержки совместимости с протоколом второй версии.

```
ecorouter(config-router) #v2-compatible
```

Настройка интерфейсов и портов.

```
ecorouter(config)#interface e0
ecorouter(config-if)#ip address 192.168.0.2/24
ecorouter(config)#interface e1
ecorouter(config-if)#ip address 192.168.100.2/24
ecorouter(config)#port ge0/0
ecorouter(config-port)#service-instance ge0/0-e0
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface e0
ecorouter(config)#port ge0/1
ecorouter(config-port)#service-instance ge0/1-e0
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface e1
```

В результате описанных действий в качестве мастера будет выбран EcoRouter (по причине более высокого значения priority). В дальнейшем в случае, если его интерфейс «e1», используемый для подключения к вышестоящему маршрутизатору, не сможет продолжать передачу трафика, приоритет EcoRouter будет понижен до значения 50.

Приоритет второго маршрутизатора в этом случае станет наибольшим в сегменте, и он сможет продолжить обработку трафика до возвращения EcoRouter.

Когда связь с вышестоящим маршрутизатором будет восстановлена, EcoRouter запустит таймер ожидания, равный 5 секундам, после чего начнет вещание VRRP-сообщений, вынудив соседа сменить роль и перестать отвечать на ARP-запросы, отправляемые владельцем IP-адреса 192.168.0.1.

19.5 Известные особенности взаимодействия EcoRouter с оборудованием других производителей

Реализация протокола VRRP в EcoRouterOS стремится к максимальному соответствию RFC-документации, однако существует ряд вопросов, связанных как с реализацией EcoRouterOS, так и реализацией других производителей, проявление которых может привести к неожиданному для пользователя поведению:

- согласно RFC 5798, маршрутизатор, выполняющий роль резервного, при получении служебных сообщений от соседей принимает во внимание только значение поля приоритета. Значение транспортного адреса принимается во внимание только

маршрутизаторами, выполняющими роль мастера. Однако, данный принцип может быть нарушен другими производителями, что приводит к тому, что два и более маршрутизаторов, обслуживающих один сегмент, могут принять роль мастера со всеми вытекающими отсюда конфликтами;

- согласно RFC 5798, маршрутизатор, выполняющий роль резервного, не должен обрабатывать трафик, отправляемый на общий виртуальный MAC-адрес. В EcoRouterOS данный принцип соблюден, что необходимо учитывать в дизайне сети так же как и поведение маршрутизаторов других производителей;
- в реализации EcoRouterOS отсутствует возможность применения авторизации в работе VRRP;
- в реализации EcoRouterOS отсутствует возможность анонсирования множества IP-адресов в качестве виртуальных.

20 BFD

20.1 Протокол BFD

Bidirectional Forwarding Detection (BFD) – это протокол, созданный для быстрого обнаружения падения линков между маршрутизаторами. BFD позволяет быстрее обнаружить потерю связности в сравнении с обычными механизмами, которые используют протоколы маршрутизации. BFD, как и протоколы маршрутизации, использует обмен Hello-сообщениями, но с гораздо меньшими интервалами отправки, измеряющимися в десятках миллисекунд (в то время как для протоколов маршрутизации интервалы для отправки Hello-сообщений измеряются десятками секунд). Протокол BFD часто применяют совместно с функционалом LFA для быстрого переключения на резервный маршрут (подробнее об LFA см. раздел "Loop-Free Alternate (LFA) в OSPF").

Команды для настройки BFD на EcoRouter приведены ниже:

Таблица 89

Команда	Описание
bfd disable	Команда вводится в контекстном конфигурационном режиме (config-if). В результате выполнения этой команды на интерфейсе выключаются все bfd-сессии (переводятся в состояние Admin-Down). Значение по умолчанию: enabled
bfd echo	Команда вводится в конфигурационном режиме (config). В результате выполнения этой команды для всех bfd-сессий будет включена echo-функция с параметрами по умолчанию. Значение по умолчанию: disabled
bfd echo interval <1-4294967>	Команда вводится в контекстном конфигурационном режиме (config-if). В результате выполнения этой команды для всех bfd-сессий на интерфейсе будет установлен интервал отправки echo-сообщений в миллисекундах. Значение по умолчанию: 1000
bfd interval <25-999> minrx <25-999> multiplier <3-50>	Команда вводится в контекстном конфигурационном режиме (config-if). В результате выполнения этой команды для всех bfd-сессий на интерфейсе будут установлены: интервал отправки bfd-control сообщений в миллисекундах, ожидаемый интервал приёма bfd-control сообщений в миллисекундах, количество пропущенных сообщений, после которого сессия считается порванной. Значения по умолчанию: 250/250/3
bfd all-interfaces	Команда вводится в контекстном конфигурационном режиме (config-router). В результате выполнения этой команды будут установлены bfd-сессии со всеми OSPF-соседями в рамках соответствующего OSPF-процесса

Команды просмотра для протокола BFD на EcoRouter приведены ниже:

Таблица 90

Команда	Описание
ecorouter#show bfd BFD ID: 00 Start Time:Tue Nov 21 08:45:34 2017 BFD Admin State: UP Number of Sessions: 1 Slow Timer: 2000 Image type: MONOLITHIC Echo Mode: Disabled BFD Notifications disabled Next Session Discriminator: 2	Показать информацию о глобальных настройках BFD. Start Time - время старта процесса oamd; BFD Admin State - административное состояние протокола на устройстве; Number of Sessions - количество активных сессий;

Команда	Описание
	<p>Slow Timer - значение slow таймера; Image type - тип обработки hello-пакетов (монолитный - производится одним процессом, распределенный - производится несколькими процессами); Echo Mode - состояние echo-функции (включена/выключена); BFD Notifications - состояние уведомлений (включена/выключена); Next Session Discriminator - идентификатор следующей сессии, которая будет поднята.</p>
<pre>ecorouter#show bfd interface Interface: loopback.0 ifindex: 8 state: UP Interface level configuration: NO ECHO, NO SLOW TMR Timers in Milliseconds Min Tx: 250 Min Rx: 250 Multiplier: 3 Interface: te0 ifindex: 9 state: UP Interface level configuration: NO ECHO, NO SLOW TMR Timers in Milliseconds Min Tx: 250 Min Rx: 250 Multiplier: 3</pre>	<p>Показать информацию о настройках BFD на всех интерфейсах, на которых включен этот протокол.</p> <p>Interface - имя интерфейса; ifindex - системный номер интерфейса; state - состояние интерфейса; Interface level configuration - настройки BFD для интерфейса; Min Tx - интервал отправки bfd-control сообщений; Min Rx - ожидаемый интервал приёма bfd-control сообщений; Multiplier - количество пропущенных сообщений, после которого сессия считается порванной</p>
<pre>ecorouter#show bfd session Sess-Idx Remote-Disc Lower-Layer Sess-Type Sess-State UP- Time Remote-Addr 1 1 IPv4 Single-Hop Up 01:12:50 10.1.1.1/32 4 1 IPv4 Single-Hop Up 00:00:01 20.1.1.1/32 Number of Sessions: 2</pre>	<p>Показать информацию обо всех активных bfd-сессиях.</p> <p>Sess-Idx - локальный id сессии; Remote-Disc - id сессии на удаленном устройстве; Lower-Layer - инкапсулирующий протокол;</p>

Команда	Описание
<pre>ecorouter#show bfd session detail ===== Session Interface Index : 9 Session Index : 1 Lower Layer : IPv4 Version : 1 Session Type : Single Hop Session State : Up Local Discriminator : 1 Local Address : 10.1.1.2/32 Remote Discriminator : 1 Remote Address : 10.1.1.1/32 Local Port : 49152 Remote Port : 3784 Options : Diagnostics : None Timers in Milliseconds : Min Tx: 250 Min Rx: 250 Multiplier: 3 Neg Tx: 250 Neg Rx: 2000 Neg detect mult: 3 Min echo Tx: 1000 Min echo Rx: 1000 Neg echo intrvl: 0 Storage type : 2 Sess down time : 00:00:00 Sess discontinue time : 00:00:00 Bfd GTSM Disabled Bfd Authentication Disabled Counters values: Pkt In : 0000000000007f5f Pkt Out : 0000000000007f5a Echo Out : 0000000000000000 UP Count : 1 UPTIME : 01:58:53 Protocol Client Info: OSPF-> Client ID: 4 Flags: 4 ----- Number of Sessions: 1</pre>	<p>Sess-Type - тип сессии (single/multi); Sess-State - состояние сессии; UP-Time - up-time сессии; Remote-Addr - адрес интерфейса удаленного маршрутизатора, с которым установлена сессия; Number of Sessions - количество активных сессий</p> <p>Показать детальную информацию обо всех активных bfd-сессиях. Session Interface Index - системный номер локального интерфейса; Lower Layer - инкапсулирующий протокол; Session Type - тип сессии (single/multi); Local Discriminator - локальный id сессии; Remote Discriminator - id сессии на удаленном устройстве; Local Port - локальный UDP-порт; Session Index - локальный id сессии; Session State - состояние сессии; Local Address - адрес интерфейса локального маршрутизатора, на котором установлена сессия; Remote Address - адрес интерфейса удаленного маршрутизатора, с которым установлена сессия; Remote Port - удаленный UDP-порт; Min Tx/Neg Tx - локальный/удаленный интервал отправки bfd-control сообщений; Min Rx/Neg Rx - локальный/удаленный ожидаемый интервал приёма bfd-control сообщений;</p>

Команда	Описание
	<p>Multiplier/Neg detect multi - количество пропущенных сообщений, после которого сессия считается порванной. Значения на локальном и удаленном роутерах;</p> <p>Min echo Tx/Min echo Rx - локальный/удаленный интервал отправки echo-сообщений;</p> <p>Sess down time - время падения сессии;</p> <p>Sess discontinue time - время, на протяжении которого сессия была в состоянии down;</p> <p>Bfd GTSM - состояние функции GTSM;</p> <p>Bfd Authentication - состоянии функции аутентификации;</p> <p>Pkt In - количество пришедших BFD-пакетов;</p> <p>Pkt Out - количество отправленных BFD-пакетов;</p> <p>Echo Out - количество отправленных echo-пакетов;</p> <p>UPTIME - up-time сессии;</p> <p>Protocol Client Info - информация о протоколе, посредством которого установлена сессия;</p> <p>Number of Sessions -- количество активных сессий</p>
<pre>ecorouter#show bfd session 10.1.1.2 10.1.1.1 Session Interface Index : 9 Session Index : 1 Lower Layer : IPv4 Session Type : Single Hop Session State : Up Local Discriminator : 1 Remote Discriminator : 1 Local Address : 10.1.1.2/32 Remote Address : 10.1.1.1/32 Local Port : 49152 Remote Port : 3784 Timers in Milliseconds : Min Tx: 250 Min Rx: 250 Multiplier: 3 UP Count : 1 UPTIME : 03:10:33</pre>	<p>Показать информацию о сессии между конкретным локальным интерфейсом с указанием его ip-адреса и конкретным удаленным интерфейсом с указанием его ip-адреса.</p> <p>Session Interface Index - системный номер локального интерфейса;</p> <p>Lower Layer - инкапсулирующий протокол;</p>

Команда	Описание
	<p>Session State - состояние сессии;</p> <p>Session Index - локальный id сессии;</p> <p>Session Type - тип сессии (single/multi);</p> <p>Local Discriminator - локальный id сессии;</p> <p>Local Address - адрес интерфейса локального маршрутизатора, на котором установлена сессия;</p> <p>Local Port - локальный UDP-порт;</p> <p>Remote Discriminator - id сессии на удаленном устройстве;</p> <p>Remote Address - адрес интерфейса удаленного маршрутизатора, с которым установлена сессия;</p> <p>Remote Port - удаленный UDP-порт;</p> <p>Min Tx - локальный интервал отправки bfd-control сообщений;</p> <p>Min Rx - локальный ожидаемый интервал приёма bfd-control сообщений;</p> <p>Multiplier - количество пропущенных сообщений, после которого сессия считается порванной;</p> <p>UPTIME - up-time сессии</p>

20.2 Пример настройки single-hop BFD-OSPF



Рисунок 40

Конфигурация EcoRouter1:

Настройка интерфейсов и портов:

```

ecorouter(config)#port te0
ecorouter(config-port)#service-instance si0
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config)#interface loopback.0
ecorouter(config-lo)#ip address 1.1.1.1/32

```

```
ecorouter(config)#interface te0
ecorouter(config-if)#ip address 10.1.1.1/24
ecorouter(config-if)#connect port te0 service-instance si0
```

Настройка OSPF и включение BFD:

```
ecorouter(config)#router ospf 100
ecorouter(config-router)#ospf router-id 1.1.1.1
ecorouter(config-router)#network 1.1.1.1/32 area 0.0.0.1
ecorouter(config-router)#network 10.1.1.0/24 area 0.0.0.1
ecorouter(config-router)#bfd all-interfaces
```

Включение echo-функции:

```
ecorouter(config)#bfd echo
```

Конфигурация EcoRouter2:

Настройка интерфейсов и портов:

```
ecorouter(config)#port te0
ecorouter(config-port)#service-instance si0
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config)#interface loopback.0
ecorouter(config-lo)#ip address 2.2.2.2/32
ecorouter(config)#interface te0
ecorouter(config-if)#ip address 10.1.1.2/24
ecorouter(config-if)#connect port te0 service-instance si0
```

Настройка OSPF и включение BFD:

```
ecorouter(config)#router ospf 100
ecorouter(config-router)#ospf router-id 2.2.2.2
ecorouter(config-router)#network 2.2.2.2/32 area 0.0.0.1
ecorouter(config-router)#network 10.1.1.0/24 area 0.0.0.1
ecorouter(config-router)#bfd all-interfaces
```

Включение echo-функции:

```
ecorouter(config)#bfd echo
```

21 BRAS

Одним из центральных элементов сети интернет-провайдера является BRAS (Broadband Remote Access Server) - сервер широкополосного удаленного доступа. Под аббревиатурой BRAS понимают устройство, которое отвечает за маршрутизацию внутри сети, предоставление доступа подписчикам/абонентам к различным сервисам (Интернет, IP-телефония, IP-телевидение) посредством одного или нескольких физических подключений. С помощью BRAS можно создать и поддерживать необходимые правила качества обслуживания (QoS) для различного типа трафика при динамично изменяющейся загрузке и параметрах каналов связи.

Основными задачами сервера широкополосного удаленного доступа являются следующие:

- назначение и применение сетевых настроек на клиентском оборудовании;
- аутентификация, авторизация и выделение индивидуальных атрибутов для абонентов;
- учет, фильтрация и тарификация трафика;
- обеспечение требуемого качества предоставляемых сервисов;
- гибкое подключение новых сервисов, услуг.

Некоторые из этих задач решаются при взаимодействии BRAS с другими устройствами в сети. Например, задачи аутентификации и авторизации могут решаться с помощью обращения к внешним Tacacs- или Radius-серверам. Устройства EcoRouter позволяют при запуске виртуальных сервисов на маршрутизаторе использовать как удаленные, так и локальные серверы AAA (запущенные непосредственно на самом маршрутизаторе).

Для предоставления интернет-услуг используется несколько протоколов. До недавнего времени наиболее распространенным был протокол PPPoE (Point-to-point Protocol over Ethernet). Технология доставки и предоставления IP-настроек абонентам (IPoE - Internet Protocol over Ethernet) в связке с применением DHCP опции 82 используется все чаще, так как требует минимум конфигурации конечного оборудования. Технология Q-in-Q, которая является расширением стандарта IEEE 802.1Q считается наиболее безопасной. При ее использовании изначально каждое конечное устройство находится в выделенном VLAN, чем гарантируется изоляция подписчиков друг от друга.

EcoRouterOS поддерживает все вышеперечисленные протоколы и технологии, а концепция EVC (Ethernet Virtual Connection) позволяет гибко работать с тегированным трафиком вне зависимости от выбранного варианта подключения пользователей, тем самым гарантируя высокую степень изоляции для IPoE- и PPPoE-сессий.

21.1 Настройки IPoE

EcoRouterOS поддерживает работу технологии IPoE как для статически сконфигурированных абонентов, так и для динамических сессий, которые были созданы посредством протокола DHCP. Для запуска сервисов IPoE необходимо создать интерфейс с именем **bmi.<NUM>**, где <NUM> - номер интерфейса в диапазоне от 0 до 9999999999.

Пример:

```

ecorouter(config)#interface bmi.1
ecorouter(config-if-bmi)#
IPoE/PPPoE interface configuration commands:
add-mirror-session           Add mirror session
bfd                         Bidirectional Forwarding Detection (BFD)
connect                      Connect interface
description                  Interface specific description
dhcp-profile                 Enable DHCP profile
echo                         echo mode
exit                         Exit from the current mode to the previous
mode
flow-export-profile          Enable options
help                         Description of the interactive help system
ip                            IP Information
isis                          Intermediate System - Intermediate System
(IS-IS)
ldp                           Label Distribution Protocol parameters
mpls                         Configure MPLS specific attributes
multicast                    Set multicast flag to interface
no                            Negate a command or set its defaults
rate-limit                   Configure rate-limit
session-trigger              Set IPoE session trigger
set                           Enable options
show                         Show running system information
shutdown                     Shutdown interface
snmp                         snmp
subscriber-map               Specify subscriber-map for this interface
virtual-router-forwarding    Associate this interface with specific
Virtual
                                Router

```

Этот интерфейс ничем не отличается от обычного L3-интерфейса и требует подключения к реальному физическому L2-порту по EVC через экземпляр сервиса (service-instance). Сетевому администратору не требуется специально настраивать поведение по команде **rewrite** и ее вариантах, так как при приеме пакетов с L2 на L3 EcoRouterOS сбросит все теги автоматически.

Далее необходимо статически выделить IP-адрес и VLAN теги для конкретного абонента. Конфигурируется подобная привязка с помощью префиксных списков (prefix-list) и пользовательских карт (subscriber-map) - более подробно см. соответствующие разделы настоящего руководства.

IP-адресу клиента необходимо сопоставить определенный префиксный список.

Для клиента с IP адресом 10.0.0.1 префиксный список будет выглядеть так:

```
ip prefix-list CLIENT_A permit 10.0.0.1/32
```

Также предусмотрена возможность для задания диапазона пользовательских адресов, которым будет присвоен один и тот же сервис.

```
ip prefix-list CLIENTS permit 192.168.1.0/24
```

Привязка абонента к карте абонента осуществляется с помощью команды **subscriber-map <NAME> <NUM>**, где <NAME> - имя, строка до 15 символов, рекомендуемый формат имени – все буквы заглавные, а <NUM> - номер в диапазоне от 1 до 65535. Порядковый номер

карты определяет порядок обработки. В первую очередь будет обработана карта с номером 1, последней будет карта с именем DEFAULT, созданная по умолчанию.

Пример:

```
ecorouter(config)#subscriber-map A 1
ecorouter(config-sub-map)#
Subscriber map configuration commands:
description Add entry description
exit      Exit from the current mode to the previous mode
help      Description of the interactive help system
match     Match subscribers
no        Negate a command or set its defaults
set       Set policies on matched subscribers
show      Show running system information
```

Для настройки пользовательской карты используются команды **match** и **set**. Логика работы пользовательских карт подобна логике карт маршрутов (route-map): при выполнении условия в правиле, заданного при помощи команды **match** (ip-адрес абонента входит в указанный диапазон), попытаться поднять сессию с настройками, указанными при помощи команд **set**. Если же ip-адрес абонента не подходит под **match**, то сессия для этого абонента заводиться не будет.

Правила бывают статические (**static**) и динамические (**dynamic**).

Все IP-адреса в статическом правиле определяются только маской /32. Также задаются **svlan** и **cvlan** при необходимости. Тогда запись сразу попадет в таблицу IPoE и будет доступна, пока команда присутствует в конфигурации маршрутизатора. Просмотр таблицы IPoE производится при помощи команды **show subscribers <NAME>**, где <NAME> - имя ВМИ-интерфейса.

В динамическом правиле IP-адреса определяются маской строго меньше /32. Записи для таких адресов создаются по первому пакету от абонента. Теги **vlan** изучаются динамически. Для данных записей применимы таймауты и сброс сессии. Подробнее см. в соответствующем разделе данного руководства, посвященном картам абонентов (subscriber-map).

Пример:

```
ecorouter(config)#subscriber-map A 1
ecorouter(config-sub-map)#match ?
  dynamic  Dynamically allocated entries
  static   Statically allocated entries
ecorouter(config-sub-map)#match static
  prefix-list Match using prefix-list
ecorouter(config-sub-map)#match static prefix-list CLIENT_A ?
  cvlan    Specify customer vlan
  svlan    Specify service vlan
  untagged Specify untagged customers
ecorouter(config)#subscriber-map A 2
ecorouter(config-sub-map)#match dynamic prefix-list CLIENTS
```

Для создания статической абонентской сессии в EcoRouterOS необходимо указать определенный номер префиксного списка, 8021.Q теги для клиентской и сервисной виртуальной локальной сети. Если используется диапазон абонентских IP-адресов, то

абонентская сессия создается динамически. При этом запоминаются теги **vlan**, с которыми каждый абонент подключается. Другими словами, сессией из определенной виртуальной локальной сети (VLAN абонента) с назначенным IP-адресом источника (адрес абонента) можно управлять (включать функции AAA, применять тарифы, ограничивать трафик и т.д.).

Для настройки ограничений для абонентской сессии нужно выполнить следующие шаги:

1. создать сервис абонента (subscriber-service) при помощи команды конфигурационного режима **subscriber-service <NAME>**, где **<NAME>** - произвольная строка до 15 символов, рекомендуемый формат имени – все буквы заглавные;
2. связать созданный subscriber-service с картой абонента (subscriber-map) при помощи команды **set subscriber-service** в контекстном режиме редактирования карты абонента.

```
ecorouter(config-sub-map)#set ?
  aaa-profile      Set AAA profile
  idle-timeout    Set idle timeout
  subscriber-service Set service
  session-timeout Set session timeout
  update-interval Set update-interval
```

Создание сервиса абонента включает в себя задание максимальной полосы пропускания трафика. Для каждого направления (от абонента/к абоненту) необходима отдельная настройка. Пример настройки см. ниже.

```
ecorouter(config)#subscriber-service TEST
ecorouter(config-sub-service)#
Subscriber service configuration commands:
  description Subscriber service description
  exit          Exit from the current mode to the previous mode
  help          Description of the interactive help system
  no            Negate a command or set its defaults
  set           Set policies on matched subscribers
  show          Show running system information
?corouter(config-sub-service)#set
  policy Set policy
?corouter(config-sub-service)#set policy
  SUBSCRIBER_POLICY_NAME Subscriber policy name
```

Ниже приведен пример конфигурации маршрутизатора с ограничением трафика до 10 Mb для абонентского подключения из VLAN 2 с IP 192.168.0.1/24.

```
ecorouter(config)#interface bmi.1
ecorouter(config-if-bmi)#ip address 192.168.0.100/24
ecorouter(config-if-bmi)#exit
ecorouter(config)#ip prefix-list CLIENT_A permit 192.168.0.1/32
ecorouter(config)#filter-map policy ipv4 ALL
ecorouter(config-filter-map-policy-ipv4)#match any any any
ecorouter(config-filter-map-policy-ipv4)#set accept
ecorouter(config-filter-map-policy-ipv4)#ex
ecorouter(config)#subscriber-policy for_client
ecorouter(config-sub-policy)#bandwidth in kbps 10000
ecorouter(config-sub-policy)#set filter-map in ALL
ecorouter(config-sub-policy)#ex
ecorouter(config)#subscriber-service ALL
```

```

ecorouter(config-sub-service)#set policy for_client
ecorouter(config-sub-service)#ex
ecorouter(config-sub-map)#match static prefix-list CLIENT_A cvlan 2
ecorouter(config-sub-map)#set service ALL
ecorouter(config-sub-map)#exit
ecorouter(config)#interface bmi.1
ecorouter(config-if-bmi)#subscriber-map A
ecorouter(config-if-bmi)#exit
ecorouter(config)#port tel
ecorouter(config-port)#service-instance test
ecorouter(config-sub-service-instance)#encapsulation dot1q 2 exact
ecorouter(config-sub-service-instance)#connect ip interface bmi.1
ecorouter(config-sub-service-instance)#exit

```

Дополнительные настройки для абонентских сессий

В контекстном режиме редактирования карты абонента помимо настройки **set subscriber-service** доступны настройки других параметров:

```

ecorouter(config-sub-map)#set ?
  aaa-profile      Set AAA profile
  idle-timeout     Set idle timeout
  subscriber-service Set service
  session-timeout   Set session timeout
  update-interval    Set update-interval

```

Команды **set session-timeout** и **set idle-timeout** позволяют установить ограничения по времени жизни сессии. **Session-timeout** - это жесткое ограничение времени жизни сессии, по истечению которого сессия удаляется в любом случае. Значение по умолчанию для этого параметра - 1440 минут. **Idle-timeout** - это ограничение времени жизни сессии в зависимости от приходящего трафика от абонента. По истечению времени, заданного параметром **idle-timeout**, сессия удаляется, только если от абонента не приходил трафик на протяжении времени **idle-timeout**. Значение по умолчанию для этого параметра - 30 минут. Значение 0 минут для обоих параметров означает бесконечно большое значение.

Команда **set update-interval** позволяет настроить частоту отправки **Interim-Update accounting** сообщений. По умолчанию update interval не настроен, т.е. **Interim-Update** сообщения не отсылаются. Диапазоны настройки в минутах для всех параметров приведены ниже:

```

ecorouter(config-sub-map)#set idle-timeout
<0-1440> Timeout (min)
ecorouter(config-sub-map)#set session-timeout
<0-527040> Timeout (min)
ecorouter(config-sub-map)#set update-interval
<5-1440> Interval (min)

```

Команда **set aaa-profile** позволяет указать RADIUS-сервер, посредством которого следует аутентифицировать абонентов.

21.1.1 Динамический IPoE

Для аутентификации абонентов в EcoRouterOS может быть использован внешний RADIUS-сервер. Все абоненты, попадающие на IPoE-интерфейс и не имеющие статической записи, будут аутентифицированы на RADIUS-сервере.

Чтобы задать конфигурацию сервера, необходимо настроить абонентский AAA-профиль (aaa-profile).

Настройка производится в режиме конфигурирования.

```
aaa-profile <NAME> radius-server <RADIUS-IP> secret <STRING> [auth-port <AUTHPORT> | acct-port <ACCTPORT>]
```

Параметры команды

<NAME> - название aaa-profile;

<RADIUS-IP> - ip адрес RADIUS сервера (на данный момент RADIUS сервер доступен только через mgmt порт);

<STRING> - пароль для установления сессии с RADIUS сервером;

<AUTHPORT> - порт для аутентификации, по умолчанию - 1812;

<ACCTPORT> - порт для аккаунтинга, по умолчанию - 1813.

Пример (для RADIUS-сервера с IP-адресом **1.1.1.1** и паролем **superpassword**).

```
ecorouter(config)#aaa-profile radius  
ecorouter(config-aaa-profile)#radius-server 1.1.1.1 secret superpassword
```

При аутентификации абонента через RADIUS-сервер EcoRouter отправляет RADIUS access request со следующей информацией:

- **User-Name:** <MAC-адрес абонента>;
- **Framed-IP-Address:** <IP-адрес абонента>;
- **Calling-Station-Id:** <MAC-адрес абонента>;
- **NAS-Identifier:** <Имя маршрутизатора, указанное в **hostname**>;
- **NAS-Port-Id:** <Имя порта маршрутизатора:имя интерфейса:c-vlan:s-vlan> - порт и интерфейс указываются те, на которые пришёл пакет-триггер (пакет, ставший триггером для отправки запроса на RADIUS-сервер). Метки **vlan** указываются те, которые присутствовали в заголовке пакета-триггера;
- **NAS-Port-Type:** <Тип порта, на который пришёл пакет-триггер>;
- **CIRCUIT_ID:** <DHCP option 82 circuit-id> - субатрибут атрибута Vendor-Specific(26). Для отображения этих параметров на RADIUS-сервере следует произвести соответствующие настройки в словаре сервера, которые приведены ниже;
- **REMOTE_ID:** <DHCP option 82 remote-id> - субатрибут атрибута Vendor-Specific(26). Для отображения этих параметров на RADIUS-сервере следует произвести соответствующие настройки в словаре сервера, которые приведены ниже;
- **NAS-IP-Address:** <IP-адрес, идентифицирующий маршрутизатор> - если на устройстве создан интерфейс **loopback.0** и на него назначен IP-адрес, то в этот атрибут будет записан адрес с интерфейса loopback.0. Если интерфейс **loopback.0** отсутствует в конфигурации маршрутизатора, то в этот атрибут будет записан IP-адрес с интерфейса, с которого был отправлен RADIUS access request;
- **Framed-Protocol:** <Тип инкапсулирующего протокола> - варианты для заполнения атрибута в текущей реализации: 1. PPP;
- **NAS-Port:** <c-vlan> - внутренняя метка **vlan** из заголовка пакета-триггера.

При аутентификации абонента через RADIUS-сервер EcoRouter обрабатывает следующие атрибуты в RADIUS access reply:

- **Idle-Timeout:** <idle-timeout сессии>;
- **Session-Timeout:** <session-timeout сессии>;
- **Acct-Interim-Interval:** <update-interval сессии>;
- **Class:** <стандартный атрибут, тип 25>;

SERVICE_NAME: <имя сервиса, который будет применен на сессию> - сервис будет применен на сессию при условии, что он создан на маршрутизаторе при помощи команды **subscriber-service <service_name>**.

После аутентификации абонента, если для него была заведена сессия, маршрутизатор отправляет accounting request сообщения со следующей информацией:

- **Acct-Status-Type:** <Тип accounting request сообщения> - в текущей реализации может принимать значения: **start, stop** и **interim-update**;
- **Acct-Session-Id:** <Идентификатор абонентской сессии> - идентификатор генерируется маршрутизатором на основе следующих ключей: ip-адрес абонента и время поднятия сессии;
- **Event-Timestamp:** <Время отправки сообщения>;
- **Acct-Authentic:** <Способ аутентификации абонента> - в текущей реализации может принимать значения: **radius** и **local**;
- **Class:** <стандартный атрибут, тип 25>;
- **Acct-Session-Time:** <Текущее время жизни сессии>;
- **Acct-Input-Packets:** <Количество пакетов, отправленных абонентом в течение сессии>;
- **Acct-Input-Octets:** <Количество байт, отправленных абонентом в течение сессии>;
- **Acct-Input-Gigawords:** <Количество переполнений счетчика **Acct-Input-Octets**>;
- **Acct-Output-Packets:** <Количество пакетов, отправленных абоненту в течение сессии>;
- **Acct-Output-Octets:** <Количество байт, отправленных абоненту в течение сессии>;
- **Acct-Output-Gigawords:** <Количество переполнений счетчика **Acct-Output-Octets**>;
- **Acct-Delay-Time:** <Время, которое было затрачено на отправку accounting request сообщения>;
- **Acct-Terminate-Cause:** <Причина, по которой сессия была сброшена маршрутизатором> - в текущей реализации может принимать значения: **Idle Timeout** (сброс сессии по истечению idle-timeout), **Session Timeout** (сброс сессии по истечению session-timeout), **Admin Reset** (сброс сессии при помощи команды **clear subscribers**), **Port Error** (сброс сессии вследствие удаления или выключения соответствующего bmi интерфейса), **Service Unavailable** (сброс сессии вследствие запроса RADIUS-сервером не настроенного на маршрутизаторе сервиса).

Пример RADIUS access request:

```
00:01:04 hub.rdp.ru-freeradius-1: (0) Received Access-Request Id 136
from 192.168.255.1:57890 to 192.168.255.2:1812 length 116
```

```

00:01:04 hub.rdp.ru-freeradius-1: (0) Service-Type = Login-User
00:01:04 hub.rdp.ru-freeradius-1: (0) User-Name = "0050.7966.6800"
00:01:04 hub.rdp.ru-freeradius-1: (0) Framed-IP-Address = 20.20.20.2
00:01:04 hub.rdp.ru-freeradius-1: (0) NAS-Identifier = "ecorouter"
00:01:04 hub.rdp.ru-freeradius-1: (0) NAS-Port-Id = "te0:bmi.1:10:4"
00:01:04 hub.rdp.ru-freeradius-1: (0) NAS-Port-Type = Ethernet
00:01:04 hub.rdp.ru-freeradius-1: (0) CIRCUIT_ID = "fffff"
00:01:04 hub.rdp.ru-freeradius-1: (0) REMOTE_ID = "fffff"
00:01:04 hub.rdp.ru-freeradius-1: (0) NAS-IP-Address = 9.8.7.1
00:01:04 hub.rdp.ru-freeradius-1: (0) Framed-Protocol = PPP
00:01:04 hub.rdp.ru-freeradius-1: (0) NAS-Port = 10

```

Пример RADIUS accounting request:

```

00:02:05 hub.rdp.ru-freeradius-1: (1) Service-Type = Login-User
00:02:05 hub.rdp.ru-freeradius-1: (1) User-Name = "0050.7966.6802"
00:02:05 hub.rdp.ru-freeradius-1: (1) Framed-IP-Address = 20.20.20.3
00:02:05 hub.rdp.ru-freeradius-1: (1) NAS-Identifier = "ecorouter"
00:02:05 hub.rdp.ru-freeradius-1: (1) NAS-Port-Id = "tel:bmi.0:4:0"
00:02:05 hub.rdp.ru-freeradius-1: (1) NAS-Port-Type = Ethernet
00:02:05 hub.rdp.ru-freeradius-1: (1) NAS-IP-Address = 20.20.20.1
00:02:05 hub.rdp.ru-freeradius-1: (1) Framed-Protocol = PPP
00:02:05 hub.rdp.ru-freeradius-1: (1) Event-Timestamp = "Mar 13 2018
08:22:08 UTC"
00:02:05 hub.rdp.ru-freeradius-1: (1) Acct-Status-Type = Stop
00:02:05 hub.rdp.ru-freeradius-1: (1) Acct-Session-Id =
"5aa78a3003141414"
00:02:05 hub.rdp.ru-freeradius-1: (1) Acct-Authentic = RADIUS
00:02:05 hub.rdp.ru-freeradius-1: (1) Acct-Session-Time = 0
00:02:05 hub.rdp.ru-freeradius-1: (1) Acct-Input-Packets = 0
00:02:05 hub.rdp.ru-freeradius-1: (1) Acct-Output-Packets = 0
00:02:05 hub.rdp.ru-freeradius-1: (1) Acct-Input-Octets = 0
00:02:05 hub.rdp.ru-freeradius-1: (1) Acct-Output-Octets = 0
00:02:05 hub.rdp.ru-freeradius-1: (1) Acct-Input-Gigawords = 0
00:02:05 hub.rdp.ru-freeradius-1: (1) Acct-Output-Gigawords = 0
00:02:05 hub.rdp.ru-freeradius-1: (1) Acct-Terminate-Cause = Idle-
Timeout
00:02:05 hub.rdp.ru-freeradius-1: (1) NAS-Port = 4
00:02:05 hub.rdp.ru-freeradius-1: (1) Acct-Delay-Time = 0

```

Пример настройки словаря RADIUS-сервера для обработки атрибута Vendor-Specific(26):

```

VENDOR      RDP      45555
BEGIN-VENDOR RDP
  ATTRIBUTE  REMOTE_ID    96  string
  ATTRIBUTE  CIRCUIT_ID   97  string
  ATTRIBUTE  SERVICE_NAME 250  string
END-VENDOR   RDP

```

21.1.2 Параметры IPoE

Инициализация IPoE-сессии в зависимости от установленного значения параметра **session-trigger** в настройках BMI-интерфейса происходит либо по первому пакету DHCP Discovery от абонента (настройка по умолчанию), либо по первому IP-пакету от абонента.

В таблице клиентов сессия, устанавливаемая по пакету DHCP Discovery, имеет тип DHCP, а в поле статус отображается последний DHCP-пакет по данной сессии. Если сессия настроена статически через CLI EcoRouter, то после прохождения пакета DHCP Ack от DHCP сервера, маршрутизатор формирует сессию с IP-адресом и типом **static**. Если данного абонента необходимо аутентифицировать через RADIUS, то сессия помечается как **IPoE**, статус переходит в состояние **in progress** до получения ответа от RADIUS-сервера. После получения ответа статус переходит в состояние **accepted** либо **rejected**.

Для просмотра абонентов, подключенных к BMI-интерфейсу, следует в пользовательском или административном режиме выполнить команду **show subscribers <NAME>**, где <NAME> - имя BMI-интерфейса.

ecorouter#show subscribers bmi.1						
IP Address	MAC Address	Port	S-tag	C-tag	Status	Type
172.16.2.10	a036.9fc7.4f10	ge14	-----	10	accepted	IPoE

IP Address - IP-адрес абонента;

MAC Address - mac адрес абонента;

Port - порт, через который подключен абонент;

S-tag, C-tag - VLAN-теги абонентского трафика;

Status - статус данного абонента.

Статус может быть одним из следующих:

accepted - абонент успешно аутентифицировался на RADIUS-сервере;

rejected - абонент заблокирован;

in progress - отправлен запрос на RADIUS-сервер.

Type - тип подключения:

static - абонент задан через CLI EcoRouter в subscriber-map;

IPoE - IPoE сессия;

PPPoE - PPPoE сессия;

dhcp - абонент находится на стадии получения IP-адреса с DHCP-сервера.

Для типа DHCP возможны следующие статусы:

discovery - получен discovery-пакет от абонента;

offer - offer-пакет отправлен абоненту;

request - абонент отправил request-пакет.

После получения сообщения **ack** сессия моментально переходит в состояние **IPoE**, поэтому этот статус не отображается.

В EcoRouterOS есть возможность вручную сбросить абонентскую сессию или счетчики пакетов и байтов по сессии. Для сброса сессии в административном режиме необходимо выполнить команду:

```
clear subscribers IFNAME ip|mac|all
```

Для сброса счетчиков по сессии в административном режиме необходимо выполнить команду:

```
clear counters subscribers IFNAME ip|mac|all
```

Абонентскую сессию или счетчики по ней можно сбросить по IP-адресу или по MAC-адресу - в случае, когда у абонента еще нет IP-адреса. Также можно сбросить все сессии (или счетчики по всем сессиям) на определенном BMI-интерфейсе. Сброс счетчиков по сессии инициирует отправку **Interim-Update** accounting сообщения с обновленными атрибутами **Acct-Input-Octets**, **Acct-Output-Octets**, **Acct-Input-Packets**, **Acct-Output-Packets**, **Acct-Input-Gigawords** и **Acct-Output-Gigawords**.

21.1.3 Логирование IPoE

Для отслеживания установления пользовательской сессии IPoE служит команда режима администрирования **debug subscriber**.

Параметры команды описаны в таблице ниже.

Таблица 91

Параметр	Описание
ip <IP ADDRESS>	IP-адрес абонента
mac <MAC ADDRESS>	MAC-адрес абонета
svlan <NUM>	сервисный VLAN, в случае модели Q-in-Q
cvlan <NUM>	клиентский VLAN
as <NAME>	префикс для сообщений отладки данного пользователя. Данный префикс добавляется в каждое сообщение

Если включена отладка по MAC-адресу, svlan или cvlan, то в логах можно наблюдать DHCP и RADIUS-логи.

Если включена отладка по IP-адресу - в логах будут только RADIUS-сообщения.

Пример отладки по MAC-адресу:

```
ecorouter#debug subscriber mac 0050.7966.6801 as PETROV
```

Логи:

```
[data-plane] [PETROV] DHCP-DISCOVER message received from client
00:50:79:66:68:01
[data-plane] [PETROV] dhcp, delete client: 00:50:79:66:68:01
[data-plane] [PETROV] DHCP-DISCOVER message received from client
00:50:79:66:68:01
[data-plane] [PETROV] dhcp, delete client: 00:50:79:66:68:01
[data-plane] [PETROV] DHCP-OFFER message received for client
00:50:79:66:68:01
[data-plane] [PETROV] DHCP-REQUEST message received from client
00:50:79:66:68:01
[data-plane] [PETROV] DHCP-ACKNOWLEDGE message received for client
00:50:79:66:68:01
[data-plane] [PETROV] Client IP: 10.1.1.3 sent request to radius client
[radius-client] [PETROV] radius_module.cpp:27(AuthRequest) Request
created. State: NEW. Client ip: 10.1.1.3
```

```
[radius-client] [PETROV] radius_module.cpp:125(sendRequests)
authenticating: client ip 10.1.1.3
[radius-client] [PETROV] radius_module.cpp:35(setState) State change:
NEW -> PENDING. Client ip: 10.1.1.3
[radius-client] [PETROV] radius_module.cpp:35(setState) State change:
PENDING -> READY. Client ip: 10.1.1.3
[radius-client] [PETROV] radius_module.cpp:35(setState) State change:
READY -> RECEIVED_OK. Client ip: 10.1.1.3
[radius-client] [PETROV] radius_module.cpp:653(parsePair) rc_auth
10.1.1.3 success
[radius-client] [PETROV] radius_module.cpp:342(finishAuth)
Authentication succeeded, client ip: 10.1.1.3
[data-plane] [PETROV] Update ipoe client session "SUBSCRIBER DYNAMIC
AUTH_COMPLETED ACTIVE " on ip : 10.1.1.3 on iface 1, (socket 0)
```

Пример отладки по IP-адресу:

```
ecorouter#debug subscriber ip 10.1.1.4 as IVANOV
```

Логи:

```
[note] [data-plane] [IVANOV] Client IP: 10.1.1.4 sent request to radius
client in first time
[debug] [radius-client] [IVANOV] radius_module.cpp:27(AuthRequest)
Request created. State: NEW. Client ip: 10.1.1.4
[info] [radius-client] [IVANOV] radius_module.cpp:125(sendRequests)
authenticating: client ip 10.1.1.4
[debug] [radius-client] [IVANOV] radius_module.cpp:35(setState) State
change: NEW -> PENDING. Client ip: 10.1.1.4
[debug] [radius-client] [IVANOV] radius_module.cpp:35(setState) State
change: PENDING -> READY. Client ip: 10.1.1.4
[debug] [radius-client] [IVANOV] radius_module.cpp:35(setState) State
change: READY -> RECEIVED_REJECT. Client ip: 10.1.1.4
[info] [radius-client] [IVANOV] radius_module.cpp:684(parsePair) rc_auth
10.1.1.4 reject
[info] [radius-client] [IVANOV] radius_module.cpp:342(finishAuth)
Authentication succeeded, client ip: 10.1.1.4
[data-plane] [IVANOV] Update ipoe client session "SUBSCRIBER
DYNAMIC AUTH_COMPLETED NOT_ACTIVE " on ip : 10.1.1.4 on iface 1, (socket
0)
```

Пример отладки по клиентскому VLAN:

```
ecorouter#debug subscriber cvlan 10 as VLAN10
```

Логи:

```
[data-plane] [VLAN10] DHCP-DISCOVER message received from client
00:50:79:66:68:01
[data-plane] [VLAN10] dhcp, delete client: 00:50:79:66:68:01
[data-plane] [VLAN10] DHCP-OFFER message received for client
00:50:79:66:68:01
[data-plane] [VLAN10] DHCP-REQUEST message received from client
00:50:79:66:68:01
[data-plane] [VLAN10] DHCP-ACKNOWLEDGE message received for client
00:50:79:66:68:01
[data-plane] [VLAN10] DHCP-DISCOVER message received from client
00:50:79:66:68:02
[data-plane] [VLAN10] DHCP-OFFER message received for client
00:50:79:66:68:02
```

```
[data-plane] [VLAN10] DHCP-REQUEST message received from client
00:50:79:66:68:02
[data-plane] [VLAN10] DHCP-ACKNOWLEDGE message received for client
00:50:79:66:68:02
[data-plane] [VLAN10] Client IP: 10.1.1.4 sent request to radius client
in first time
[radius-client] [VLAN10] radius_module.cpp:27(AuthRequest) Request
created. State: NEW. Client ip: 10.1.1.4
[radius-client] [VLAN10] radius_module.cpp:125(sendRequests)
authenticating: client ip 10.1.1.4
[radius-client] [VLAN10] radius_module.cpp:35(setState) State change:
NEW -> PENDING. Client ip: 10.1.1.4
[radius-client] [VLAN10] radius_module.cpp:35(setState) State change:
PENDING -> RETRY. Client ip: 10.1.1.4
[radius-client] [VLAN10] radius_module.cpp:166(sendRequests) No servers
left to try. rc_auth_async returned code -1, client ip: 10.1.1.4
[radius-client] [VLAN10] radius_module.cpp:35(setState) State change:
RETRY -> SEND_FAILED. Client ip: 10.1.1.4
[radius-client] [VLAN10] radius_module.cpp:338(finishAuth)
Authentication failed, client ip: 10.1.1.4
```

Кроме того, удобно отслеживать установление сессии при помощи команды режима администрирования **terminal monitor <LINE>**. Где **LINE** - слово, по которому будет произведена выборка из логов. Данная команда отображает только интересующие пользователя сообщения.

21.1.4 Команды просмотра карт абонентов и сервисов абонентов

Подробную информацию по определенной карте абонента можно узнать, выполнив команду **show subscriber-map <SMNAME>**, где <SMNAME> - имя карты абонента.

Пример:

```
ecorouter#sh subscriber-map clients
Subscriber-map "clients" is applied for:
  Interface      IP-Address
    bmi.1        10.1.1.1/24
    bmi.2        unassigned
Sequence 10
  match static prefix-list pc2
  match static prefix-list pc2222
  set service 2mbps
Sequence 20
  description: "test"
  match dynamic prefix-list pc2
  set service 5mbps
Implicit default rule: "DROP"
```

Если карта применена на BMI-интерфейсе, то информация по интерфейсу будет присутствовать в выводе команды с указанием сконфигурированного IP-адреса.

Ниже приведен вывод при отсутствии примененной карты абонента на интерфейсе (subscriber-map не была применена на BMI-интерфейсе).

```
Subscriber-map "clients" is applied for:
  Interface      IP-Address
    <empty>      <empty>
```

Краткая информация по всем картам абонентов доступна, если при вызове команды **show subscriber-map** имя карты опущено.

Пример:

```
ecorouter#sh subscriber-map
Subscriber-map      Interface          IP-Address
-----
clients            bmi.1           10.1.1.1/24
                  bmi.2           2.2.2.2/28
                  bmi.3           unassigned
test              <empty>        <empty>
```

Просмотр счетчиков по всем абонентам на BMI-интерфейсе производится при помощи команды: **show counters subscribers <INAME> all**, где <INAME> - имя интерфейса.

Пример:

```
ecorouter#sh counters subscribers bmi.1 all
IP Address      | Wan Bytes       | Lan Bytes       | Wan Packets   |
Lan Packets    |                   |
-----+-----+-----+-----+
-----+-----+
20.20.20.2     |         96614 |         3164 |         67
|             4  |
20.20.20.3     |         1551788 |         3122 |        1078
|             3  |
```

Для просмотра счетчиков по конкретному абоненту при вызове команды следует указать адрес абонента: **show counters subscribers <INAME> <IP>**, где <INAME> - имя интерфейса, <IP> - адрес абонента.

Пример:

```
ecorouter#sh counters subscribers bmi.1 20.20.20.2
Policy          | Wan Bytes       | Lan Bytes       | Wan Packets   |
Lan Packets    |                   |
-----+-----+-----+-----+
-----+-----+
test            |         196 |         0 |         2 |         0 |
(default)       |         96614 |         3164 |         67
|             4  |
TOTAL:          |         96614 |         3164 |         67 |         4
```

Просмотр информации по всем абонентам производится при помощи команды: **show subscribers <INAME>**, где <INAME> - имя интерфейса.

Пример:

```
ecorouter#sh subscribers bmi.1
Total subscribers: 4
accepted: 4, rejected: 0, auth. in progress: 0, getting IP by DHCP: 0
Codes: L - local, R - remote AAA, U - unknown, N - not specified
IP Address      MAC Address     Port      S-tag  C-tag  Status     Type
-----
20.20.20.2      3e3a.6af3.6edd  tel      -----  ----- accepted(L)  IPOE
20.20.20.3      7e6e.5221.bf2a   tel      -----  ----- accepted(L)  IPOE
```

```

20.20.20.5    0000.0000.0000  tel      ----- -----
- accepted(L) static
20.20.20.6    8e5e.5223.e212  tel      ----- -----
- accepted(L) PPPoE

```

Просмотр краткой информации по всем абонентам производится при помощи команды: **show subscribers <INAME> brief**, где <INAME> - имя интерфейса.

Пример:

```

ecorouter#sh subscribers bmi.1 brief
Total subscribers: 2
  accepted: 2, rejected: 0, auth. in progress: 0, getting IP by DHCP: 0
  Codes: L - local, R - remote AAA, U - unknown, N - not specified
IP Address      MAC Address      Status      Type
-----
20.20.20.2      3e3a.6af3.6edd  accepted(L)  IPoE
20.20.20.3      7e6e.5221.bf2a   accepted(L)  IPoE

```

Просмотр информации только по статическим абонентам производится при помощи команды: **show subscribers <INAME> static**, где <INAME> - имя интерфейса.

Пример:

```

ecorouter#sh subscribers bmi.1 static
Total subscribers: 1
  accepted: 1, rejected: 0, auth. in progress: 0, getting IP by DHCP: 0
  Codes: L - local, R - remote AAA, U - unknown, N - not specified
IP Address      MAC Address      Port      S-tag  C-tag  Status      Type
-----
20.20.20.5      0000.0000.0000  tel      ----- -----
- accepted(L) static

```

Просмотр информации только по PPPoE абонентам производится при помощи команды: **show subscribers <INAME> pppoe**, где <INAME> - имя интерфейса.

Пример:

```

ecorouter#sh subscribers bmi.1 pppoe
Total subscribers: 1
  accepted: 1, rejected: 0, auth. in progress: 0, getting IP by DHCP: 0
  Codes: L - local, R - remote AAA, U - unknown, N - not specified
IP Address      MAC Address      Port      S-tag  C-tag  Status      Type
-----
20.20.20.6      8e5e.5223.e212  tel      ----- -----
- accepted(L) PPPoE

```

Просмотр информации только по IPoE абонентам производится при помощи команды: **show subscribers <INAME> ipoe**, где <INAME> - имя интерфейса.

Пример:

```

ecorouter#sh subscribers bmi.1 ipoe
Total subscribers: 2
  accepted: 2, rejected: 0, auth. in progress: 0, getting IP by DHCP: 0
  Codes: L - local, R - remote AAA, U - unknown, N - not specified
IP Address      MAC Address      Port      S-tag  C-tag  Status      Type

```

-----	-----	-----	-----	-----	-----
20.20.20.2	3e3a.6af3.6edd	tel	-----	-----	accepted(L) IPoE
20.20.20.3	7e6e.5221.bf2a	tel	-----	-----	accepted(L) IPoE

Для просмотра подробной информации по конкретному абоненту при вызове команды следует указать адрес абонента: **show subscribers <INAME> <IP>**, где <INAME> - имя интерфейса, <IP> - адрес абонента.

Пример:

```
ecorouter#sh subscribers bmi.1 20.20.20.2
ip: 20.20.20.2
mac: 3E:3A:6A:F3:6E:DD
port: tel
service: ddff
session timeout: 3 min
session time remaining: 0 min
idle timeout: 3 min
idle time remaining: 0 min
authentification status: accepted
type: IPoE
encapsulation: untagged
wan pkts: 67
lan pkts: 4
wan bytes: 96.614 K (96614)
lan bytes: 3.164 K (3164)
```

Для проверки сконфигурированных абонентских сервисов служит команда **show subscriber-service <SNAME>**, где <SNAME> - имя сервиса.

Пример:

```
ecorouter#sh subscriber-service test
Subscriber-service "test" is applied for:
  SUB-MAP
  ipoe_test
  ipoe_test2
Subscriber-policy:
  CCC
  BBB
  AAA
```

В результате выполнения команды будет показана информацию по subscriber-policy, service-policy, а также выведен список карт абонентов, на которых применен указанный сервис.

Для просмотра счетчиков по CoA и Disconnect запросам служит команда **show counters subscribers coa-messages**.

Пример:

CoA-Messages				
Remote	CoA-Req	CoA-ACK	CoA-NAK	Drops
-----	-----	-----	-----	-----
1. 1. 1. 2	3	2	1	3

192.168.255.	2	0	0	0	0
Total	3	2	1	3	
Disconnect-Messages					
Remote	Disc-Req	Disc-ACK	Disc-NAK	Drops	
-----	-----	-----	-----	-----	-----
1. 1. 1. 2	1	1	0	3	
192.168.255.	2	0	0	0	
Total	1	1	0	3	

В результате выполнения команды будут показаны две таблицы с количеством пришедших запросов, а также количеством ответов ACK, NAK и количеством отброшенных запросов.

21.1.5 Функционал ARP Proxy

При настройке функционала IPoE у абонентов, находящихся в одной подсети, но в разных VLAN, отсутствует связность. В некоторых случаях требуется обеспечить связность между абонентами. Для этого на BMI-интерфейсе используется функционал ARP Proxy. ARP Proxy позволяет в случае ARP-запроса со стороны абонента ответить MAC-адресом самого BMI-интерфейса (если MAC-адрес присутствует в ARP-таблице маршрутизатора). Таким образом абоненты (или устройства) в одной подсети могут связываться между собой.

Функционал ARP Proxy по умолчанию выключен. Для включения ARP Proxy используется команда **proxy-arp** в режиме конфигурации BMI-интерфейса.

Команда **show interface bmi.<Номер>** используется для проверки текущего статуса ARP Proxy.

Пример:

show interface bmi.1

```
Interface bmi.1 is up
Snmp index: 7
Ethernet address: 1c87.7640.8002
MTU: 1500
NAT: no
session-trigger ip
```

ARP proxy is disabled

```
CMP redirection is on
Label switching is disabled
<UP,BROADCAST,RUNNING,MULTICAST>
Connect port te0 service instance static symmetric
Connect port te0 service instance dynamic symmetric
net 1.1.1.1/24 broadcast 1.1.1.255/24
total input packets 23870, bytes 35354935
total output packets 49700, bytes 49917061
```

21.2 Настройки PPPoE

Для создания профиля PPPoE используется команда **pppoe-profile <NAME>** конфигурационного режима, где <NAME> – название профиля PPPoE, длина названия не более 15 символов.

После ввода команды создается указанный профиль PPPoE и производится переход в контекстный режим конфигурации pppoe-profile. Приглашение в командной строке изменит вид на следующий:

```
ecorouter(config-pppoe) #
```

В данном режиме доступны следующие команды:

PPPoE configuration commands:	
description	Profile description
dns	DNS IP address
exit	Exit from the current mode to the previous mode
gateway	Gateway IP address
help	Description of the interactive help system
no	Negate a command or set its defaults
pado-timeout	PADO timeout
pool	Set the IP address pool
ppp	Point-to-Point Protocol
set	Set policies
show	Show running system information
tag-ac-name	Set access concentrator name tag
tag-service-name	Set service name tag

Часть настроек выполняется с использованием ключевого слова **set** (см. раздел "Команды set для конфигурирования PPPoE").

```
?corouter(config-pppoe) #set
  aaa                      Set subscriber AAA profile
  idle-timeout             Set idle timeout
  session-timeout          Set session timeout
  subscriber-service        Set subscriber service
  update-interval          Set update interval
```

Таблица 92

Команда	Описание
dns	Задать IP-адрес DNS. Допускается создание одной (primary) или двух (primary и secondary) записей. Подробнее см. пример ниже
gateway	Задать IP-адрес шлюза
pado-timeout <0-65535>	Задать величину задержки между получением PADI и ответом PADO в миллисекундах. Диапазон значений 0-65535
pool	Задать пул IP-адресов (см. раздел "Пул IP-адресов для PPPoE клиентов")
ppp	Команды для настройки Point-to-Point Protocol (см. раздел "Point-to-Point Protocol")
set	Команды для задания политик (см. раздел "Команды set для конфигурирования PPPoE")
tag-ac-name <ACNAME>	Задать значение тега PPPoE AC-name, которое будет отображаться в ответном PADO пакете
tag-service-name <SRVNAME>	Задать значение тега PPPoE service-name, которое будет отображаться в ответном PADO пакете

Пример создания, конфигурации и просмотра PPPoE-профиля:

```
ecorouter(config)#pppoe-profile 1
ecorouter(config-pppoe)#dns ipv4 192.168.10.100
ecorouter(config-pppoe)#dns ipv4 192.168.10.200 secondary
ecorouter(config-pppoe)#pado-timeout 50
ecorouter(config-pppoe)#tag-ac-name ER-1
```

```
ecorouter(config-pppoe)#tag-service-name Srv1
```

Для просмотра информации о профилях PPPoE в режиме оператора используется команда **show pppoe-profile [<NAME>]**, где <NAME> - название профиля PPPoE. При вызове команды без указания имени будет показана информация по всем существующим профилям PPPoE.

Пример:

```
ecorouter#show pppoe-profile 111
pppoe-profile 111
  AAA profile: 111111
  Service: SUB_SERV
  AC-Name tag: ER-1
  Service-Name tags: Srv1
  PADO timeout: 50
  PPP options
    Authentication: no
    Configure-Request limit: 10
    Configure-Nak limit: 5
    Terminate-Request limit: 1
    Echo-Request limit: 5
    Retry timeout: 3
    Echo timeout: 10
  Gateway address: 192.168.10.1
  Primary DNS address: 192.168.10.100
  Secondary DNS address: 192.168.10.200
  IPv4 pool: dead

ecorouter#show pppoe-profile
pppoe-profile 111
  AAA profile: 111111
  AC-Name tag: ER-1
  Service-Name tags: Srv1
  PPP options
    Authentication: no
    Configure-Request limit: 10
    Configure-Nak limit: 5
    Terminate-Request limit: 1
    Echo-Request limit: 5
    Retry timeout: 3
    Echo timeout: 10
  Gateway address: 192.168.10.1
  Primary DNS address: 192.168.10.100
  Secondary DNS address: 192.168.10.200
  IPv4 pool: dead

pppoe-profile 2
  AAA profile: 111111
  AC-Name tag: ER-2
  Service-Name tags: Srv2
  PPP options
    Authentication: no
    Configure-Request limit: 10
    Configure-Nak limit: 5
    Terminate-Request limit: 1
```

```

Echo-Request limit: 5
Retry timeout: 3
Echo timeout: 10
Gateway address: 192.168.10.2
Primary DNS address: 192.168.10.101
Secondary DNS address: 192.168.10.201
IPv4 pool: 111

```

Просмотр счетчиков для PPPoE-абонентов аналогичен просмотру счетчиков для IPoE-абонентов (подробнее см. раздел "**Команды просмотра карт абонентов и сервисов абонентов**").

Пример вывода одного из вариантов команды **show subscribers** приведен ниже.

```

ecorouter> show subscribers bmi.1 192.168.10.2
ip: 192.168.10.2
mac: 12:34:56:78:9A:10
port: ge0
service: default(L)
session timeout: 1440 min
session time remaining: 1440 min
idle timeout: 30 min
idle time remaining: 30 min
PPPoE session-id: a3af
authentification status: accepted(L)
type: PPPoE
encapsulation: untagged
wan pkts: 1
lan pkts: 1
wan bytes: 98
lan bytes: 106

```

21.2.1 Протокол Point-to-Point (PPP)

Настройка параметров Point-to-Point Protocol производится в контекстном режиме конфигурирования профиля PPPoE (config-pppoe). Для конфигурации PPP доступны следующие команды:

```
?corouter(config-pppoe)#ppp
authentication      Authentication
max-configure       Configure-Request limit
max-echo            Echo-Request limit
max-failure         Configure-Nak limit
max-terminate        Terminate-Request limit
timeout-echo         Echo timeout
timeout-retry        Client response timeout
```

Подробнее см. таблицу ниже.

Таблица 93

Параметр с диапазоном значений	Описание
authentication	Настройка аутентификации (подробнее см. в разделе "Аутентификация PPPoE")
max-configure <1-20>	Максимальное количество запросов Configure-Request перед получением ответа (значение по умолчанию 10)

Параметр с диапазоном значений	Описание
max-failure <1-10>	Максимальное количество запросов Configure-Nak (значение по умолчанию 5)
max-echo <1-10>	Максимальное количество запросов Echo-Request перед получением ответа (значение по умолчанию 5)
max-terminate <1-10>	Максимальное количество запросов Terminate-Request (значение по умолчанию 1)
timeout-echo <1-10>	Количество секунд перед повторной отсылкой запроса Echo-Request (значение по умолчанию 10)
timeout-retry <1-10>	Количество секунд перед повторной отсылкой запроса Configure-Request/Configure-Terminate (значение по умолчанию 3)

21.2.2 Пул IP-адресов

В EcoBNGOS необходимо создать пул IP-адресов для выдачи их PPPoE-абонентам.

Создание пула IP-адресов производится с помощью команды конфигурационного режима **ip pool <IP_POOL> <RANGE>**, где **IP_POOL** - имя пула, **RANGE** - диапазон IP-адресов. Диапазон может состоять из одного или нескольких IP-адресов и интервалов IP-адресов, разделенных запятыми ",". Интервал задается начальным и конечным IP-адресом, разделенными символом минус "-".

Пример:

```
ecorouter(config)#ip pool 111 1.1.1.1,2.2.2.2-3.3.3.3
```

Для удаления пула IP-адресов используется команда конфигурационного режима **no ip pool <IP_POOL>**.

Для просмотра информации по пулу IP-адресов используется команда **show ip pool**. В результате выполнения этой команды будет показана информация по всем существующим пулам.

```
ecorouter#show ip pool
Pool      Begin          End          Free       In use
-----
0        192.168.10.2   192.168.10.254 1        252
0        192.168.12.2   192.168.12.2   10       243
```

Для просмотра информации по выбранному пулу используется команда **show ip pool <IP_POOL>**.

```
ecorouter#show ip pool 111
Pool      Begin          End          Free       In use
-----
111     1.1.1.1         1.1.1.1       1        0
2.2.2.2   3.3.3.3       16843010    0
```

Для назначения пула выделяемых по умолчанию IP-адресов используется команда **pool ipv4 <IP_POOL>** контекстного режима конфигурации (**config-pppoe**), где **IP_POOL** - имя пула.

Для отмены назначения пула выделяемых по умолчанию IP-адресов используется команда **no pool ipv4 <IP_POOL>**.

21.2.3 Аутентификация PPPoE

В EcoBNGOS поддерживается PPPoE-аутентификация абонентов.

Для выбора протоколов аутентификации необходимо выполнить следующие шаги:

1. Перейти в контекстный режим конфигурирования PPPoE-профиля.
2. Включить аутентификацию через PPPoE.
3. Указать группу RADIUS-серверов, которые будут использованы для удаленной аутентификации.

Подробнее шаги описаны ниже.

Для перехода в контекстный режим конфигурирования PPPoE-профиля следует в конфигурационном режиме выполнить команду **pppoe-profile <NAME>**, где **<NAME>** - имя профиля. Если профиль до этого не существовал, он будет создан.

```
ecorouter(config)#pppoe-profile 1  
ecorouter(config-pppoe) #
```

Для выбора протокола аутентификации следует воспользоваться командой **ppp authentication**, возможные варианты которой показаны ниже.

```
?corouter(config-pppoe)#ppp authentication  
chap Challenge Handshake Authentication Protocol  
ms-chap Microsoft PPP CHAP Extensions  
ms-chap-v2 Microsoft PPP CHAP Extensions v2  
pap Password Authentication Protocol
```

После того, как протокол аутентификации выбран, следует добавить группу RADIUS-серверов для профиля PPPoE при помощи команды **set aaa** (данная команда выполняется в контекстном конфигурационном режиме (config-pppoe)). Профиль должен быть предварительно создан (см. раздел **IP over Ethernet**).

ВНИМАНИЕ: аутентификация производится только при помощи RADIUS-серверов, локальная аутентификация не поддерживается.

21.2.4 Особенность подключения PPPoE-абонента

При подключении PPPoE-абонента происходит автоматическое добавление маршрута в таблицу FIB с маской /32, при этом в таблице RIB этот маршрут не отображается. Трафик от абонента в таком случае может передаваться даже без указания IP-адреса на bmi-интерфейсе.

В случае если необходимо анонсировать сеть, выданную PPPoE-абонентам, через динамические протоколы маршрутизации, то существует несколько способов решить данную задачу:

1. Задать адрес на bmi-интерфейсе из PPPoE-подсети и включить интерфейс bmi в протокол динамической маршрутизации так же, как и обычный IP-интерфейс.
2. Создать статический маршрут до PPPoE-абонентов через NULL-интерфейс и перераспределить (redistribute) этот маршрут в процесс протокола динамической маршрутизации. При таком варианте ответный трафик, пришедший на маршрутизатор, не будет отброшен, так как в FIB будут более специфичные /32 маршруты до абонентов.

21.2.5 Команда просмотра состояния PPPoE сессии

Состояние PPPoE сессии можно посмотреть с помощью команды **show interface bmi.0 pppoe clients**:

```
?corouter#show interface bmi.0 pppoe clients
| Output modifiers
> Output redirection
<cr>
```

В результате выполнения команды отображается таблица с основными характеристиками состояния сессии, в том числе и для еще не установленной (пояснения вывода см. в таблицах ниже):

```
ecorouter#show interface bmi.0 pppoe clients
MAC Address C-tag S-tag Port ID Service PPP-State PPP-
Auth User IP Address
-----
-----
2a62.55af.4c6f 30 30 te2 63651 serv1 network pap adm
in 192.168.10.2
```

Таблица 94

Параметр	Пояснение
MAC Address	Физический адрес устройства
C-tag	Внутренний тег
S-tag	Внешний тег
Port	Физический порт маршрутизатора для подключения абонента
ID	ID сессии
Service	Сервис для сессии
PPP-State	Состояние сессии
PPP-Auth	Состояние авторизации
User	Логин пользователя
IP Address	Выданный абоненту IP address

Параметр **PPP-State** может принимать следующие значения:

Таблица 95

Значение	Пояснение
down	physical-layer not ready
establish	Link Establishment Phase
authenticate	Authentication Phase
network	Network-Layer Protocol Phase
terminate	Link Termination Phase

Параметр **PPP-Auth** может принимать следующие значения:

Таблица 96

Значение	Пояснение
none	Без аутентификации
pap	Аутентификации по протоколу PAP
chap	Аутентификации по протоколу CHAP

Значение	Пояснение
ms-chap-v1	Аутентификации по протоколу MS-CHAPv1
ms-chap-v2	Аутентификации по протоколу MS-CHAPv2

21.2.6 Параметры PPPoE при аутентификации через RADIUS-сервер

Протокол PAP (Password Authentication Protocol)

При аутентификации PPPoE-абонента через RADIUS-сервер с использованием протокола PAP EcoRouter отправляет RADIUS access request со следующей информацией:

Service-Type - тип сервиса, который запросил клиент, для PPPoE это всегда "Framed";

User-Name - логин абонента;

User-Password - пароль абонента в зашифрованном виде;

Calling-Station-Id - MAC-адрес абонента;

NAS-Identifier - имя маршрутизатора, указанное в hostname;

NAS-Port-Id - имя порта маршрутизатора:имя интерфейса:c-vlan:s-vlan - порт и интерфейс указываются те, на которые пришёл пакет-триггер (пакет, ставший триггером для отправки запроса на RADIUS-сервер). Метки vlan указываются те, которые присутствовали в заголовке пакета-триггера;

NAS-Port-Type - тип порта, на который пришёл пакет-триггер;

Acct-Session-Id - идентификатор абонентской сессии - идентификатор генерируется маршрутизатором на основе следующих ключей - ip-адрес абонента и время поднятия сессии;

NAS-IP-Address - IP-адрес, идентифицирующий маршрутизатор - если на устройстве создан интерфейс loopback.0 и на него назначен IP-адрес, то в этот атрибут будет записан адрес с интерфейса loopback.0. Если интерфейс loopback.0 отсутствует в конфигурации маршрутизатора, то в этот атрибут будет записан IP-адрес с интерфейса, с которого был отправлен RADIUS access request;

Framed-Protocol - тип инкапсулирующего протокола - варианты для заполнения атрибута в текущей реализации - 1. PPP;

NAS-Port - c-vlan - внутренняя метка vlan из заголовка пакета-триггера.

Протокол CHAP (Challenge Handshake Authentication Protocol)

При аутентификации PPPoE абонента через RADIUS-сервер с использованием протокола CHAP EcoRouter отправляет вместо атрибута User-Password следующие атрибуты:

CHAP-Password - md5 хэш на основе пароля абонента и challenge;

CHAP-Challenge - генерируемое маршрутизатором случайное значение, необходимое для генерации chap-password.

Остальные атрибуты совпадают с атрибутами при использовании протокола PAP.

Параметры accounting request

После аутентификации абонента, если для него была заведена сессия, маршрутизатор отправляет accounting request сообщения со следующей информацией:

Acct-Status-Type - тип accounting request сообщения - в текущей реализации может принимать значения - start, stop и interim-update;

Acct-Session-Id - идентификатор абонентской сессии - идентификатор генерируется маршрутизатором на основе следующих ключей - ip-адрес абонента и время поднятия сессии;

Event-Timestamp - время отправки сообщения;

Framed-IP-Address - IP-адрес абонента;

User-Name - логин абонента;

NAS-Port - c-vlan - внутренняя метка vlan из заголовка пакета-триггера.

NAS-Identifier - имя маршрутизатора, указанное в hostname;

NAS-Port-Id - имя порта маршрутизатора:имя интерфейса:c-vlan:s-vlan - порт и интерфейс указываются те, на которые пришёл пакет-триггер (пакет, ставший триггером для отправки запроса на RADIUS-сервер). Метки vlan указываются те, которые присутствовали в заголовке пакета-триггера;

NAS-Port-Type - тип порта, на который пришёл пакет-триггер;

NAS-IP-Address - IP-адрес, идентифицирующий маршрутизатор - если на устройстве создан интерфейс loopback.0 и на него назначен IP-адрес, то в этот атрибут будет записан адрес с интерфейса loopback.0. Если интерфейс loopback.0 отсутствует в конфигурации маршрутизатора, то в этот атрибут будет записан IP-адрес с интерфейса, с которого был отправлен RADIUS access request;

Service-Type - тип сервиса, который запросил клиент, для PPPoE это всегда "Framed";

Framed-Protocol - тип инкапсулирующего протокола - варианты для заполнения атрибута в текущей реализации - 1. PPP;

Acct-Authentic - способ аутентификации абонента - в текущей реализации может принимать значения - radius и local;

Event-Timestamp - дата и время отправки сообщения;

Acct-Status-Type - start/stop/Interim-Update;

Calling-Station-Id - MAC-адрес абонента;

Acct-Session-Time - текущее время жизни сессии;

Acct-Input-Packets - количество пакетов, отправленных абонентом в течение сессии;

Acct-Input-Octets - количество байт, отправленных абонентом в течение сессии;

Acct-Input-Gigawords - количество переполнений счетчика Acct-Input-Octets;

Acct-Output-Packets - количество пакетов, отправленных абоненту в течение сессии;

Acct-Output-Octets - количество байт, отправленных абоненту в течение сессии;

Acct-Output-Gigawords - количество переполнений счетчика Acct-Output-Octets;

Acct-Delay-Time - время, которое было затрачено на отправку accounting request сообщения;

Acct-Terminate-Cause - причина, по которой сессия была сброшена маршрутизатором, в текущей реализации может принимать следующие значения:

- Idle Timeout (истечение idle-timeout),
- Session Timeout (истечение session-timeout),
- Admin Reset (выполнение команды **clear subscribers**),
- Port Error (удаление или выключение соответствующего bmi-интерфейса),
- Service Unavailable (запрос RADIUS-сервером не настроенного на маршрутизаторе сервиса).

21.3 Общие настройки для BRAS

21.3.1 Настройка скорости доступа для абонентов BRAS

Приведенные ниже сведения относятся как к абонентам IPoE, так и PPPoE.

Для настройки скорости доступа для профиля (IPoE / PPPoE) необходимо создать subscriber-service. Созданный subscriber-service может быть привязан к профилю вручную или получен с RADIUS-сервера:

```
?corouter(config)#subscriber-service
SUBSCRIBER_SERVICE  Subscriber service name
```

Для subscriber-service следует назначить subscriber-policy.

```
?corouter(config-sub-service)#set
policy  Set policy
?corouter(config-sub-service)#set policy
SUBSCRIBER_POLICY_NAME  Subscriber policy name
<cr>
```

В subscriber-policy указывается скорость абонента для upstream- и downstream-пакетов в kbps и применяется filter-map policy также для upstream и downstream:

```
ecorouter(config)#subscriber-policy <NAME>
?corouter(config-sub-policy)#bandwidth
in Upstream packets
out Downstream packets
?corouter(config-sub-policy)#bandwidth in
kbps Bandwidth value in kbps
?corouter(config-sub-policy)#bandwidth in kbps
<64-10000000> Kbits per second
ecorouter(config-sub-policy)#bandwidth in kbps
?corouter(config-sub-policy)#set filter-map
in Upstream packets
out Downstream packets
?corouter(config-sub-policy)#set filter-map in
FILTER_MAP_POLICY_IPV4 Filter map name
ecorouter(config-sub-policy)#set filter-map in
```

В filter-map policy указывается параметр, по которому к абонентам будут применяться настройки.

```
?corouter(config)#filter-map policy ipv4
FILTER_MAP_POLICY_IPV4 Filter map name
```

```
?corouter(config)#filter-map policy ipv4 <NAME>
<0-65535> Sequence number
<cr>
ecorouter(config)#filter-map policy ipv4 <NAME> 10
```

Например:

```
filter-map policy ipv4 <NAME> 10
match any any any
set accept
```

После настройки subscriber-service можно вручную задать его применение в профиле (пример с pppoe-profile):

```
?corouter(config-pppoe)#set subscriber-service
SUBSCRIBER_SERVICE Specify subscriber service name
```

Также subscriber-service применится, если RADIUS-сервер передаст атрибут с указанием этого сервиса.

Ниже приведен пример полной настройки для PPPoE:

1. filter-map policy:

```
ecorouter(config)#filter-map policy ipv4 50kk 10
ecorouter(config-filter-map-policy-ipv4)#match any any any
ecorouter(config-filter-map-policy-ipv4)#set accept
```

2. subscriber-policy:

```
ecorouter(config)#subscriber-policy 50kk
ecorouter(config-sub-policy)#bandwidth in kbps 500032
ecorouter(config-sub-policy)#bandwidth out kbps 500032
ecorouter(config-sub-policy)#set filter-map in 50kk
ecorouter(config-sub-policy)#set filter-map out 50kk
```

3. subscriber-service

```
ecorouter(config)#subscriber-service 50kk
ecorouter(config-sub-service)#set policy 50kk
```

4.1 задание subscriber-service:

Применение subscriber-service вручную к pppoe-profile:

```
ecorouter(config)#pppoe-profile 0
ecorouter(config-pppoe)#set subscriber-service 50kk
```

4.2 в случае применения сервиса с RADIUS-сервера на нем необходимо задать атрибут.

5. После установки соединения состояние сервиса можно посмотреть командой **show subscribers <interface bmi> <ip addr>**.

5.1 в случае задания subscriber-service вручную после сервиса будет добавлено "(L)", что означает local.

```
ecorouter#show subscribers bmi.0 192.168.10.2
...
service: 50kk(L)
...
```

5.2 в случае получения subscriber-service от RADIUS-сервера после названия сервиса будет добавлено "(R)", что означает remote aaa.

```
ecorouter#show subscribers bmi.0 192.168.10.2
```

```
...
service: 50kk(R)
...
```

22 SNMP

SNMP (англ. Simple Network Management Protocol – простой протокол сетевого управления) – стандартный интернет-протокол для управления устройствами в IP-сетях на основе архитектур TCP/UDP. С помощью протокола SNMP, программное обеспечение для управления сетевыми устройствами может получать доступ к информации, которая хранится на управляемых устройствах (например, на коммутаторе). На управляемых устройствах SNMP хранит информацию об устройстве, на котором он работает, в базе данных, которая называется MIB.

SNMP является одним из протоколов, реализующих концепцию технологий управления сетью Internet Standard Management Framework.

В рамках данной концепции для управления сетью строится система, состоящая из трех основных элементов:

- SNMP manager управляет и наблюдает за сетевой активностью устройств. Его часто называют Network Management System (NMS);
- SNMP agent – программное обеспечение, запускаемое на управляемом устройстве, либо на устройстве, подключенном к интерфейсу управления управляемого устройства. Собирает данные с управляемого устройства и передает их на SNMP manager;
- Management Information Base (MIB) – база данных, которая используется для управления устройствами в сети. Имеет древовидную структуру в которой хранится информация о хостах. Элементы MIB имеют символьные имена и соответствующие им числовые значения – OID (формата N.N.N....N).

EcoRouter поддерживает версии протокола SNMPv1, SNMPv2c и SNMPv3.

22.1 Запуск и остановка сервиса SNMP

Для запуска SNMP сервиса используется команда конфигурационного режима **snmp-server enable snmp (mgmt | vr <VR_NAME | default>)**.

При запуске SNMP указывается, какие порты будет обслуживать сервис:

mgmt - management-порт,

vr - порты виртуального маршрутизатора.

Если значение данного параметра не указывается, то SNMP будет включен для management-порта.

```
ecorouter(config)#snmp-server enable snmp vr virt1
```

Если SNMP включается на виртуальном маршрутизаторе, для него необходимо разрешить входящий трафик на UDP-порт 161 через настройку профилей безопасности (подробнее о профилях безопасности можно прочитать в соответствующем разделе).

Для того чтобы переключить SNMP на другой виртуальный маршрутизатор, необходимо сначала выключить SNMP, а потом включить снова с указанием нужного виртуального маршрутизатора.

Пример настройки профиля безопасности и переключения сервиса на другой виртуальный маршрутизатор:

```
ecorouter(config)#security-profile 2
ecorouter(config-security-profile)#rule 0 permit udp any any eq 161
ecorouter(config-security-profile)#ex
ecorouter(config)#virtual-router virt2
ecorouter(config-vr)#ex
ecorouter(config)#security vr virt2 2
ecorouter(config)#no snmp-server enable
ecorouter(config)#snmp-server enable snmp vr virt2
```

Для выключения SNMP сервиса используется команда конфигурационного режима **no snmp-server enable snmp**.

```
ecorouter(config)#no snmp-server enable snmp
```

Для переподключения определенного протокола к SNMP в EcoRouter используется команда конфигурационного режима **snmp restart <bgp | isis | ldp | mrib | ospf | pim | rib | vrrp>**.

```
ecorouter(config)#snmp restart bgp
```

22.2 Настройка SNMP community

SNMP community - ключевое слово, имя объединения (сообщества) для взаимодействия по протоколу SNMP 1 или 2 версии. Сообщество состоит из одного или нескольких агентов и менеджеров. Один хост с установленным на нем агентом может одновременно принадлежать к нескольким сообществам, при этом агент будет принимать запросы только от устройств управления, принадлежащих к этим группам. Безопасность обмена сообщениями между агентами и менеджером в этом случае обеспечивается при помощи передачи в теле сообщения в открытом виде имени сообщества или community-string.

Для создания **community** используется команда конфигурационного режима **snmp-server community**. Синтаксис команды: **snmp-server community <COMMUNITY-NAME> (view <VIEW-NAME> (ro | rw) | (group <GROUP-NAME>) | (ro | rw))**.

Таблица 97

Параметр	Описание
<COMMUNITY-NAME>	Community-string. Максимальная длина 32 символа
view <VIEW-NAME>	Указать имя представления, определяющего поддерево MIB, доступное данному сообществу. Представление должно быть предварительно создано командой snmp-server view
<GROUP-NAME>	Имя группы
ro	Доступ только на чтение - значение выставляется по умолчанию
rw	Доступ на чтение и запись, если она разрешена

```
ecorouter(config)#snmp-server community MyComm view MyView1 version v2c
rw
```

Для сообщества нельзя одновременно указать и представление, и группу. Если не указано ни представление, ни группа, а только имя сообщества, то данному сообществу будет предоставлен доступ из любой сети ко всем MIB.

Для удаления **community** используется команда конфигурационного режима **no snmp-server community <COMMUNITY-NAME>**.

22.3 Настройка представлений (SNMP views)

Представления создаются для того, чтобы ограничить доступ к объектам дерева MIB. Для создания и настройки представления используется команда конфигурационного режима **snmp-server view**. Синтаксис команды: **snmp-server view <VIEW-NAME> <OID-TREE> (included | excluded)**.

Таблица 98

Параметр	Описание
<VIEW-NAME>	Имя представления. Максимальная длина 32 символа
<OID-TREE>	Идентификатор поддерева MIB, которое должно быть включено в представление или исключено из него. Указывается в виде строки из цифр, разделенных точками, например, 1.3.6.2.4, может быть задано именем
included	Включить поддерево в SNMP представление
excluded	Исключить поддерево из SNMP представления

```
ecorouter(config)#snmp-server view myView3 1.3.6.1.6.3.18 excluded
```

Для добавления поддерева к существующему представлению (или исключения из него) используется эта же команда.

Для удаления представления используется команда конфигурационного режима **no snmp-server view <VIEW- NAME >**.

22.4 Настройка отправки асинхронных сообщений

При передаче информации между менеджерами и агентами в общем виде используются следующие сценарии:

менеджер отправляет запрос агенту и получает ответ;

менеджеру отправляется сообщение (агентом или другим менеджером), которое требует уведомления о получении (**inform**);

агент отправляет информацию о себе менеджеру без запроса с его стороны и без уведомления о получении (**trap**).

Для включения отправки **trap** сообщений используется команда **snmp-server enable traps**.

```
ecorouter(config)#snmp-server enable traps
```

Для отключения отправки **trap** сообщений используется команда **no snmp-server enable traps**.

```
ecorouter(config)#no snmp-server enable traps
```

Для того чтобы осуществлять отправку **trap** сообщений менеджеру или NMS, необходимо указать адрес нужного хоста и его настройки. Для этого используется команда **snmp-server host**. Синтаксис команды:

```
snmp-server host <A.B.C.D|HOSTNAME> (traps ( | version (1 | 2c)) | informs)
<COMMUNITY-STRING> (| udp-port <1-1024>)
```

Таблица 99

Параметр	Описание
A.B.C.D	IP сервера
HOSTNAME	DNS-имя сервера
traps	Отправлять сообщения типа trap (без уведомления). Параметр по умолчанию
informs	Отправлять сообщения типа inform (с уведомлением)
version	Версия протокола SNMP. Значения параметра: 1 или 2c
<COMMUNITY-STRING>	Community-string, от имени какого сообщества отправляются сообщения. Максимальная длина 32 символа
udp-port	Порт, который слушает сервер. Диапазон значений от 1 до 1024, по умолчанию 162

```
ecorouter (config)#snmp-server host 192.168.0.1 traps version 1
MyCommPass
```

Если в параметрах указывается отправка сообщений типа **inform**, то параметр **version** не задается, так как он может быть равен только **v2c**.

Для удаления записи о менеджере или NMS используется команда **no snmp-server host**.

```
ecorouter (config)#no snmp-server host < A.B.C.D | HOSTNAME >
```

22.5 SNMPv3

Протокол SNMPv3 - это следующая стадия развития протокола SNMP. Он полностью совместим с предыдущими версиями. Отличие от предыдущих версий:

понятия "менеджер" и "агент" заменены на "сущность" (entity), понятия "агент" и "менеджер" остались в качестве ролей;

стали доступны службы ограничения доступа, защиты данных и аутентификации пользователя (см. стандарты RFC 3411-3415).

В версии SNMPv3 предусмотрено три уровня безопасности:

noAuthNoPriv - аутентификация не производится, конфиденциальность данных отсутствует;

authNoPriv - аутентификация без конфиденциальности;

authPriv - аутентификация и шифрование, максимальный уровень защиты.

22.5.1 Операции с пользователем

Создание пользователя производится в режиме конфигурации при помощи команды **snmp-server user <USERNAME> [group <GROUPNAME>] [encrypted] [auth (md5 | sha) <AUTH-PASSWORD> [priv (des | aes) <PRIV-PASSWORD>]]**. Описание параметров вызова команды приведено в таблице ниже.

Таблица 100

Параметр	Описание
USERNAME	Имя пользователя

Параметр	Описание
GROUPNAME	Имя группы
encrypted	Указание этого параметра означает, что далее введен уже зашифрованный пароль (пароли), и к нему (к ним) хэширование применять уже не надо
auth (md5 sha)	Выбор алгоритма хэширования аутентификационного пароля. Если будет задан параметр priv (des aes), то пароль для шифрования сообщений в сессии также будет хэширован по выбранному алгоритму (md5 или sha)
AUTH-PASSWORD	Аутентификационный пароль
priv (des aes)	Выбор алгоритма шифрования на основе <PRIV-PASSWORD>. Выбор возможен, только если задействован параметр auth
PRIV-PASSWORD	Пароль для шифрования сообщений в сессии

Пользователь может входить только в одну группу или не входить ни в одну.

Удаление пользователя производится при помощи команды **no snmp-server user <USERNAME> [group <GROUPNAME>] [auth (md5 | sha) <AUTH-PASSWORD> [priv (des | aes) <PRIV-PASSWORD>]]**.

22.5.2 Операции с группой

Создание группы производится в режиме конфигурации при помощи команды **snmp-server group <GROUPNAME> (v1 | v2c | (v3 (auth | noauth | priv))) (read VIEW-NAME |) (write VIEW-NAME |).**

Таблица 101

Параметр	Описание
GROUPNAME	Имя группы
v1 v2c v3	Версии протокола SNMP
auth noauth priv	В зависимости от параметра в сессиях, соответствующих выбранной модели безопасности, пользователям будет предоставлен определенный доступ. При указании auth доступ к представлению этой группы будет предоставлен аутентифицированному пользователю, при указании noauth - неаутентифицированному, при указании priv - пользователю, использующему аутентификацию и шифрование
VIEW-NAME	Имя представления, определяющего поддерево MIB, доступное данной группе для чтения или записи соответственно. Представление должно быть предварительно создано командой snmp-server view

Редактирование группы выполняется той же командой, что и создание.

Каждая группа может быть настроена по-разному для работы с каждой версией SNMP. Для SNMPv3 возможны различные настройки для одной и той же группы для разных уровней безопасности.

```
ecorouter(config)#snmp-server group test v1 read view1 write view2
ecorouter(config)#snmp-server group test v2c read view3
ecorouter(config)#snmp-server group test v3 auth read view4 write view5
ecorouter(config)#snmp-server group test v3 priv write view6
```

Удаление группы производится при помощи команды **no snmp-server group <GROUPNAME> ((v1 | v2c | v3 (auth | noauth | priv)) (read VIEW-NAME |) (write VIEW-NAME |)).**

22.5.3 Команды просмотра

Просмотр информации о SNMP-пользователях производится в режиме администрирования при помощи команды **show snmp user [<USERNAME>]**. Если указать параметр <USERNAME>, то будет выведена информация о выбранном пользователе.

```
ecorouter#show snmp user MyUsEr
User name: MyUsEr
Group name: Gr1
Authentication: md5
Privacy: DES
```

В результате выполнения команды **show snmp user** будет выведена информация обо всех пользователях SNMP. Пример выполнения такой команды:

```
ecorouter#show snmp user
User name: MYSNMPUSER
Authentication: No
Privacy: No
User name: MyUsEr
Group name: Gr1
Authentication: md5
Privacy: DES
```

Просмотр информации о SNMP-группах производится в режиме администрирования при помощи команды **show snmp group [<GROUPNAME>]**. Если указать параметр <GROUPNAME>, то будет выведена информация о выбранной группе.

```
ecorouter#show snmp group 2
Group name: 2
Authentication: No
```

В результате выполнения команды **show snmp group** будет выведена информация обо всех группах SNMP. Если группа имеет отдельные настройки для разных версий протокола, то они будут показаны отдельно. Пример выполнения такой команды:

```
ecorouter#show snmp group
Group name: test
Security level: no Authentication
Snmp version: 1
Read view: view1
Write view: view2
Group name: test
Security level: no Authentication
Snmp version: 2c
Read view: view3
Group name: test
Security level: Authentication
Snmp version: 3
Read view: view4
Write view: view5
Group name: test
Security level: Authentication and Privacy
Snmp version: 3
Write view: view6
```

23 QoS

QoS (англ. quality of service – качество обслуживания) – этим термином называют вероятность того, что сеть связи соответствует заданному соглашению о трафике. Также QoS обозначает возможность гарантировать доставку пакетов, контроль пропускной способности, назначение приоритетов для разных классов сетевого трафика.

23.1 Архитектура QoS

В EcoRouter схема реализации QoS разделена логически на несколько взаимодействующих блоков:

- Классификатор/Classifier
- RED
- Планировщик/Scheduler

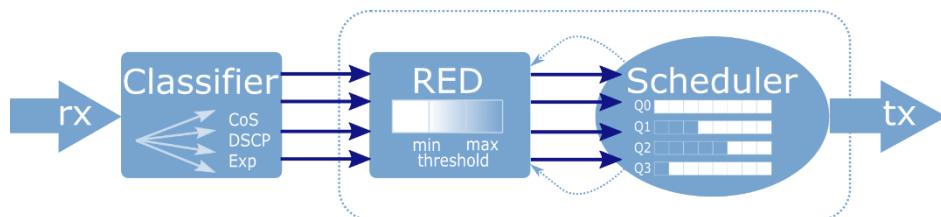


Рисунок 41

Трафик, приходящий на интерфейс, поступает в Классификатор, где ему присваиваются метки, в соответствии с установленными классами. Далее при помощи механизма RED происходит выравнивание трафика по предустановленным параметрам и данным, приходящим с Планировщика, и отбрасывается часть пакетов. После чего, пакеты ставятся в очереди Планировщика и пропускаются на выход по заданным правилам. Правила Планировщика начинают выполняться только в том случае, если объем трафик превышает заданное значение полисера.

Данная схема реализуется для каждого сервисного интерфейса.

Ниже более подробно описан каждый из блоков.

23.2 Классификация трафика

Для настройки классификации в EcoRouterOS необходимо использовать специальные карты классов, создать соответствующий профиль трафика и привязать его к экземпляру сервиса (service-instance). В таком случае входящие в service-instance пакеты могут быть классифицированы, т.е. обработаны и рассмотрены другим QoS-функционалом.

Карты классов создаются в конфигурационном режиме при помощи команды **class-map <NAME>**, где NAME может быть любой строкой, рекомендуемый формат имени – все буквы заглавные.

Пример:

```
ecorouter(config)# class-map VIDEO
ecorouter(config)# class-map IPVOICE
ecorouter(config)# class-map MYCLASS
```

При создании карты класса пользователь оказывается в режиме ее конфигурирования.

Пример:

```
ecorouter(config)# class-map VOICE
ecorouter(config-cmap) #?
Traffic classifier configuration commands:
exit  Exit from the current mode to the previous mode
help  Description of the interactive help system
match Classification criteria
no    Negate a command or set its defaults
set   Set marking values
show  Show running system information
```

В режиме конфигурации карты классов пользователю доступна команда **match**, которая позволит выделять определенные пакеты из общего потока трафика путем указания значения поля или его наименования в заголовках Ethernet, MPLS или IP. По значениям этих полей будет осуществляться классификация трафика. Введение нескольких правил **match** будет соответствовать логической операции «**ИЛИ**».

Пример:

```
ecorouter(config-cmap) #match ?
cos  IEEE 802.1Q class of service priority values
dscp Match DSCP in IP packets
exp  Match MPLS experimental
ecorouter(config-cmap) #match cos ?
<0-7> Enter class-of-service values
ecorouter(config-cmap) #match dscp ?
<0-63> Enter DSCP values
ecorouter(config-cmap) #match exp ?
<0-7> Enter MPLS exp values
```

Как видно из примера, классификация в EcoRouterOS может осуществляться по полям **cos**, **dscp** и **exp**. Значения могут задаваться только в десятичном виде. Можно задавать набор значений, используя в качестве разделителя запятую «,» или диапазон, используя в качестве разделителя дефис «-».

Для создания профилей трафика используется команда **traffic-profile <NAME>**, где **<NAME>** может быть любым наименованием, рекомендуемый формат имени – цифры или все буквы заглавные.

При создании профиля трафика пользователь оказывается в режиме его конфигурирования.

Пример:

```
ecorouter(config)# traffic-profile 1
ecorouter(config-traffic-profile)# ?
Traffic profile configuration commands:
  class Select a class to configure
  exit Exit from the current mode to the previous mode
  help Description of the interactive help system
  no Negate a command or set its defaults
  show Show running system information
```

Для привязки классов трафика к профилю используется команда **class** с указанием имени ранее сконфигурированной карты классов.

Пример:

```
ecorouter(config)#traffic-profile 1
ecorouter(config-profile)#class VIDEO
ecorouter(config-profile)#class IPVOICE
```

Для включения классификации, возможности обрабатывать пакеты отдельно друг от друга и применять различные политики в зависимости от типа поступающего трафика пользователь должен применить профиль трафика к заранее созданной политике. Сделать это можно с помощью команды в конфигурационном режиме **service-policy <NAME>**, где <NAME> может быть любым наименованием, рекомендуемый формат имени – цифры или заглавные буквы.

Пример:

```
ecorouter(config)#service-policy ECO
ecorouter(config-policy)#traffic-profile 1
```

Далее необходимо применить политику на экземпляре сервиса (**service-instance**) во входящем направлении. Классификация трафика в исходящем направлении невозможна.

Пример:

```
ecorouter(config)#port ge1
ecorouter(config-port)#service-instance test
ecorouter(config-service-instance)#service-policy ECO in
```

Пример включения классификации голосового и видео-трафика во входящем направлении по отношению к порту ge1:

```
ecorouter(config)#class-map VIDEO
ecorouter(config-cmap)#match dscp 1
ecorouter(config-cmap)#exit
ecorouter(config)# class-map IPVOICE
ecorouter(config-cmap)#match dscp 2
ecorouter(config-cmap)#exit
ecorouter(config)#traffic-profile TEST
ecorouter(config-traffic-profile)#class VIDEO
ecorouter(config-traffic-profile)#class IPVOICE
```

```
ecorouter(config-cmap)#exit
ecorouter(config)#service-policy ECO
ecorouter(config-policy)#traffic-profile TEST
ecorouter(config)#port ge1
ecorouter(config-port)#service-instance test
ecorouter(config-service-instance)#service-policy ECO in
```

Для проверки сконфигурированных параметров можно воспользоваться командами:

```
ecorouter#sh class-map
Class map default
Class map IP0
Match dscp: 2
Class map IP1
Match dscp: 4
Class map IP2
Match dscp: 8
Class map IP3
Match dscp: 12
show traffic-profile
Traffic profile prof-dscp
Class IP0
Class IP1
Class IP2
Class IP3
```

23.3 RED

Механизм RED действует как часть планировщика, предваряя его работу и основываясь на поступающих с него данных о загруженности очередей.

В общем виде, планировщик представляет собой механизм, распределяющий полосу пропускания в момент, когда передаваемого трафика больше, чем выделенной полосы пропускания. Такая ситуация называется Congestion. Она чревата тем, что в этот момент массово и одновременно происходит потеря во всех потоках трафика, за исключением малых потоков, чья скорость не превышает гарантированную. Массовая одновременная потеря пакетов приводит к тому, что TCP-сущности одновременно запускают механизм реинициализации TCP окна, и скорость всех потоков одновременно падает, после чего, одновременно растет. В итоге, график загрузки интерфейса выглядит пилюобразно, и реальная загрузка интерфейса никогда не принимает устоявшегося значения, т.е. интерфейс не используется полностью в одни моменты времени, и испытывает перегрузки в другие. Для того, чтобы избежать подобного поведения, применяется механизм RED.

Работа механизма RED заключается в случайном отбрасывании пакетов ранее, чем они поступят в очередь. Это позволяет добиться того, что TCP-сессии меняют размер окна попаременно. Вероятность отбрасывания пакетов в этом случае является адаптивным значением. Пользователем устанавливаются значения загруженности интерфейса, при которой вероятность становится отличной от 0 и начинает расти. Помимо этого, устанавливается максимальная вероятность отбрасывания пакета и значение загрузки интерфейса, при котором вероятность становится равной этому значению. При изменении загруженности

интерфейса в рамках этих двух скоростей вероятность отбрасывания растет от 0 до указанного максимального значения, согласно принятой математической функции, учитывающей среднюю загруженность полосы пропускания, количество пакетов, пропущенных без отбрасывания.

Настройка RED

Для включения механизма RED необходимо ввести команду **random-detect** в режиме конфигурирования планировщика.

Параметры механизма RED задаются при конфигурировании очередей в планировщике.

Для каждой очереди задаются две границы: минимальная и максимальная граница диапазона, из которого будут отбрасываться случайные пакеты (min/max threshold).

Границы задаются соответственно параметрами **red-min <NUM>** и **red-max <NUM>**. Так как в EcoRouterOS длина очередей определяется динамически, то значения могут быть установлены в диапазоне от 0% до 100% от максимальной для очереди скорости (PIR). Значение **red-min** не должно быть больше значения **red-max**.

Если значения обоих параметров **red-min** и **red-max** равны **0**, то механизм RED будет отключен.

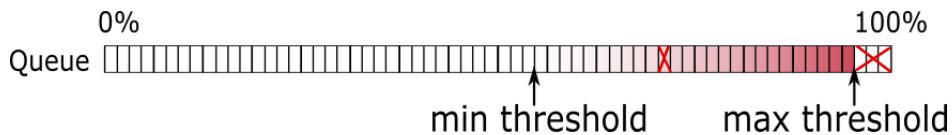


Рисунок 42

До достижения минимальной границы вероятность того, что пакет будет отброшен, равна нулю. После этого вероятность начинает расти до максимально возможного уровня, который регулируется параметром **red-inv-prob**. Этот параметр устанавливает значение знаменателя в дроби, определяющей вероятность отбрасывания пакета (**Probability = 1 / X**).

Значения параметра могут быть установлены в диапазоне [1 - 255]. Значение по умолчанию **10**.

При таком значении вероятность того, что пакет будет отброшен, равна 0,1 (**Probability = 1 / 10 = 0,1**), иными словами, будет отбрасываться каждый 10-ый пакет.

Настройка WRED

Механизм RED позволяет предотвращать переполнение очереди, относящейся к сервисному интерфейсу в целом.

Механизм WRED позволяет предотвращать переполнение любой сконфигурированной в планировщике очереди. Таким образом, позволяя настроить параметры WRED для каждой очереди в отдельности.

Для включения механизма WRED необходимо ввести команду **weighted-random-detect** в режиме конфигурирования планировщика.

Параметры механизма WRED задаются при конфигурировании очередей в планировщике.

Для каждой очереди задаются две границы: минимальная и максимальная граница диапазона, из которого будут отбрасываться случайные пакеты (min/max threshold).

Границы задаются соответственно параметрами **wred-min <NUM>** и **wred-max <NUM>**. Так как в EcoRouterOS длина очередей определяется динамически, то значения могут быть установлены в диапазоне от 0% до 100% от максимальной для очереди скорости (PIR). Значение **wred-min** не должно быть больше значения **wred-max**.

Если значения обоих параметров **wred-min** и **wred-max** равны **0**, то механизм WRED будет отключен.

До достижения минимальной границы вероятность того, что пакет будет отброшен, равна нулю. После этого вероятность начинает расти до максимально возможного уровня, который регулируется параметром **wred-inv-prob**. Этот параметр устанавливает значение знаменателя в дроби, определяющей вероятность отбрасывания пакета (**Probability = 1 / X**).

Значения параметра могут быть установлены в диапазоне **[1 - 255]**. Значение по умолчанию **10**.

При таком значении вероятность того, что пакет будет отброшен, равна 0,1 (**Probability = 1 / 10 = 0,1**), иными словами, будет отбрасываться каждый 10-ый пакет.

23.4 Планировщик/Scheduler

Планировщик управляет механизмом очередей. Под очередью (queue) в концепции EcoRouter понимается программно реализуемая очередь пакетов. Пакеты в такой очередидерживаются средствами планировщика до тех пор, пока не освободится место в аппаратной очереди (порт не станет доступным) для дальнейшей отправки пакетов.

В EcoRouter есть 8 очередей: queue 0 - queue 7. Приоритет очереди, обозначаемый ее номером, определяет порядок, в котором они обрабатываются (см. рисунок ниже). То есть, после передачи гарантированного объема трафика (CIR) первой будет обрабатываться очередь 0 с наивысшим приоритетом. Далее будет обрабатываться очередь 1, 2 и так далее.

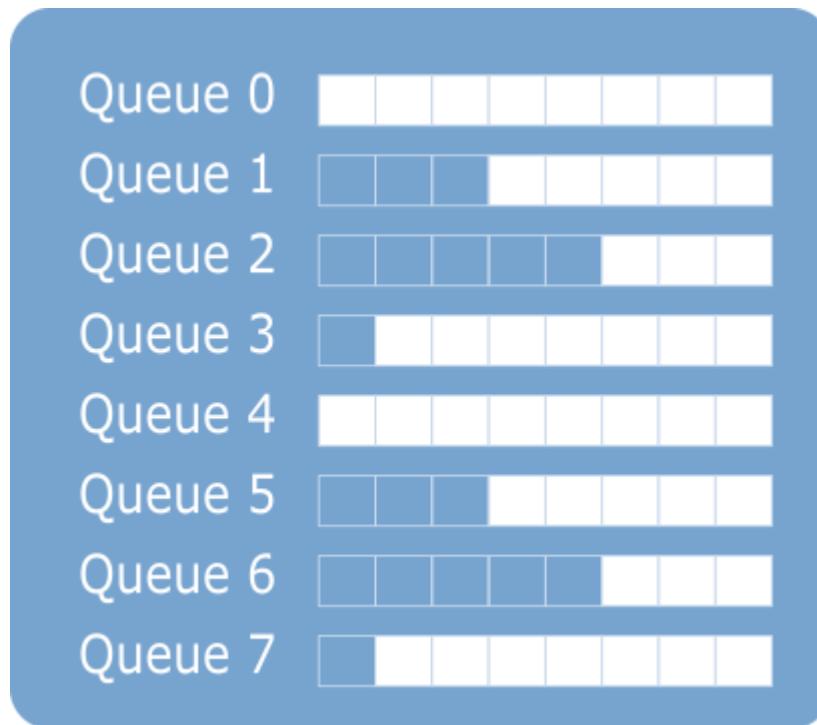


Рисунок 43

Размер каждой очереди динамически изменяется. Это необходимо для поддержания приемлемых значений полосы пропускания, задержки и дрожания фазы для не приоритетных очередей. Это придает гибкость при различных вариантах построения сети и типах передаваемого трафика. Сетевому администратору не придется задумываться о сохранении приемлемых значений параметров задержки и дрожания фазы, необходимо лишь задать полосу пропускания для конкретного типа трафика.

Очереди соотносятся с классами трафика, при этом возможны настройки, при которых часть трафика конкретного класса имеет больше гарантий по доставке. Это разделение происходит на основании количества трафика конкретного класса, переданного с начала итерации до определенного момента. Для этого вводятся понятия CIR и PIR.

CIR (Committed Information Rate) - это объем передаваемого за дельту времени трафика, который будет передан гарантированно. PIR (Peak Information Rate) - максимальное для очереди значение полосы пропускания. Трафик, превышающий PIR, будет безусловно отброшен. Если в других очередях есть трафик, он может вытеснить трафик, превышающий значение CIR, в соответствии с приоритетом.

Для каждой очереди можно задать параметры CIR и PIR в процентах или в абсолютном значении (Kbps). Также может быть задано значение **remainder**, отвечающий за выделение оставшейся незанятой части полосы пропускания.

Класс трафика очереди 7 по умолчанию - **default**. Это служебный класс, в который попадает любой трафик, не указанный остальными классах. Данный класс нельзя настроить, но можно назначить на любую очередь.

На схеме ниже представлен алгоритм обслуживания очередей планировщиком.

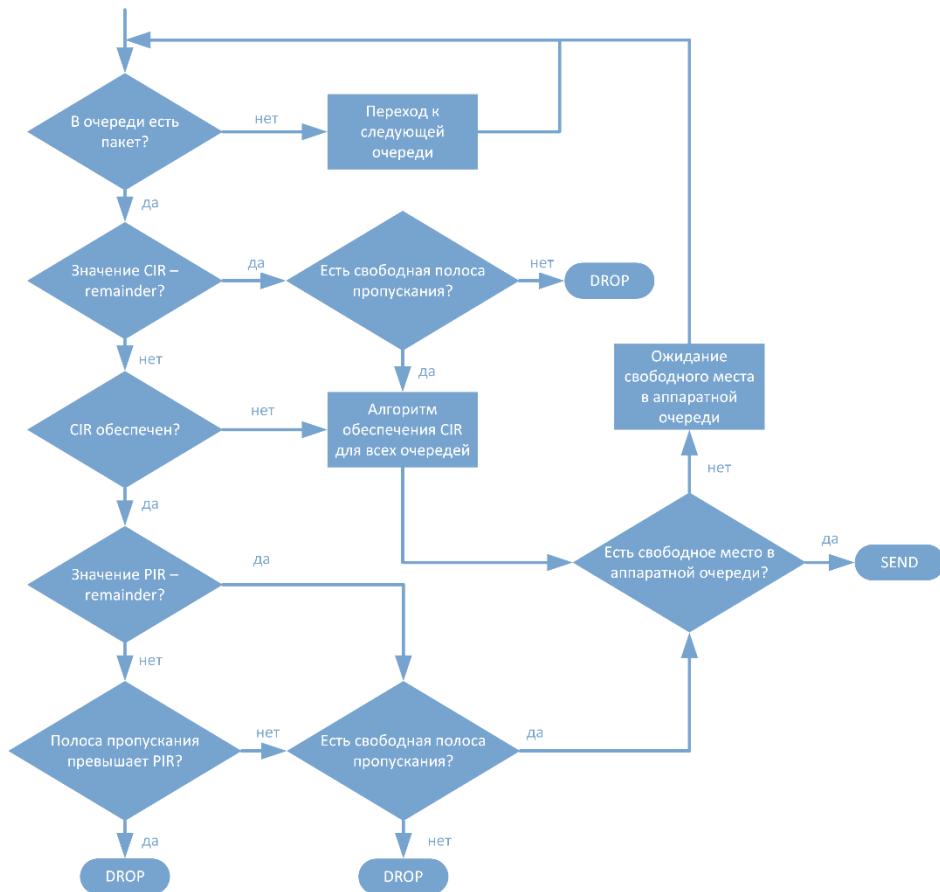


Рисунок 44

Как показано на рисунке, если в приоритетной очереди есть пакет, то планировщик сначала будет пытаться обеспечить указанный CIR для всех очередей и лишь затем распределять пакеты согласно приоритетам. После проверок обеспечения CIR и PIR для очереди пакет передается на сетевую карту и отправляется при наличии свободного места в аппаратной очереди. Если приоритетная очередь больше не содержит пакетов на передачу, то планировщик переходит к обработке пакетов из другой очереди. Затем процесс повторяется вновь через приоритетную очередь.

23.4.1 Настройка планировщика и очередей

Для создания планировщика в конфигурационном режиме используется команда: **traffic-scheduler pqwrr.<NUM>**.

Название планировщика обязательно должно начинаться с префикса "pqwrr".

Далее в созданном планировщике задаются очереди.

Синтаксис команды: **queue <0-31> class <NAME> cir <CIR> pir <PIR> (wred-min <0-100> wred-max <0-100>) (wred-inv-prob <1-255>) (cos <0-7>) (dscp <0-64>)**, параметры команды описаны в таблице ниже.

Таблица 102

Параметр	Описание
0-31	Номер очереди
NAME	Имя созданного класса трафика или default (это служебный класс, в который попадает любой трафик, не указанный остальных классах)

Параметр	Описание
CIR	Объем передаваемого за dt трафика, который будет передан гарантированно. Задается одним из следующих способов: в процентах (от 0 до 100); в абсолютных величинах (в Kbps). Для задания значения в абсолютных величинах, после значения параметра должен быть постфикс kbps , например: 500000 kbps ; оставшаяся нераспределенной полоса - remainder . Суммарное значение CIR в очередях одного планировщика не может превышать 100%
PIR	Трафик, превышающий PIR (Peak Information Rate), будет безусловно отброшен. Задается одним из следующих способов: в процентах (от 0 до 100); в абсолютных величинах (в Kbps). Для задания значения в абсолютных величинах, после значения параметра должен быть постфикс kbps , например: 500000 kbps ; оставшаяся нераспределенной полоса - remainder
wred-min	Минимальная граница диапазона, из которого будут отбрасываться случайные пакеты (min/max threshold). Устанавливается в диапазоне от 0 до 100%. Значение wred-min не должно быть больше значения wred-max . Значение по умолчанию - 0
wred-max	Максимальная граница диапазона, из которого будут отбрасываться случайные пакеты (min/max threshold). Устанавливается в диапазоне от 0 до 100%. Значение по умолчанию - 0
wred-inv-prob	Максимальная вероятность того, что пакет будет отброшен. Задается значение знаменателя дроби Probability = 1 / X . Значения устанавливаются в диапазоне (0 - 255). Значение по умолчанию - 10
cos	Перемаркировка поля CoS пакетов при обработке очередей. Допустимые значения от 0 до 7
dscp	Перемаркировка поля DSCP пакетов при обработке очередей. Допустимые значения от 0 до 64

Параметры **wred-min**, **wred-max** и **wred-inv-prob** устанавливают настройки механизма WRED.

В рамках одного планировщика каждый traffic-class может назначаться только одной очереди.

Трафик, который не попал под правила классификатора, попадает в дефолтную очередь - с наименьшим приоритетом. То есть обслуживается только в случае, если остальные очереди полностью реализовали весь трафик в рамках их ограничений.

Пример настройки очередей планировщиков:

```
ecorouter(config)#traffic-scheduler pqwrr.0
ecorouter(config-traffic-scheduler)# queue 2 class IPVOICE cir 60 pir
100 wred-min 45 wred-max 80 wred-inv-prob 100 cos 7 dscp 32
ecorouter(config-traffic-scheduler)# queue 5 class VIDEO cir 80 pir 100
wred-min 40 wred-max 83 wred-inv-prob 250 dscp 40
% Available CIR is 40 percent
ecorouter(config-traffic-scheduler)# queue 5 class VIDEO cir 40 pir 100
wred-min 40 wred-max 83 wred-inv-prob 250 dscp 40
ecorouter(config-traffic-scheduler)# exit
ecorouter(config)#traffic-scheduler pqwrr.1
ecorouter(config-traffic-scheduler)# queue 4 class IPVOICE cir 20000
kbps pir 50000 kbps wred-min 50 wred-max 100
ecorouter(config-traffic-scheduler)# queue 10 class VIDEO cir 100000
kbps pir 500000 kbps wred-min 5 wred-max 20 wred-inv-prob 200
ecorouter(config-traffic-scheduler)# exit
```

23.5 Счетчики

Для просмотра счетчиков QoS используется команда административного режима **show counters port <NAME> queues**.

Внимание: в EcoRouterOS в командах группы **show** при подсчете количества данных не учитываются следующие поля Ethernet-фрейма: Preamble, Frame delimiter, FCS, Interpacket gap (24 байта).

Показания счетчиков группируются по портам и выводятся в виде таблицы, в которой указывается класс трафика, количество пропущенных пакетов/байт и количество отброшенных пакетов/байт в связи с переполнением очереди при использовании алгоритма RED.

Пример:

Таблица 103

Консоль	Комментарий
ecorouter#show counters port te1 queues	Вывести значения счетчиков QoS для порта te1
<pre>Port te0 Service instance te0/eth1 Traffic scheduler pqwrr.0 Early detection algorithm: RED RED-drop QoS Statistics: packets/bytes 0/0 Match WRED-drop Tail-drop Total-drop queue class packets/bytes packets/bytes packets/bytes packets/bytes 0 IP0 27922/42262228 0/0 3776/5716144 3776/5716144 1 IP1 5170/7817860 0/0 1241/1878874 1241/1878874 2 IP2 0/0 0/0 0/0 0/0 3 IP3 0/0 0/0 0/0 0/0 4 --- 0/0 0/0 0/0 0/0 5 --- 0/0 0/0 0/0 0/0 6 --- 0/0 0/0 0/0 0/0 7 default 47/4102 0/0 0/0 0/0</pre>	Вывод команды

Для просмотра счетчиков QoS при использовании алгоритма WRED используется команда административного режима **show counters port <NAME> wred**.

Показывается класс трафика, сконфигурированные параметры, глубину очереди в % от PIR и количество отброшенных пакетов/байт при использовании алгоритма WRED.

Пример:

Таблица 104

Консоль	Комментарий
ecorouter#show counters port te0 wred	Вывести значения счетчиков QoS с учетом WRED для порта te0

Консоль							Комментарий
<pre>Port te0 Service instance te0/eth1 traffic scheduler pqwrr.0 thresholds mark current WRED-drop queue class min max probability load packets/bytes 0 IP0 0 0 1/10 44 0/0 1 --- 0 0 1/0 5 0/0 2 IP1 0 0 1/10 0 0/0 3 --- 0 0 1/0 0 0/0 4 --- 0 0 1/0 0 0/0 5 --- 0 0 1/0 0 0/0 6 --- 0 0 1/0 0 0/0 7 --- 0 0 1/0 0 0/0 ecorouter#</pre>							Вывод команды

Для просмотра счетчиков QoS по количеству ограниченного трафика используется команда административного режима **show counters port <NAME> policer {in | out}**.

Показания счетчиков группируются по портам, выводятся данные по пройденным и отброшенным пакетам/байтам.

Пример:

Таблица 105

Консоль	Комментарий
ecorouter#show counters port te1 policer in	Вывести значения счетчиков ограниченного трафика для порта te1, входящий трафик
<pre>Port te1 Service instance te1.te1/eth2_2 traffic limiter policer.0 MATCHED DROPPED packets/bytes packets/bytes 30129/45596138 3184/4818608 Service instance te1.te1/eth3_3 traffic limiter policer.0 MATCHED DROPPED packets/bytes packets/bytes 30722/46494788 3142/4756164</pre>	Вывод команды

Для сброса счетчиков можно воспользоваться командами **clear**.

```
ecorouter#clear counters port tel ?
policer  policer statistics
queues   QoS queues statistics
red-algorithms  QoS RED/WRED algorithms statistics
```

23.6 Ограничение скорости

Для ограничения скорости/пропускной способности интерфейсов в EcoRouter используются ограничители (полисеры). При помощи полисеров сервисным интерфейсам может быть задано ограничение пропускной способности для того, чтобы сбалансировать распределение нагрузки между несколькими сервисными интерфейсами.

Для создания полисера необходимо создать сервисную политику и указать в ней максимально допустимое значение полосы пропускания. Для создания политики используется команда **service-policy <NAME>**, где <NAME> может быть любым наименованием, рекомендуемый формат имени – заглавные буквы или цифры. Полоса

пропускания задается командой **bandwidth {gbps | mbps | kbps | percent} <VALUE>**, где <VALUE> – значение максимальной скорости в бит/с или в процентах от общей пропускной способности порта. Здесь необходимо указать верхнюю границу выделяемой полосы пропускания. Минимальное значение скорости в килобитах в секунду, которое можно установить, равно 64. Диапазон допустимых значений при указании ограничения в килобитах в секунду - от 64 до 100000000. Создав подобную политику, ее можно применить на нужный экземпляр сервиса (service-instance) в нужном направлении (см. соответствующий раздел руководства).

Пример включения ограничения исходящего трафика:

```
ecorouter(config)#service-policy ECO
ecorouter(config-policy)#bandwidth mbps 10
ecorouter(config)#port ge1
ecorouter(config-port)#service-instance test
ecorouter(config-service-instance)#service-policy ECO out
```

Результат работы ограничителя трафика в EcoRouterOS при приеме данных со скоростью, превышающей установленный лимит, отображает следующий график.

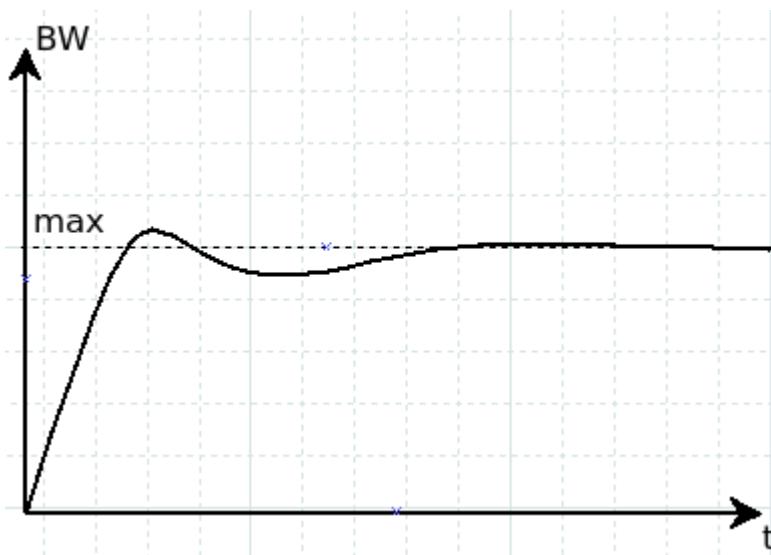


Рисунок 45

Такая обработка трафика производится для предотвращения глобальной TCP-синхронизации при совместной работе ограничителя и алгоритмов раннего обнаружения заполнения очередей в планировщике. Таким образом, пользователям может показаться, что количество трафика превышает установленные лимиты в ограничителе. Для накопления достаточного объема данных и усреднения необходимо довольно продолжительное время (при подаче трафика на одной и той же скорости ≈ 300 сек). Для определения реального объема пропускаемого трафика удобнее воспользоваться командой **show counters port queues-speed**.

23.7 Маркировка трафика

Маркировка трафика настраивается в EcoRouterOS при помощи сущности filter-map (см. раздел "Списки доступа"). Таким образом, к трафику определенного вида применяются

различные действия, в том числе, маркировка. Под маркировкой здесь понимается то, что трафику, попадающему под действие правила, присваивается определенный класс (class-map).

Ниже приведен пример маркировки трафика с созданием двух карт классов с именами L2 и L3, соответствующими уровням фильтрации, которые устанавливают значения поля dscp 30 и 40.

```
ecorouter(config)#class-map L2
ecorouter(config-cmap)#set dscp 30
ecorouter(config)#class-map L3
ecorouter(config-cmap)#set dscp 40
```

Создание карты фильтрации для L3.

```
ecorouter(filter-map-ipv4)#filter-map ipv4 L3 10
```

Добавление правил.

```
ecorouter(filter-map-ipv4)#match icmp host 10.10.10.10 host 192.168.1.10
ecorouter(filter-map-ipv4)#set class-map L3
```

Создаем еще один блок фильтрации для L3.

```
ecorouter(filter-map-ipv4)#filter-map ipv4 L3 20
ecorouter(filter-map-ipv4)#match icmp host 10.10.10.10 host 192.168.1.11
ecorouter(filter-map-ipv4)#set accept
```

Создание карты фильтрации для L2. Здесь aaa.bbb.ccc - MAC-адрес хоста 192.168.1.10.

```
ecorouter(filter-map-ethernet)#filter-map ethernet L2 10
ecorouter(filter-map-ethernet)#match any host aaa.bbb.ccc
```

Назначение действия для L2.

```
ecorouter(filter-map-ethernet)#set class-map L2
ecorouter(filter-map-ethernet)#filter-map ethernet L2 20
ecorouter(filter-map-ethernet)#match any any
ecorouter(filter-map-ethernet)#set accept
```

Назначение filter-map L3 на вход интерфейса.

```
ecorouter(config)#int test
ecorouter(config-if)#set filter-map in L3
```

Назначение filter-map L2 на вход service-instance порта.

```
ecorouter(config)#port tel
ecorouter(config-port)#service-instance test
ecorouter(config-service-instance)#set filter-map in L2
```

При поступлении трафика на сервисный интерфейс есть возможность изменить значение его поля DSCP или сбросить в 0. Для этого используется команда контекстного режима конфигурирования сервисного интерфейса **qos reset dscp <0-63>**). Отменить сброс значения поля DSCP можно при помощи команды контекстного режима конфигурирования сервисного интерфейса **no qos reset dscp <0-63>**). Если новое значение поля не указано, то по умолчанию оно сбрасывается в 0.

```
ecorouter(config)#port tel
ecorouter(config-port)#service-instance 100
ecorouter(config-service-instance)#qos reset dscp 63
```

23.8 Перемаркировка трафика

EcoRouterOS позволяет перемаркировать поля DSCP, CoS, MPLS EXP. В режиме конфигурации карты классов пользователю доступна команда **set**, с помощью которой производится перемаркировка полей в заранее выделенных из общего потока трафика пакетах (правило **match**) путем указания новых значений для полей DSCP, CoS, MPLS EXP в заголовках IP, 802.1Q, MPLS .

Пример:

```
class-map test
match dscp 8
set dscp 18
```

EcoRouterOS позволяет классифицировать трафик по одним полям а маркировать по другим.

Пример:

```
class-map test
match dscp 8
set cos 1
```

EcoRouterOS позволяет перемаркировать несколько полей одновременно. Для перемаркировки нескольких полей необходимо, чтобы сценарий передачи фреймов предусматривал обработку соответствующих заголовков.

Пример:

```
class-map test
match dscp 8
set cos 1
  set exp 2
```

Для применения функционала перемаркировки требуется создать профиль трафика, привязать к нему созданные классы трафика, создать политику и привязать ее к экземпляру сервиса (**service-instance**) в исходящем направлении. Более подробную информацию об этих шагах можно прочитать в разделах посвященных классификации трафика и созданию сервисных политик. Ниже приведен только пример конфигурирования функционала перемаркировки исходящего трафика в EcoRouterOS. Перемаркировка в входящем направлении невозможна.

Пример включения перемаркировки трафика, исходящего из порта ge1:

```
ecorouter(config)#class-map VIDEO
ecorouter(config-cmap)#match dscp 1
ecorouter(config-cmap)#set dscp 11
ecorouter(config-cmap)#exit
ecorouter(config)#class-map IPVOICE
ecorouter(config-cmap)#match dscp 2
ecorouter(config-cmap)#set dscp 12
```

```
ecorouter(config-cmap) #exit
ecorouter(config)#traffic-profile TEST
ecorouter(config-traffic-profile)#class VIDEO
ecorouter(config-traffic-profile)#class IPVOICE
ecorouter(config-cmap) #exit
ecorouter(config)#service-policy ECO
ecorouter(config-policy)#traffic-profile TEST
ecorouter(config)#port ge1
ecorouter(config-port)#service-instance test
ecorouter(config-service-instance)#service-policy ECO out
```

23.9 Сервисные политики

В EcoRouterOS для применения следующего функционала:

- классификации данных (classifier);
- ограничения трафика (limiter);
- управления очередями и алгоритмами раннего обнаружения их заполнения (scheduler)

необходимо настраивать сервисные политики и применять их на экземплярах сервиса (service-instance) в нужном направлении.

Для создания политики используется команда **service-policy <NAME>**, где <NAME> может быть любым наименованием, рекомендуемый формат имени – заглавные буквы или цифры.

После ввода команды следует переход в контекстный режим конфигурирования политики, здесь доступны следующие команды:

```
ecorouter(config)#service-policy ECO
ecorouter(config-policy) #?
Service policy configuration commands:
bandwidth      Bandwidth
exit          Exit from the current mode to the previous mode
help          Description of the interactive help system
no            Negate a command or set its defaults
scheduler     Select a traffic-scheduler to configure
show          Show running system information
traffic-profile Select a traffic-profile to use
```

Для настройки ограничения трафика следует настроить параметр **bandwidth**. Администратор имеет возможность выбрать способ задания максимальной полосы пропускания. Значения можно указывать в Кбит/с, Мбит/с, Гбит/с или в процентах от максимальной скорости работы порта.

```
ecorouter(config-policy) #bandwidth ?
gbps    Bandwidth value in gbps
kbps    Bandwidth value in kbps
mbps    Bandwidth value in mbps
percent Bandwidth value as a percentage
```

Для применения политики на экземпляре сервиса ее требуется указать в нужном service-instance и выбрать соответствующее направление. Команда выглядит следующим образом: **ecorouter(config-service-instance)#service-policy <NAME> {in | out}**, где <NAME> – имя заранее сконфигурированной политики, а ключевые слова **in** и **out** указывают, к трафику какого направления следует применять политику.

От заданного направления зависит в целом работа функционала QoS и ограничителя трафика. Так во входящем направлении работают классификация данных, общее ограничение трафика и ограничение трафика по классам. При настройке политики в исходящем направлении работают общее ограничение трафика, перемаркировка трафика, планировщик очередей, алгоритмы раннего обнаружения заполнения очередей.

Для настройки классификации следует привязать созданный ранее профиль трафика к сервисной политике (service-policy) и применить во входящем направлении. Для работы с планировщиком следует привязать созданный ранее профиль планировщика к сервисной политике (service-policy) и применить в исходящем направлении в нужном экземпляре сервиса (service-instance).

Примеры:

Конфигурация ограничения трафика во входящем направлении:

```
ecorouter(config)#service-policy ECO
ecorouter(config-policy)#bandwidth mbps 10
ecorouter(config)#port ge1
ecorouter(config-port)#service-instance test
ecorouter(config-service-instance)#service-policy ECO in
```

Конфигурация ограничения трафика в исходящем направлении:

```
ecorouter(config)#service-policy ECO
ecorouter(config-policy)#bandwidth mbps 10
ecorouter(config)#port ge1
ecorouter(config-port)#service-instance test
ecorouter(config-service-instance)#service-policy ECO out
```

Конфигурация классификации трафика во входящем направлении:

```
ecorouter(config)#service-policy ECO
ecorouter(config-policy)#traffic-profile TEST
ecorouter(config)#port ge1
ecorouter(config-port)#service-instance test
ecorouter(config-service-instance)#service-policy ECO in
```

Конфигурация ограничения трафика по классам во входящем направлении:

```
ecorouter(config)#service-policy ECO
ecorouter(config-policy)#traffic-profile TEST
ecorouter(config-policy)#bandwidth mbps 10
ecorouter(config)#port ge1
```

```
ecorouter(config-port)#service-instance test
ecorouter(config-service-instance)#service-policy ECO in
```

Конфигурация включения функций планировщика очередей:

```
ecorouter(config)#service-policy ECO_rx
ecorouter(config-policy)#traffic-profile TEST
ecorouter(config)#service-policy ECO_tx
ecorouter(config-policy)#traffic-profile TEST
ecorouter(config-policy)#bandwidth gbps 1
ecorouter(config-policy)#scheduler FAST
ecorouter(config)#port ge1
ecorouter(config-port)#service-instance test1
ecorouter(config-service-instance)#service-policy ECO_rx in
ecorouter(config)#port ge2
ecorouter(config-port)#service-instance test2
ecorouter(config-service-instance)#service-policy ECO_tx out
```

Более подробно конфигурирование вышеуказанного функционала изложено в соответствующих разделах документации.

Для проверки сконфигурированных данных в политике следует воспользоваться командой **show service-policy**.

23.10 Профиль трафика

В EcoRouterOS пользователю доступно составление профилей входящего в маршрутизатор трафика. Посредством созданных профилей и заранее сконфигурированных карт классов (class-map) пользователь может применять к этим профилям различные QoS-политики и функционал ограничения трафика. Профиль создается при помощи команды **traffic-profile <NAME>**, где <NAME> может быть любым, рекомендуемый формат имени – заглавные буквы или цифры.

После создания профиля трафика пользователь производится переход в режим его конфигурирования.

Пример:

```
ecorouter(config)# traffic-profile 1
ecorouter(config-traffic-profile)# ?
Traffic profile configuration commands:
  class  Select a class to configure
  exit   Exit from the current mode to the previous mode
  help   Description of the interactive help system
  no    Negate a command or set its defaults
  show  Show running system information
```

Для привязки классов трафика к профилю используется команда **class** с указанием имени ранее сконфигурированной карты классов.

```
ecorouter(config)#traffic-profile 1
ecorouter(config-profile)#class VIDEO
ecorouter(config-profile)#class IPVOICE
```

В профиле трафика нельзя добавить классы с пересекающимися значениями полей DSCP, CoS, MPLS EXP. В профиле трафика существует еще одно правило. Легче всего пояснить его на конкретном примере. Допустим, на маршрутизатор приходит пакет с тегированным полем MPLS EXP = 1 и DSCP = 3.

При этом профиль трафика и карты классов сконфигурированы следующим образом:

```
ecorouter(config)#class-map A
ecorouter(config-cmap)#match dscp 3
ecorouter(config-cmap)#exit
ecorouter(config)#class-map B
ecorouter(config-cmap)#match cos 1
ecorouter(config-cmap)#exit
ecorouter(config)#traffic-profile C
ecorouter(config-profile)#class A
ecorouter(config-profile)#class B
```

В таком случае при поступлении пакета с MPLS EXP = 1 и DSCP = 3 пакет будет принадлежать классу B, так как заголовок DOT1Q идет перед заголовком IP. Исходя из этого EcoRouterOS сначала проверит поле CoS, затем MPLS и лишь в конце поле DSCP.

Профили трафика применяются абсолютно для всего функционала QoS и требуют применения на конкретной сервисной политике (service-policy). Подробнее данный функционал описан в соответствующем разделе руководства.

23.11 Карты классов

За создание классов трафика и привязку к ним конкретных значений полей DSCP, CoS, MPLS EXP в EcoRouterOS отвечают карты классов (class-map). Подобные карты являются неотъемлемой частью всех функций QoS в маршрутизаторе EcoRouter, поскольку именно они позволяют работать по отдельности с различными типами входящего в маршрутизатор трафика.

Карты настраиваются в конфигурационном режиме. Для создания новой карты требуется ввести команду **class-map <NAME>**, где <NAME> может быть любым, рекомендуемый формат имени – все буквы заглавные. После ввода команды происходит переход в контекстный режим конфигурирования карты классов.

```
ecorouter(config)# class-map VOICE
ecorouter(config-cmap)#
Traffic classifier configuration commands:
 exit  Exit from the current mode to the previous mode
 help  Description of the interactive help system
 match Classification criteria
 no    Negate a command or set its defaults
```

```
set    Set marking values
show   Show running system information
```

Для указания соответствия определенного значения полей DSCP, CoS, MPLS EXP и самой карты, следует воспользоваться командой **match**.

```
ecorouter(config-cmap) #match ?
cos  IEEE 802.1Q class of service priority values
dscp Match DSCP in IP packets
exp  Match MPLS experimental
ecorouter(config-cmap) #match cos ?
<0-7> Enter class-of-service values
ecorouter(config-cmap) #match dscp ?
<0-63> Enter DSCP values
ecorouter(config-cmap) #match exp ?
<0-7> Enter MPLS exp values
```

Пользователю доступно введение в класс несколько команд **match** и определение класса по нескольким полям разного типа. Таким образом, в карте начинает работать логическое правило «ИЛИ». При первом совпадении входящего трафика со значением любого поля, сконфигурированного в классе, трафик будет соответствовать этому классу.

Для установки нового значения в поля DSCP и CoS при выходе трафика из EcoRouter следует воспользоваться командой **set**.

```
ecorouter(config-cmap) #set ?
cos  IEEE 802.1Q class of service priority values
dscp Match DSCP in IP packets
ecorouter(config-cmap) #set cos ?
<0-7> Enter class-of-service values
ecorouter(config-cmap) #set dscp ?
<0-63> Enter DSCP values
```

В командах **match** и **set** значения могут задаваться только в десятичном виде. Можно задавать набор значений, используя в качестве разделителя запятую «,», или диапазон, используя в качестве разделителя дефис «-».

Карты классов позволяют классифицировать трафик, ограничивать его по классам, распределять трафик в разные очереди и применять к ним разные политики обслуживания.

23.12 Ограничение входящего трафика по классам

В EcoRouterOS помимо возможности ограничения трафика на экземплярах сервиса (*service-instance*) в различных направлениях существует возможность ограничивать входящий трафик по классам. Приходящие на маршрутизатор данные необходимо классифицировать, а затем в созданном профиле трафика указать максимально допустимые скорости (PIR) для каждого класса. Скорости можно задавать в бит/с и в процентах от максимально допустимого значения полосы пропускания в ограничителе трафика.

Команда для задания ограничения скорости в профиле трафика:

class <NAME> {kbps | mbps | gbps | percent} <VALUE>, где <NAME> может быть любым наименованием, рекомендуемый формат имени – все заглавные буквы или цифры.

Пример:

```
traffic-profile test
class test10 kbps 500
class test7 mbps 5
class test8 mbps 2
class test9 mbps 2
traffic-profile test2
class A percent 50
class B percent 20
class C percent 20
class D percent 10
```

Внимание: в профиле трафика необходимо придерживаться одного стиля задания скорости, то есть если для первого сконфигурированного класса скорость была указана в процентах, то и последующие ограничения скоростей для классов должны быть указаны в процентах.

Далее необходимо привязать сконфигурированный профиль трафика к сервисной политике (service-policy) и указать максимально допустимую **общую для всех классов** полосу пропускания трафика.

```
service-policy CLIENT_A
traffic-profile test
bandwidth max mbps 100
```

Далее для включения ограничения входящего трафика необходимо в контекстном режиме конфигурирования экземпляра сервиса (service-instance) указать сконфигурированную политику и задать ее во входящем направлении.

```
port te0
service-instance A
  service-policy CLIENT_A in
```

Просмотр данных об ограниченном трафике производится при помощи команды **show counters port <NAME> policer in**.

При необходимости можно данные статистики можно сбросить при помощи команды **clear counters port <NAME> policer in**.

24 Настройки зеркалирования

Зеркалирование – это функция дублирования пакетов одного или нескольких портов (интерфейсов) на другом, также называемая отслеживанием порта или SPAN (Switched Port Analyzer – в терминологии Cisco). В основном она применяется для мониторинга всего трафика в целях безопасности, либо оценки производительности/загрузки сетевого оборудования с применением аппаратных средств.

В концепции EcoRouter данная функция реализована программными средствами, и в качестве SPAN-порта может быть настроен любой физический сетевой интерфейс (port) маршрутизатора.

24.1 Mirror-session

Для настройки функции зеркалирования используются объекты конфигурации типа **mirror-session**, которые располагаются после описания портов. Данный объект конфигурации включает в себя параметры, описанные в таблице ниже.

Таблица 106

Параметр	Описание
mirror-session <название>	Название правила, по которому осуществляется зеркалирование трафика. Название может быть задано только цифрами
description	Описание правила. Необязательный параметр
destination port <название>	Порт, на который отправляется зеркалируемый трафик. Рекомендуется, чтобы к данному порту не был привязан interface и service-instance (подробнее с концепцией port, interface и service-instance можно ознакомиться в разделе Виды интерфейсов)
source <тип> <название> <параметры>	Источник, трафик которого дублируется. В качестве источника может быть указан: port, interface, service-instance. У правила может быть несколько источников трафика, в этом случае они указываются с новой строки. Для удаления одного из источников в конфигурации mirror-session используется команда no source <тип> <название> . Возможность настройки правил зеркалирования одновременно с конфигурированием сервисного интерфейса EcoRouter описана ниже
Параметры source	
<направление>	Определяет, какой именно трафик необходимо дублировать: tx – исходящий, rx – входящий, both – оба направления. Для service-instance возможно зеркалирование только входящего трафика (rx)
<операции над метками>	Необязательный параметр. К зеркалируемому трафику могут быть применены операции над метками. Подробнее о метках можно прочитать в разделе Сервисные интерфейсы
push <метка1> <метка2>	Добавление одной метки или двух. Верхняя метка указывается первой. Доступно для трафика, зеркалируемого с interface и service-instance
pop <количество меток>	Снятие метки или меток. Количество меток может быть 1 или 2. Доступно для трафика, зеркалируемого с service-instance

Параметр	Описание
translate <количество меток>-to-<количество меток> <метка>	Замена одних меток другими. Доступно для трафика, зеркалируемого с service-instance

Для создания правила зеркалирования используется команда: **mirror-session <название>**.

Для удаления правила зеркалирования используется команда: **no mirror-session <название>**.

Источники зеркалирования можно указывать не только при конфигурировании соответствующего правила, но и при конфигурировании самого источника (**port**, **interface**, **service-instance**). Для этого используется команда **add-mirror-session <название> <направление> [операции над метками]**.

Настраиваемая сессия уже должна быть определена. Данная команда не сохраняется в конфигурации, а преобразуется в параметр **source** в разделе конфигурации, относящемся к **mirror-session**.

Пример создания правила для дальнейшей настройки:

```
ecorouter#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#mirror-session 0
ecorouter(config-mirror)#destination port te1
```

Пример настройки правила зеркалирования при конфигурировании **port**:

```
ecorouter(config)#port te2
ecorouter(config-port)#add-mirror-session 0 both
```

Пример настройки правила зеркалирования при конфигурировании **interface**:

```
ecorouter(config)#interface e3
ecorouter(config-if)#add-mirror-session 0 tx push 107
```

Пример настройки правила зеркалирования при конфигурировании **service-instance**:

```
ecorouter(config)#port te3
ecorouter(config-port)#service-instance te3
ecorouter(config-service-instance)#add-mirror-session 0 rx push 100
```

Вывод конфигурации после вышеуказанных настроек правил зеркалирования:

```
!
mirror-session 0
destination port te1
source port te2 both
source interface e3 tx push 107
source port te3 service-instance te3 rx push 100
!
```

Для одного интерфейса (**port**, **interface** или **service-instance**) может быть создано до 8 правил зеркалирования. При этом, правила с зеркалированием трафика в обоих направлениях, считаются двойными. Всего в конфигурации EcoRouter может быть заведено 1024 правила.

24.2 Пример настройки зеркалирования

Рассмотрим пример настройки зеркалирования для маршрутизатора и двух клиентских устройств, сконфигурированных, как представлено на схеме ниже.

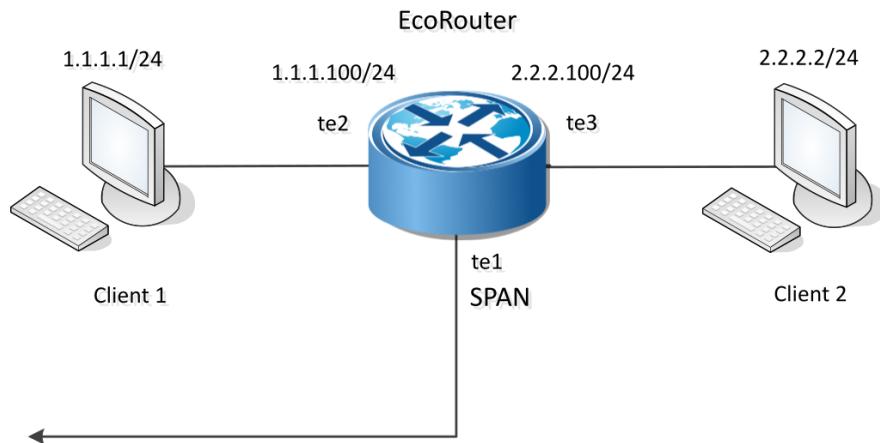


Рисунок 46

В конфигурации EcoRouter настроены следующие соответствия сервисных интерфейсов:

port te2 – service-instance te2 – interface e2,

port te3 – service-instance te3 – interface e3.

Конфигурация EcoRouter:

```
!
interface e2
 ip address 1.1.1.100/24
!
interface e3
 ip address 2.2.2.100/24
!
port tel
!
port te2
 service-instance te2
 encapsulation untagged
 connect ip interface e2
!
port te3
 service-instance te3
 encapsulation untagged
 connect ip interface e3
!
```

Ниже рассмотрено несколько примеров правил зеркалирования. Для того чтобы эти правила не выполнялись все вместе, необходимо либо удалять ненужные правила, либо приостанавливать их, как описано ниже в пункте Приостановка зеркалирования.

24.2.1 Пример правила 1

В конфигурацию EcoRouter вносим правило зеркалирования, при котором весь трафик с **port te2** будет зеркалироваться на **port te1**.

```
ecorouter(config)# mirror-session 0
ecorouter(config-mirror)# destination port tel
ecorouter(config-mirror)# source port te2 both
```

В выводе конфигурации при помощи команды `show run` это правило будет выглядеть следующим образом:

```
!
mirror-session 0
destination port tel
source port te2 both
```

Работу правила **mirror-session 0** можно проиллюстрировать, выполнив с клиентского устройства Client 1 команду **ping 1.1.1.100** и отследив изменение значений счетчиков для **port te2** и **port tel**. Схема зеркалирования, реализуемая правилом **mirror-session 0** представлена ниже.

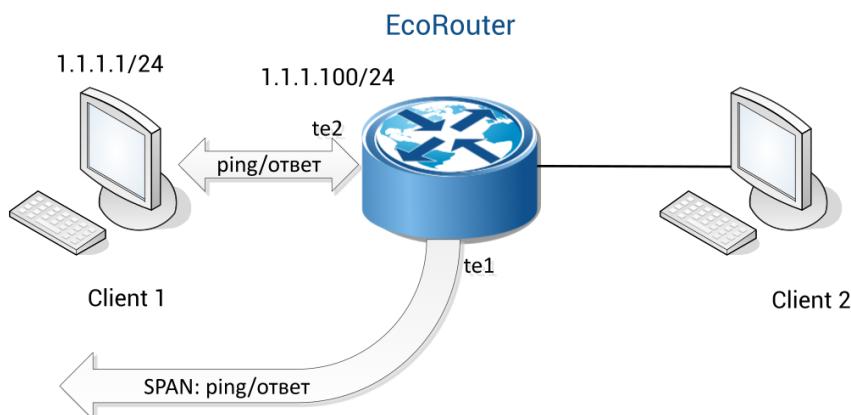


Рисунок 47

При этом, если Client 1 отправил на EcoRouter 10 пингов и получил от него 10 ответов, прирост значений счетчиков будет:

```
port te2
Total received packets: 10
Total transmitted packets: 10
port tel
Total transmitted packets: 20
```

24.2.2 Пример правила 2

В конфигурацию EcoRouter вносим правило зеркалирования, при котором входящий трафик **service-instance te3** зеркалируется на **port te1**.

```
ecorouter(config)# mirror-session 1
ecorouter(config-mirror)# destination port tel
ecorouter(config-mirror)# source port te3 service-instance te3 rx
```

В выводе конфигурации при помощи команды `show run` это правило будет выглядеть следующим образом:

```
!
mirror-session 1
destination port tel
source port te3 service-instance te3 rx
```

Работу правила **mirror-session 1** можно проиллюстрировать, выполнив с клиентского устройства Client 2 команду **ping 2.2.2.100** и отследив изменение значений счетчиков для **port te3** и **port tel**. Схема зеркалирования, реализуемая правилом **mirror-session 1** представлена ниже.

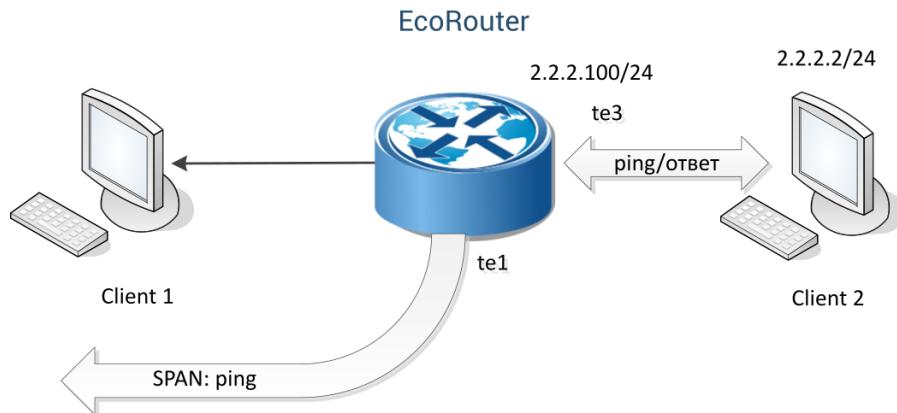


Рисунок 48

При этом, если Client 2 отправил на EcoRouter 10 пингов и получил от него 10 ответов, прирост значений счетчиков будет:

```
port te3
  Total received packets: 10
  Total transmitted packets: 10
port tel
  Total transmitted packets: 10
```

24.2.3 Пример правила 3

В конфигурацию EcoRouter вносим правило зеркалирования, при котором исходящий трафик **interface e3** зеркалируется на **port te1**.

```
ecorouter(config)# mirror-session 2
ecorouter(config-mirror)# destination port tel
ecorouter(config-mirror)# source interface e3 tx
```

В выводе конфигурации при помощи команды **show run** это правило будет выглядеть следующим образом:

```
!
mirror-session 2
destination port tel
source interface e3 tx
```

Работу правила **mirror-session 2** можно проиллюстрировать, выполнив с клиентского устройства Client 2 команду **ping 2.2.2.100** и отследив изменение значений счетчиков для **port te3** и **port tel**. Схема зеркалирования, реализуемая правилом **mirror-session 2** представлена ниже.

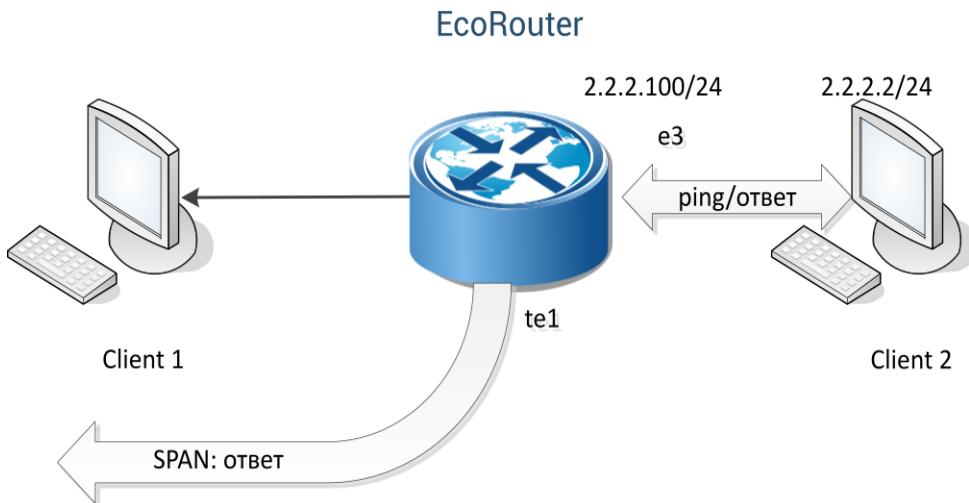


Рисунок 49

При этом, если Client 2 отправил на EcoRouter 10 пингов и получил от него 10 ответов, прирост значений счетчиков будет:

```
interface e3
  Total received packets: 10
  Total transmitted packets: 10
port te1
  Total transmitted packets: 10
```

24.3 Приостановка зеркалирования

Для того чтобы приостановить действие правила, используется параметр **shutdown**. Пример ввода параметра:

```
ecorouter#conf t
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#mirror-session 3
ecorouter(config-mirror)#shutdown
```

Возобновление действия правила осуществляется удалением параметра **shutdown** при помощи команды **no shutdown**.

```
ecorouter(config)#mirror-session 3
ecorouter(config-mirror)#no shutdown
```

24.4 Просмотр правил зеркалирования

Список существующих правил зеркалирования и их состояния выводится по команде **show mirror-session rules**. Данная команда действует в конфигурационном режиме консоли.

Пример вывода команды:

```
ecorouter#show mirror-session rules
Mirror session 0 is up
  10001.rx: rx port te2 -> port te1
  10001.tx: tx port te2 -> port te1
Mirror session 1 is administratively down
  10031.rx: rx service instance te3/te3 -> port te1
```

```
Mirror session 2 is administratively down  
 6.tx: tx interface e3 -> port tel
```

Для просмотра настроек правил зеркалирования и статистики по ним используется команда **show mirror-session [<название>]**. В случае, если не указано название правила, команда выводит для просмотра информацию по всем существующим правилам. Данная команда действует в конфигурационном режиме консоли.

Пример вывода команды:

```
ecorouter#show mirror-session  
Mirror session 0 is up  
  Destination: port tel  
    port te2 both  
      rx packets 0, bytes 0  
      tx packets 17, bytes 1022  
Mirror session 1 is up  
  Destination: port tel  
    service instance te3/3 rx  
      rx packets 7, bytes 570  
Mirror session 2 is up  
  Destination: port tel  
    interface e3 tx  
      tx packets 0, bytes 0
```

Для сброса значений счетчиков правил зеркалирования используется команда **clear counters mirror-session [<название>]**. В случае, если не указано название правила, счетчики будут обнулены для всех правил. Данная команда действует в конфигурационном режиме консоли.

25 Встроенный NAT

NAT (Network Address Translation) - это механизм, позволяющий маршрутизатору осуществлять трансляцию (подмену) сетевых адресов для транзитного трафика. Наряду с адресами отправителя/получателя могут также подменяться номера TCP или UDP-портов отправителя/получателя. Технология NAT чаще всего используется для предоставления одного публичного IP-адреса множеству локальных пользователей с приватными адресами. А также для обеспечения доступа из LAN в WAN, то есть обеспечения возможности устройствам с приватными адресами отсылать/получать данные из глобальной сети (от устройств с публичными адресами). При использовании NAT топология внутренней сети скрыта и доступ из внешней сети может быть ограничен.

Существует два вида NAT:

- source NAT (SNAT),
- destination NAT (DNAT),

и три основных концепции трансляции адресов (в рамках функционала EcoRouter):

- static NAT,
- dynamic NAT,
- NAT with overload (PAT).

Source NAT - это наиболее распространенный тип NAT, суть механизма работы которого состоит в подмене IP-адреса отправителя (источника) на пути пакета из внутренней сети во внешнюю и обратной подмене адреса получателя на пути пакета из внешней сети во внутреннюю. Частый сценарий применения: обеспечение доступа из LAN в WAN.

Destination NAT - тип NAT, суть механизма работы которого состоит в подмене IP-адреса получателя (назначения) на пути пакета из внешней сети во внутреннюю и обратной подмене адреса отправителя на пути пакета из внутренней сети во внешнюю. Частый сценарий применения: обеспечение доступа извне к каким-либо сервисам, предоставляемым серверами, находящимися в LAN-сети.

Static NAT - статическая трансляция один-в-один - подмена одного заранее определенного IP-адреса на другой, также заранее определенный. Правило о такой подмене хранится в таблице трансляций неограниченное количество времени или до тех пор, пока сохраняется соответствующая конфигурация маршрутизатора.

Dynamic NAT - неоднозначная трансляция один-в-один, то есть подмена одного из заранее определенных IP-адресов на первый свободный из обозначенного диапазона (пула). Правило о такой подмене хранится в таблице трансляций до тех пор, пока внутренний и внешний хосты продолжают обмен данными. Если в течение некоторого установленного времени трафик по этой трансляции отсутствует, правило удаляется и адрес освобождается, то есть возвращается в пул.

NAT with overload (PAT) - трансляция много-в-один, то есть подмена нескольких заранее определенных внутренних адресов на один и тот же внешний. Правило о такой

подмене кроме самих адресов содержит TCP/UDP порт источника, который используется для идентификации трафика на принадлежность тому или иному внутреннему хосту.

Описание команд для настройки NAT на EcoRouter представлено в таблице ниже.

Таблица 107

Команда	Описание
ip nat inside	Команда вводится в конфигурационном режиме (config-if). В результате выполнения этой команды интерфейс помечается как "внутренний интерфейс NAT", это означает, что весь трафик, вошедший на этот интерфейс помечается как "возможно транслируемый"
ip nat outside	Команда вводится в конфигурационном режиме (config-if). В результате выполнения этой команды интерфейс помечается как "внешний интерфейс NAT", это означает, что весь трафик, предназначенный для выхода через этот интерфейс и помеченный как "возможно транслируемый" будет подвергаться трансляции
ip nat source static A.B.C.D Q.W.E.R [vrf]	Команда вводится в конфигурационном режиме (config). В результате выполнения этой команды будет создана статическая трансляция адрес-в-адрес. Параметр vrf является необязательным и без указания определенного vrf правило будет создано для default vrf
ip nat source static network A.B.C.D Q.W.E.R mask [vrf]	Команда вводится в конфигурационном режиме (config). В результате выполнения этой команды будет создано сразу несколько статических трансляций адрес-в-адрес для двух равных диапазонов адресов. Количество трансляций определяется параметром mask (маска подсети). Параметр vrf является необязательным и без указания определенного vrf правила будет создано для default vrf
ip nat source static A.B.C.D interface <IF_NAME> [vrf]	Команда вводится в конфигурационном режиме (config). В результате выполнения этой команды будет создана статическая трансляция адрес-в-адрес. В качестве inside global адреса для трансляции будет взят адрес, назначенный на указанный в команде интерфейс. Параметр vrf является необязательным и без указания определенного vrf правила будет создано для default vrf
ip nat pool <POOL_NAME> <RANGE>	Команда вводится в конфигурационном режиме (config). В результате выполнения этой команды будет создан пул адресов, который можно будет использовать для задания правил динамических трансляций. Диапазон адресов можно задавать через дефис и через запятую: 1.1.1.1-1.1.1.10,2.2.2.2,3.3.3.5-3.3.4.5
ip nat source dynamic inside pool <POOL_NAME> overload A.B.C.D [vrf]	Команда вводится в конфигурационном режиме (config). В результате выполнения этой команды начнут создаваться динамические трансляции много-в-один для пакетов из LAN, source IP которых будет попадать в диапазон адресов, определяемый пулом. Время жизни трансляции после прохождения последнего пакета - 300 секунд. В качестве inside global адреса для трансляции будет взят адрес, указанный после ключевого слова overload . Параметр vrf является необязательным и без указания определенного vrf правила будет создано для default vrf
ip nat source dynamic inside pool <POOL_NAME> overload interface <IF_NAME> [vrf]	Команда вводится в конфигурационном режиме (config). В результате выполнения этой команды начнут создаваться динамические трансляции много-в-один для пакетов из LAN, source IP которых будет попадать в диапазон адресов, определяемый пулом. Время жизни трансляции после прохождения последнего пакета - 300 секунд. В качестве inside global адреса для трансляции будет взят адрес, назначенный на указанный в команде интерфейс. Параметр vrf является необязательным и без указания определенного vrf правила будет создано для default vrf

Посмотреть состояние таблицы трансляций на EcoRouter можно при помощи команды
show ip nat translations:

```
ecorouter#show ip nat translations
Static translations:
Source      Translated      VRF
3.3.3.3      4.4.4.4        default
PAT translations:
  Source      Translated      Destination      IF
Time: 5s, Protocol: ICMP, VRF: default
IN: 10.10.10.10      20.20.20.21      20.20.20.20      N/A
OUT: 20.20.20.20     20.20.20.21      20.20.20.21      N/A
Time: 3s, Protocol: TCP, VRF: default
IN: 10.10.10.10:171  20.20.20.21:35005  20.20.20.20:35091  N/A
OUT: 20.20.20.20:35091  20.20.20.21:35005  20.20.20.21:35005  N/A
```

25.1 Пример конфигурации static source NAT

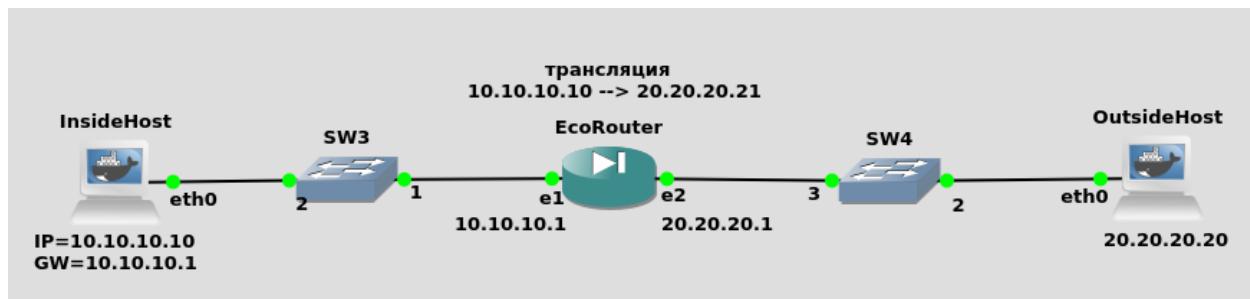


Рисунок 50

Конфигурация EcoRouter:

Настройка интерфейсов и портов:

```
ecorouter(config)#port te0
ecorouter(config-port)#service-instance si0
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config)#port tel
ecorouter(config-port)#service-instance sil
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config)#interface in
ecorouter(config-if)#ip address 10.10.10.1/24
ecorouter(config-if)#ip nat inside
ecorouter(config-if)#connect port te0 service-instance si0
ecorouter(config)#interface out
ecorouter(config-if)#ip address 20.20.20.1/24
ecorouter(config-if)#ip nat outside
ecorouter(config-if)#connect port tel service-instance sil
```

Задание статической трансляции:

```
ecorouter(config)#ip nat source static 10.10.10.10 20.20.20.21
```

25.2 Пример конфигурации static source NAT

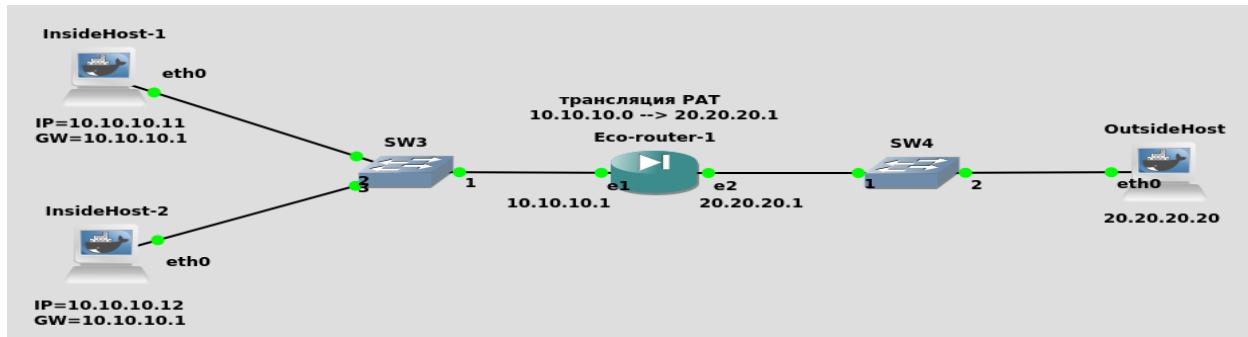


Рисунок 51

Конфигурация EcoRouter:

Настройка интерфейсов и портов:

```
ecorouter(config)#port te0
ecorouter(config-port)#service-instance si0
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config)#port tel
ecorouter(config-port)#service-instance si1
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config)#interface in
ecorouter(config-if)#ip address 10.10.10.1/24
ecorouter(config-if)#ip nat inside
ecorouter(config-if)#connect port te0 service-instance si0
ecorouter(config)#interface out
ecorouter(config-if)#ip address 20.20.20.1/24
ecorouter(config-if)#ip nat outside
ecorouter(config-if)#connect port tel service-instance si1
```

Создание пула адресов для входящего трафика:

```
ecorouter(config)#ip nat pool POOL 10.10.10.0-10.10.10.20
```

Задание правила трансляций:

```
ecorouter(config)#ip nat source dynamic inside pool POOL overload
20.20.20.21
```

26 Операции с прошивкой

В EcoRouter есть несколько видов встроенного программного обеспечения (прошивки).

Factory - заводская версия программного обеспечения, не **подлежит** изменению. Factory представляет собой базовую версию с урезанным функционалом.

Для полноценной работы устройства необходима установка второго уровня программного обеспечения - image. Базовая версия image-прошивки поставляется предустановленной на маршрутизатор.

На одном устройстве одновременно может быть установлена factory прошивка и не более двух image-прошивок.

Для просмотра информации о доступных на устройстве прошивках используется команда административного режима **show boot**. Данная команда выводит информацию о том, с какой прошивки был произведен запуск, состояние каждой прошивки и стабильность.

```
ecorouter# show boot
F: vX.X.X, not loaded, active, stable
A: vX.X.X, not loaded, inactive, stable
B: vX.X.X, loaded, active, unstable
```

Здесь F - factory-прошивка, A и B - image-прошивки.

Первый столбец показывает, с какой прошивки произведена загрузка, второй столбец показывает, активна ли данная прошивка в случае перезагрузки, является временной для загрузки или признана неисправной (active/inactive/temporary/failed), а третий - ее стабильность.

26.1 Скачивание образа прошивки

Для обновления image-прошивки предусмотрена возможность скачивания ее с FTP или TFTP-сервера. Команды для скачивания описаны в таблице ниже.

Таблица 108

Команда	Описание
copy ftp image ftp://user:password@xxx.xxx.xxx.xxx/ mgmt	С FTP-сервера будет скачан подходящий образ прошивки для обновления с текущей версии прошивки, FTP-сервер доступен через менеджмент-порт (mgmt). EcoRouter сам определит, какой файл на сервере подходит для скачивания и обновления.
copy ftp image ftp://user:password@xxx.xxx.xxx.xxx/filename vr default	С FTP-сервера будет скачан указанный файл, если он подходит для текущей платформы и возможно обновление до этой версии. Доступ к FTP-серверу осуществляется через интерфейс виртуального маршрутизатора, выбранного по умолчанию.
copy tftp image tftp://xxx.xxx.xxx.xxx/ vr vrname	С TFTP-сервера будет скачан подходящий образ прошивки для обновления с текущей версии прошивки. EcoRouter сам

Команда	Описание
	определит, какой файл на сервере подходит для скачивания и обновления. Доступ к TFTP-серверу осуществляется через интерфейс виртуального маршрутизатора с именем vrname.
copy tftp image tftp://xxx.xxx.xxx.xxx/filename mgmt	С TFTP-сервера будет скачан указанный файл, если он подходит для текущей платформы и возможно обновление до этой версии; доступ к TFTP-серверу осуществляется через менеджмент-порт (mgmt).

В общем виде команда для скачивания образа прошивки маршрутизатора выглядит следующим образом: **copy <ftp | tftp> image <URL> <mgmt | vr default | vr <VR_NAME>>**. Обязательно указание интерфейса, через который осуществляется доступ к ftp или tftp.

ВНИМАНИЕ! Во время скачивания образа CLI не будет реагировать на другие команды.

Скачивание прошивки с меньшим номером версии, чем нынешняя (downgrade), невозможно.

После скачивания на устройство непосредственно перед попыткой установки образ проходит проверку целостности. Также проверка целостности производится в процессе выполнения команды **show**.

Для просмотра информации о скачанных образах и их состоянии используется команда административного режима **show images storage** (для просмотра образов, размещенных на внутреннем накопителе устройства) или **show images usb** (для просмотра образов, размещенных на подключенных USB-устройствах). Если установлена только factory-прошивка, вывод команды будет пустым.

```
ecorouter# show images
"EcoRouterOS-ER-1004-3.2.1.0.8942-release-20f197c.image": version
v3.2.1.0.8942, verification is ok, is not suitable for installation.
Version dependency check failed
"EcoRouterOS-ER-1004-3.2.1.0.8949-release-20f197c.image": version
v3.2.1.0.8949, verification is ok, is not suitable for installation.
Version dependency check failed
"EcoRouterOS-ER-116-3.2.1.0.8942-release-20f197c.image": version
v3.2.1.0.8942, verification is ok, is not suitable for installation.
EcoRouterOS-ER-116-3.2.1.0.8942-release-20f197c.image is not for
platform ER-1004
Available free space on device (27.72GiB) is 23.80GiB.
```

Здесь:

verification is ok - образ успешно прошел проверку целостности,

verification is failed - образ не прошел проверку целостности.

Соответственно, образы могут подходить для установки (suitable for installation) или не подходить (not suitable for installation) по разным причинам. В приведенном примере первый и второй образы не прошли проверку на зависимость версий, а третий несовместим с платформой ER-1004.

26.2 Установка скачанного образа прошивки

Для установки образа используется команда **image install [storage] <IMAGE_NAME> [force]**, где **IMAGE_NAME** - один из образов, указанных в выводе команды **show images storage**. По умолчанию установка производится с внутреннего накопителя маршрутизатора. Указание параметра **force** позволяет установить прошивку с меньшим номером версии, чем установленная (downgrade), работоспособность маршрутизатора при этом не гарантируется.

Возможен вариант установки заранее скачанного образа с USB-устройства, для этого используется команда **image install usb <IMAGE_NAME>**, где **IMAGE_NAME** указывается полностью, например, **EcoRouterOS-ER-1004-L-3.2.0.0.8167-develop-7bf31860.image**.

После завершения инсталляции в выводе команды **show boot** появится установленная версия со статусами **not loaded**, **temporary**, **unstable**. Для загрузки с проинсталлированного image необходимо перезагрузить устройство.

Во время загрузки будет предпринято максимум три попытки запуститься с проинсталлированной image прошивки. При успешной загрузке с новым image его статус изменится на **active**. При неуспешной загрузке статус с **temporary** будет изменен на **failed**. Порядок выбора прошивки для загрузки описан ниже.

Ниже представлены примеры вывода команды **show boot** на разных стадиях обновления прошивки.

Установлена только прошивка А, которая загружена в данный момент и является основной прошивкой для данного устройства.

```
F: vx.x.x, not loaded, inactive, stable  
A: vx.x.x, loaded, active, stable  
B: not installed
```

Загружена прошивка А, только что была установлена прошивка В, которая установлена для тестовой загрузки после перезагрузки.

```
F: vx.x.x, not loaded, inactive, stable  
A: vx.x.x, loaded, active, stable  
B: vx.x.x, not loaded, temporary, unstable
```

Если при загрузке с прошивки, отмеченной как **temporary**, произошла перезагрузка маршрутизатора по любой причине, то статус прошивки будет изменен на **failed**. Если в течение 8 часов при загрузке с прошивки со статусом **active** произойдет 3 неуспешных перезапуска, то статус такой прошивки также будет изменен на **failed**.

Устройство успешно загрузилось с установленной прошивки В, которая была отмечена для временной загрузки.

```
F: vx.x.x, not loaded, inactive, stable  
A: vx.x.x, not loaded, active, stable  
B: vx.x.x, loaded, active, unstable
```

Если установленная прошивка показывает себя стабильной в работе, то её можно отметить как стабильную следующей командой административного режима **boot b-image stable** или **boot a-image stable**, в зависимости от того, какую прошивку необходимо отметить. Для того чтобы пометить прошивку как нестабильную, необходимо выполнить команду **no boot b-image stable** или **no boot a-image stable**. Прошивка factory всегда является стабильной.

Чтобы исключить или включить загрузку с прошивки А или В в случае перезагрузки, можно изменить статус активности командой административного режима **boot a-image active** или **no boot b-image active**.

Приоритет выбора прошивки для загрузки

При загрузке соблюдается следующий порядок выбора прошивки по убыванию приоритетов:

1. Незаводская прошивка со статусом temporary.
2. Незаводская прошивка со статусом active.
3. Незаводская прошивка со статусом stable.

Factory-прошивка.

26.3 Действия после установки образа прошивки

После установки новой версии прошивки и перезагрузки устройства рекомендуется выполнить команду **show running-config diff** для отображения загруженных команд из **startup** конфигурации. Данная команда используется для отображения различий между **startup** и **running** конфигурациями. Для корректной работы этой команды в системе должна быть создана **startup** конфигурация (для ее создания достаточно один раз выполнить команду **write memory** или **copy running-config startup-config**). Выполнение команды **show running-config diff** допускается в виртуальных маршрутизаторах VR.

Таблица 109

Значение	Описание
— <i>line1, line2</i>	Диапазон номеров строк, где произошли изменения (— для running конфигурации, *** для startup конфигурации).
**** <i>line1, line2</i>	
+ <i>text</i>	Команда присутствует в running конфигурации, отсутствует в startup конфигурации.
- <i>text</i>	Команда присутствует в startup конфигурации, отсутствует в running конфигурации.
! <i>text</i>	Команды присутствуют и в startup конфигурации и в running конфигурации, но нарушен порядок следования команд.

Пример:

```
ecorouter#conf terminal
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#interface test
ecorouter(config-if)#ip address 10.0.0.1/24
ecorouter(config-if)#exit
ecorouter(config)#exit
ecorouter#sh running-config diff
*** Startup-config
--- Running-config
*****
*** 48,53 ****
--- 48,57 ----
 port te2
```

```

mtu 9728
!
+ interface test
+ ip mtu 1500
+ ip address 10.0.0.1/24
+ !
arp request-interval 1
arp request-number 3
arp expiration-period 5
ecorouter#

```

26.4 Удаление образа прошивки

Для того чтобы удалить файл image прошивки, который больше не будет использоваться, существует команда **image delete storage <IMAGE_NAME>**, где **IMAGE_NAME** - один из образов, указанных в выводе команды **show images storage**.

Для удаления установленной прошивки существуют команды **image delete firmware a-image** и **image delete firmware b-image**. Удаление прошивки factory невозможно. Удаление прошивки возможно только в случае одновременного выполнения трех условий: она отмечена как неактивная, нестабильная и с неё не произведена загрузка в данный момент.

26.5 Выгрузка образа прошивки

При необходимости, образ прошивки устройства можно скопировать (выгрузить) на внешний FTP/TFTP-сервер.

В общем виде команда для выгрузки образа прошивки маршрутизатора выглядит следующим образом: **copy image <ftp | tftp> <IMAGE_NAME> <URL> <mgmt | vr default | vr <VR_NAME> >**. Здесь: **URL** - адрес сервера, на который будет осуществляться выгрузка, **IMAGE_NAME** - имя образа, должно соответствовать одному из указанных в выводе команды **show images storage**. При вводе команды **copy image** обязательно указание интерфейса, через который осуществляется доступ к ftp или tftp.

ВНИМАНИЕ! Во время выгрузки образа CLI не будет реагировать на другие команды.

26.6 Проверка целостности системных файлов

Для проверки целостности системных файлов используется команда режима администрирования **show hw integrity**.

Данная команда проверяет соответствие контрольных сумм бинарных файлов активной прошивки эталонным значениям. По итогам проверки на консоль выводятся контрольные суммы, имена файлов и результат проверки соответствия (**OK** или **FAIL**). После списка файлов выводится итоговая строка проверки соответствия: **Checksum validation PASSED** или **Checksum validation FAILED**.

Пример.

```

ecorouter#show hw integrity
7dd6d620d71ad0722571951a05812b78 rmt: OK

```

```
aa473b734e46f8479a0ec5feecfdad65 chacl: OK
96b48926e25f3854738f763dbb3ccb50 getfacl: OK
14aabeeeab6ffc8fd8503d0f587c80ff setfacl: OK
...
5f589159b5d17849bfa0c3840a4a4c4c sshd-keygen-start: OK
771e77b5d1ffb9db37b958d2ae2faab libpcre.so.1.2.7: OK
a6aa50ed7b77fc1fd06d8626d8b7d78c libpcre.la: OK
b9fd49b80acaf6173a22b7d5bb6b4f1c libpcreposix.so.0.0.4: OK
60f530c64889d00ad21dd15534e11dea libpcreposix.la: OK
b9f29f6dedee7bfdcc52d9cd3386e51e er-ripd-ns@-start: OK
Checksum validation PASSED
ecorouter#
```

26.7 Сброс до factory

В EcoRouter существует механизм сброса встроенного программного обеспечения до заводской версии (factory).

ВНИМАНИЕ! При этом удаляются все версии image-прошивок и конфигурационные файлы.

Для сброса на factory устройство необходимо перезагрузить или выключить и включить.

Во время загрузки устройства на экран выводится:

```
Stage: boot
starting version NNN
```

Где NNN - некое число, которое может быть разным в разных версиях EcoRouter.

В этот момент необходимо нажать клавишу **[F8]**.

На экране появится строка:

```
^[[19~^[[19~^[[19~^[[19~
```

После чего можно отпустить клавишу **[F8]**. На экране появится сообщение и символ строки ввода.

```
To restore the router's factory settings enter "YES".
!ATTENTION!
This action will erase all configuration!
>
```

Для сброса на factory необходимо ввести заглавными буквами **YES**, при вводе любого другого набора символов механизм сброса не будет запущен.

После подтверждения будет запущен механизм сброса на заводскую прошивку с минимальной стартовой конфигурацией.

26.8 "Мягкий" сброс

Команда **copy empty-config startup-config** позволяет произвести "мягкий" сброс конфигурации, в результате которого будут удалены все записи о пользователях и конфигурация будет возвращена к заводским настройкам. При этом записи о пользователях

удаляются непосредственно после выполнения команды, а возврат конфигурации маршрутизатора к заводской - после перезагрузки устройства.

```
#copy empty-config startup-config
```

При попытке ввода любой команды появится сообщение:

```
ecorouter#conf t  
% User is logged out by timeout
```

После выполнения команды из конфигурации будут удалены все сведения о пользователях. Пользовательская сессия завершена, авторизация на маршрутизаторе возможна только от имени пользователя по умолчанию - **admin**, пароль - **admin**.

```
<<< EcoRouter 3.2.2.0.9678-develop-eb0cf38 (x86_64) - ttys0 >>>  
ecorouter login:
```

Для замены записанной на маршрутизаторе конфигурации на заводскую следует выполнить команду **reload**.

27 Экспорт и импорт конфигурации

Для импорта и экспорта конфигурации EcoRouter используется команда **copy** в административном режиме.

В общем виде логика команды может быть представлена следующим образом:

```
copy <откуда> <куда> <что> <через_интерфейс>
```

Ниже более подробно описан синтаксис каждого из элементов команды.

27.1 Подключение к серверу

EcoRouter может экспортировать / импортировать архив с конфигурационными файлами на / с FTP или TFTP сервера.

Для подключения к FTP серверу указываются следующие параметры: имя пользователя, пароль и IP-адрес FTP сервера.

Для подключения к TFTP сервера указывается только его IP-адрес.

27.2 Путь копирования

После задания IP-адреса сервера можно также задать путь к директории, в которой будет храниться файл архива, и имя этого файла (имена файлов конфигурации, выдаваемые по умолчанию, описаны в параграфе "Архив конфигурации").

Например, если идет копирование на TFTP сервер с IP-адресом 192.168.10.10, можно задать путь копирования одним из способов, описанных в таблице ниже.

Таблица 110

Вариант записи пути	Расположение файла	Имя файла
tftp://192.168.10.10/	корневая директория сервера	по умолчанию
tftp://192.168.10.10/folder/	определенная директория	по умолчанию
tftp://192.168.10.10/name	корневая директория сервера	указанное имя файла, расширение по умолчанию
tftp://192.168.10.10/folder/name	определенная директория	указанное имя файла, расширение по умолчанию
tftp://192.168.10.10/folder/name.res	определенная директория	указанное имя файла, указанное расширение

Приведенный пример демонстрирует гибкость задания пути при копировании архива конфигурации.

27.3 Архив конфигурации

При экспорте конфигурации по умолчанию создается архив с названием следующего вида: **startup_backup_имяхоста_ГГГГММДДЧЧММСС.tar.gz**, например, **startup_backup_EcoRouterOS_20160623175405.tar.gz**.

Внутри этого архива будут располагаться два файла:

- crc – файл с контрольной суммой архива startup_backup.tar,
- startup_backup.tar – архив с конфигурацией.

В свою очередь, внутри архива startup_backup.tar будут:

- configuration.json – конфигурационный файл модуля,
- EcoRouterOS.conf – конфигурационный файл с настройками EcoRouter,
- vrN – папки с конфигурационными файлами настроек виртуальных маршрутизаторов,
- aaa.db.bak – файл базы данных AAA.

27.4 Выбор интерфейса

По умолчанию импорт и экспорт осуществляются через Management-порт (с маркировкой MNG/E0).

При необходимости можно настроить отправку и получение через виртуальный маршрутизатор, используемый по умолчанию, или через любой другой виртуальный маршрутизатор. Для этого используется параметр команды **copy**:

```
vr <default|NAME>
```

27.5 Экспорт конфигурации

В случае экспорта конфигурации происходит копирование из startup-config на FTP или TFTP сервер. При этом копируется последняя сохраненная версия конфигурации (при помощи команды **write**). Если какие-либо изменения были внесены после сохранения конфигурации, они не попадут в экспортируемый файл.

Синтаксис команды экспорта:

```
copy startup-config ftp|tftp <ADDRESS>/<PATH>/< |NAME.RES> vr
<default|NAME>
```

Ниже представлены примеры команд экспорта конфигурации.

Таблица 111

Команда	Описание
FTP	
copy startup-config ftp ftp://user:password@192.168.10.10/	Экспорт на указанный FTP сервер, параметры по умолчанию
copy startup-config ftp ftp://user:password@192.168.10.10/my_name_of_archive	Экспорт на указанный FTP сервер, имя архива задано
copy startup-config ftp ftp://user:password@192.168.10.10/my_name_of_archive.res	Экспорт на указанный FTP сервер, имя и расширение архива задано
copy startup-config ftp ftp://user:password@192.168.10.10/ vr default	Экспорт на указанный FTP сервер через виртуальный маршрутизатор по умолчанию
copy startup-config ftp ftp://user:password@192.168.10.10/ vr VR1	Экспорт на указанный FTP сервер через заданный виртуальный маршрутизатор
TFTP	
copy startup-config tftp tftp://192.168.10.10/	Экспорт на указанный TFTP сервер, параметры по умолчанию

Команда	Описание
copy startup-config tftp tftp://192.168.10.10/my_name_of_archive	Экспорт на указанный TFTP сервер, имя архива задано
copy startup-config tftp tftp://192.168.10.10/my_name_of_archive.res	Экспорт на указанный TFTP сервер, имя и расширение архива задано
copy startup-config tftp tftp://192.168.10.10/ vr default	Экспорт на указанный TFTP сервер через виртуальный маршрутизатор по умолчанию
copy startup-config tftp tftp://192.168.10.10/ vr VR1	Экспорт на указанный TFTP сервер через заданный виртуальный маршрутизатор

27.6 Импорт конфигурации

В случае импорта конфигурации происходит копирование архива с FTP или TFTP сервера на EcoRouter и распаковка полученного архива в startup-config. При этом происходит архивирование последней сохраненной конфигурации. В случае если загружаемый с сервера файл поврежден или по каким-либо другим причинам не может быть установлен в качестве конфигурационного файла, система автоматически восстановит последнюю сохраненную конфигурацию и сообщит об ошибке.

После импорта конфигурации необходимо перезагрузить EcoRouter, чтобы изменения вступили в действие.

Синтаксис команды импорта:

```
copy ftp|tftp startup-config <ADDRESS>/<PATH>/<NAME> vr <default|NAME>
```

Для импорта необходимо указывать имя файла архива.

Ниже представлены примеры команд импорта конфигурации.

Таблица 112

Команда	Описание
FTP	
copy ftp startup-config ftp://user:password@192.168.10.10/startup_backup_EcoRouterOS_20160623175405.tar.gz	Импорт с указанного FTP сервера, параметры по умолчанию
copy ftp startup-config ftp://user:password@192.168.10.10/my_name_backup vr default	Импорт с указанного FTP сервера через виртуальный маршрутизатор по умолчанию
copy ftp startup-config ftp://user:password@192.168.10.10/my_name_backup vr VR1	Импорт с указанного FTP сервера через заданный виртуальный маршрутизатор
copy ftp startup-config ftp://user:password@192.168.10.10/my_name_backup mgmt	Импорт с указанного FTP сервера через management-интерфейс
TFTP	
copy tftp startup-config tftp://192.168.10.10/my_name_backup	Импорт с указанного TFTP сервера, параметры по умолчанию
copy tftp startup-config tftp://192.168.10.10/my_name_backup vr default	Импорт с указанного TFTP сервера через виртуальный маршрутизатор по умолчанию

Команда	Описание
copy tftp startup-config tftp://192.168.10.10/ startup_backup_EcoRouterOS_20160623175405.tar.gz vr VR1	Импорт с указанного TFTP сервера через заданный виртуальный маршрутизатор
copy tftp startup-config tftp://192.168.10.10/my_name_backup mgmt	Импорт с указанного TFTP сервера через management- интерфейс

28 NTP

NTP (network time protocol) – протокол синхронизации времени в сети.

NTP синхронизирует время на устройствах сети относительно UTC (Coordinated Universal Time). Это используется для настройки сервисов безопасности и логирования. NTP использует иерархическую уровневую систему источников времени. Каждый уровень системы называется «Стратум» и имеет определенный номер. Нумерация начинается с нуля с верхнего уровня. Стратум 0 определяет систему, непосредственно в которой находится источник точного времени. Система, подключенная к стратуму 0, начинает относиться к стратуму 1 и так далее. Номер уровня определяет удаленность от первоисточника времени.

Протокол работает на основе протокола UDP, используя 123 порт.

Синхронизация с заданным NTP сервером происходит каждые 15 минут.

Команды настройки NTP представлены в таблице ниже.

Таблица 113

Команда	Описание
ntp authentication-key <1-65535> md5 string	Задание ключа для аутентификации сервера. Первое значение является порядковым номером ключа. Сам ключ задается в открытом виде, после чего хранится в зашифрованном
ntp server <ip-адрес сервера> ... <ip-адрес сервера> <key>	Задание ip-адреса NTP сервера. В строку может быть задано несколько адресов серверов через пробел с одинаковым номером ключа. Аргумент с номером ключа не является обязательным
ntp server <ip-адрес сервера> ... <ip-адрес сервера> mgmt	Указание на работу протокола только через management порт
ntp server <ip-адрес сервера> ... <ip-адрес сервера> <имя виртуального маршрутизатора> <key>	Задание ip-адреса NTP сервера доступный через виртуальный маршрутизатор и номер ключа
ntp timezone <часовой пояс UTC>	Задание временного пояса. Возможные значения UTC, UTC+1...UTC-12.
ntp date <гггг.мм.дд> <чч:мм>	Задание даты и времени

28.1 Базовая настройка

Шаг 1. Настройка производится из конфигурационного режима.

```
ecorouter>en
ecorouter#conf t
Enter configuration commands, one per line. End with CNTL/Z.
```

Шаг 2. Настройка адреса сервера.

```
ecorouter(config)#ntp server 89.109.251.21
```

Шаг 3. Настройка временной зоны.

```
ecorouter(config)#ntp timezone ?
utc Greenwich Mean Time, Universal Time (Default)
utc+1 Central European Time
utc+10 Vladivostok Time
utc+11 Magadan Time
utc+12 Kamchatka Time
utc+2 Eastern European Time, Kaliningrad Time
```

```

utc+3 Further-eastern European Time, Moscow Time
utc+4 Samara Time
utc+5 Yekaterinburg Time
utc+6 Omsk Time
utc+7 Krasnoyarsk Time
utc+8 Irkutsk Time
utc+9 Yakutsk Time
utc-1 East Greenland Time
utc-10 Hawaii-Aleutian Standard Time
utc-11 Samoa Standard Time
utc-2 South Georgia Time
utc-3 West Greenland Time
utc-4 Atlantic Standard Time
utc-5 Eastern Standard Time
utc-6 Central Standard Time
utc-7 Mountain Standard Time
utc-8 Eastern Standard Time
utc-9 Alaska Standard Time
ecorouter(config)#ntp timezone UTC+3

```

Для применения результата выполнения команды **ntp timezone** необходимо сохранить конфигурацию командой **write**.

Шаг 4. Настройка текущей даты и времени вручную.

```
ecorouter(config)#ntp date 2016.07.01 11:35
```

Устройство будет использовать последнее заданное время. В случае если сначала было указано время с помощью команды **ntp date**, то оно будет использоваться до тех пор, пока не будет получено время с указанного ntp сервера.

28.2 Команды просмотра NTP

Таблица 114

Команда	Описание
show ntp status	Отображает адреса ntp-серверов для синхронизации
show ntp date	Отображает текущую дату и время
show ntp timezone	Отображает текущий часовой пояс

Команда **show ntp status** отображает список всех использующихся серверов и сервер, с которым устройство синхронизирует системное время.

```

ecorouter#show ntp status
Status  Description
*      best
+      sync
-      failed
-----
-----
Status |   VR name    |   Server     | Stratum |   Delay   | Version
|   Offset  |   Last
-----
* |       mgmt    |  95.104.192.10 |      2    |  0.0441  |      4    |  0.0001
|   6

```

+	mgmt		91.206.16.3		2		0.0639		4		0.0034
0											

Синхронизация будет производиться с сервером с наименьшим стратумом или, в случае если стратумы совпадают, с сервером, до которого минимальная задержка при эхо-запросе.

Команда просмотра часового пояса на устройстве.

```
ecorouter#show ntp timezone  
System Time zone: UTC
```

Команда просмотра текущей даты на устройстве.

```
ecorouter#show ntp date  
Wed Jul 13 12:08:23 UTC 2016
```

29 Precision Time Protocol

PTP (Precision Time Protocol) - протокол, используемый для синхронизации часов по компьютерной сети. В локальных сетях он обеспечивает точность синхронизации до десятков наносекунд (для сравнения, протокол NTP может обеспечить точность синхронизации до миллисекунд), которая требуется для некоторых измерительных систем и систем управления. Существует две версии протокола, поддерживает только вторую, т.е. PTPv2. Протокол PTP работает по принципу master-slave, т.е. в одной схеме синхронизации должен присутствовать источник (master) и приемник синхронизации (slave). Устройства, которые не являются источником или приемником синхронизации, могут участвовать в схеме распространения синхронизации в качестве промежуточных устройств при условии заполнения correction field в соответствующих PTP-пакетах.

Существуют следующие типы устройств, участвующих в схеме распространения синхронизации по протоколу PTPv2:

- ordinary clock (устройство, которое участвует в схеме только в одной роли - master или slave);
- boundary clock (устройство, которое участвует в схеме в обеих ролях - master и slave. Например, принимает синхронизацию из одного сегмента сети в роли slave и передает синхронизацию в другой сегмент сети в роли master);
- transparent clock (устройство, которое участвует в схеме в качестве промежуточного узла между master и slave и заполняет correction field в соответствующих PTP-пакетах).

Существуют следующие режимы работы протокола PTPv2:

- E2E (end-to-end - корректировка учитывает только время задержки на промежуточных устройствах);
- P2P (peer-to-peer - корректировка учитывает время задержки на промежуточных устройствах, а также время распространения сигнала между промежуточными устройствами).

Существуют следующие уровни работы протокола PTPv2:

- L2 (IEEE 802.3 Ethernet с использованием следующих multicast адресов: 01-1B-19-00-00-00, 01-80-C2-00-00-0E);
- L3 (IPv4/IPv6 с использованием следующих multicast адресов: 224.0.1.129/FF0x::181, 224.0.0.107/FF02::6B).

В текущей реализации маршрутизатор поддерживает L2/L3 E2E transparent/boundary clock режимы работы.

Команда конфигурационного режима (config) для настройки PTPv2 на маршрутизаторе имеет вид:

```
ptp mode {transparent|boundary} {e2e|p2p} {ethernet|udp}
```

Параметры команды представлены в таблице ниже.

Таблица 115

Параметр	Описание
transparent boundary	Тип часов. transparent - transparent clock; boundary - boundary clock
e2e p2p	Режим работы протокола PTPv2. e2e – End-to-End режим; p2p – Peer-to-Peer режим
ethernet udp	Режим сообщений. ethernet – L2-режим; udp – L3-режим

Примечание: режим работы **udp** будет доступен для настройки только после указания ip-адреса для отправки служебных сообщений. Команда конфигурационного режима (config) для настройки ip-адреса для отправки служебных сообщений имеет вид:

```
ptp source <A.B.C.D>
```

Команда контекстного конфигурационного режима (config-port) для включения на выбранном порту PTPv2 имеет вид:

```
ptp {transparent|slave|master|bmca}
```

В результате выполнения этой команды на соответствующем порту будет включен протокол PTPv2 в режиме **transparent**, **slave**, **master** или будет включен алгоритм выбора грандмастера - **bmca** (Best Master Clock Algorithm), который позволит автоматически определить режим работы порта (**master** или **slave**).

Режим порта **transparent** доступен, только если маршрутизатор настроен для работы по типу **transparent**.

Режимы порта **slave** и **master** доступны только, если маршрутизатор настроен для работы по типу **boundary**.

При включении **bmca** с настройками по умолчанию значения параметров **priority1** и **priority2** равны 128. Значения приоритетов для заполнения соответствующих полей в анонсах можно изменить при помощи команды конфигурационного режима (config):

```
ptp announcement priority <0-255> <0-255>
```

29.1.1 Команды просмотра

Таблица 116

Команда и результат ее выполнения	Комментарий
show ptp status	Показать текущий статус PTP
Device type: boundary Delay measurement mechanism: end-to-end Mode: udp Clock ID: 1c8776ffffe4005a1 Ports: ge3: slave	Тип часов Режим измерения задержки Режим сообщений ID часов Порты, участвующие в PTP, и их режимы
show ptp boundary-clock	Показать подробную информацию PTP (только для типа boundary)
ge3: State: slave Assigned by: static	Порт, информация о котором показана Режим порта Способ задания режима порта (static/bmc)

Команда и результат ее выполнения	Комментарий
Grandmaster ID: 1c8776ffffe4005a1 Priority: N/A Offset: 456 ns Path Delay: 783 ns	ID Grandmaster часов Приоритет часов. Используется для ВМС (для статического способа задания режима порта N/A) Последнее значение рассчитанного смещения в наносекундах (если режим порта master , N/A) Последнее значение рассчитанной задержки передачи сообщения в наносекундах

30 Flow export

Протокол использует UDP и порт 4739 для передачи данных коллектору.

Сенсор выделяет из проходящего трафика потоки, характеризуемые следующими совпадающими параметрами:

- адрес источника;
 - адрес назначения;
 - порт источника для UDP и TCP;
 - порт назначения для UDP и TCP;
 - тип и код сообщения для ICMP;
 - номер протокола IP;
 - сетевой интерфейс (параметр ifindex SNMP);
 - IP Type of Service;
- маска источника;
 - маска назначения.

Потоком считается набор пакетов, проходящих в одном направлении. Когда сенсор определяет, что поток закончился (по изменению параметров пакетов, либо по сбросу TCP-сессии), он отправляет информацию в коллектор. В зависимости от настроек он также может периодически отправлять в коллектор информацию о все еще идущих потоках.

Для управления сенсорами используются объекты конфигурации, называемые профилями сенсоров (**flow-export-profile**). Для создания профиля сенсора используется команда конфигурационного режима **flow-export-profile <NUM>**, где <NUM> - индекс профиля.

Для настройки профиля используется та же команда. Команды, доступные в режиме конфигурирования профиля, описаны в таблице ниже.

Таблица 117

Команда	Описание
description <DESCRIPTION>	Создание описания профиля
destination <IP> [port <1-65535>] [vrf <NAME>]	IP-адрес коллектора. Адрес задается в формате А.В.С.Д. После указания адреса можно указать UDP-порт коллектора. Также можно указать виртуальную таблицу маршрутизации (VRF), через которую будет производиться передача данных (параметр недоступен для виртуальных маршрутизаторов)
packet-sampling <50-1000>	Порядковый номер пакета из потока, который будет передан на коллектор. Например, каждый 50-ый. Значение по умолчанию - 500
timeout active <1-300>	Временной интервал, по истечении которого данные будут переданы на коллектор при активной сессии, в секундах. Значение по умолчанию - 60
timeout inactive<5-300>	Временной интервал, по истечении которого данные будут переданы на коллектор после закрытия сессии, в секундах. Значение по умолчанию - 15

Команда	Описание
timeout template <1-30>	Временной интервал, по истечении которого на коллектор будет передан шаблон сообщений о потоке, в секундах. Значение по умолчанию - 15

Привязка профиля сенсора к интерфейсу осуществляется при помощи контекстной команды режима конфигурирования интерфейса **flow-export-profile <NUM>**.

Настройка профилей сенсоров также доступна для виртуальных маршрутизаторов. Команды конфигурирования, аналогичные описанным выше, вводятся в интерфейсе виртуального маршрутизатора.

30.1 Пример настройки

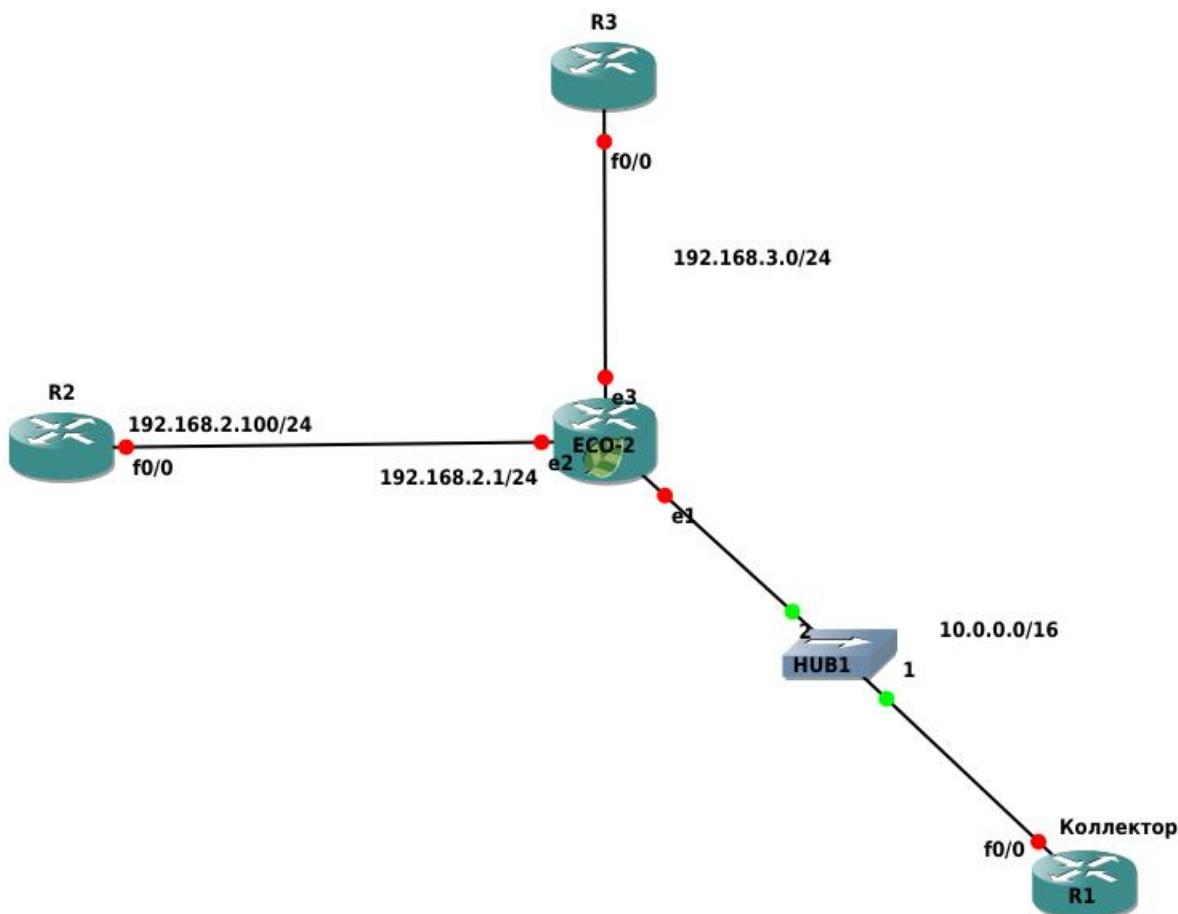


Рисунок 52

В данном сценарии приводится настройка сенсора на интерфейсе e3 устройства ECO-2.

Шаг 1. Настройка осуществляется в режиме глобальной конфигурации.

```
ecorouter>en
ecorouter#configure terminal
```

Шаг 2. Настройка интерфейсов и портов устройства.

```
ecorouter(config)#interface e1
ecorouter(config-if)#ip add 172.16.0.1/16
```

```

ecorouter(config)#interface e2
ecorouter(config-if)#ip add 192.168.2.1/24
ecorouter(config)#interface e3
ecorouter(config-if)#ip add 192.168.3.1/24
ecorouter(config)#port te0
ecorouter(config-port)#service-instance te0/e1
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip int e1
ecorouter(config)#port te1
ecorouter(config-port)#service-instance te1/e2
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip int e2
ecorouter(config)#port te2
ecorouter(config-port)#service-instance te2/e3
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip int e3

```

Шаг 3. Создание профиля сенсора.

```

ecorouter(config)#flow-export-profile 10
ecorouter(config-flow-export)#description Netflow

ecorouter(config-flow-export)#destination 172.16.0.2
ecorouter(config-flow-export)#packet-sampling 1

ecorouter(config-flow-export)#timeout active 30
ecorouter(config-flow-export)#timeout inactive 30

```

Шаг 4. Назначение профиля сенсора на интерфейс.

```

ecorouter(config)#interface e3
ecorouter(config-if)#flow-export-profile 10

```

30.2 Команды просмотра

Просмотр сконфигурированного профиля осуществляется командами административного режима **show flow-export-profile** и **show flow-export-profile <NUM>**. Эти команды выводят весь список сконфигурированных сенсоров на устройстве без указания номера профиля и определенный профиль с номером.

```

ecorouter#show flow-export-profile
NetFlow profile 10
Description: Netflow
Destination: 172.16.0.2
Active timeout: 30
Inactive timeout: 30
Packet sampling: 1

```

Для просмотра статистики по Netflow используется та же команда административного режима, что и для просмотра информации о состоянии интерфейса - **show interface <NAME>**.

Пример.

```

ecorouter#sh interface e1
Interface e1 is up
Ethernet address: 1c87.7640.d603

```

```
MTU: 100
ICMP redirection is on
Label switching is disabled
<UP,BROADCAST,RUNNING,MULTICAST>
Connect service instance te0.te0/e1 symmetric
inet 10.0.0.1/16 broadcast 10.0.255.255/16
NetFlow profile 0
Destination: 10.0.0.2:9996
Total packets: 2077, dropped packets: 0, flow count: 10
total input packets 103844, bytes 6647020
total output packets 100917, bytes 6463274
```

Здесь:

Total packets - количество пакетов, переданных в netflow буфер маршрутизатора,

dropped packets - количество пакетов, не переданных в netflow буфер в результате возникшей ошибки,

flow count - количество потоков в буфере.

31 CoPP

CoPP (Control-Plane Policing) - политика уровня управления.

Политика уровня управления служит для защиты от возможных атак на сетевое оборудование. Весь трафик, поступающий на уровень контроля с уровня коммутации, проходит через фильтрующие правила. CoPP ограничивает полосу пропускания для наиболее известных протоколов. Таким образом при атаке на сетевое оборудование количество пакетов, попадающих на уровень контроля, не будет превышать установленный порог полосы пропускания. Если по конкретному протоколу наблюдаются растущие потери, то можно предположить, что в сети существует аномальное количество трафика по этому протоколу.

Полосы пропускания CoPP, заданные по умолчанию для маршрутизатора EcoRouter, описаны в таблице ниже.

Таблица 118

Протокол	Количество пакетов в секунду
Входящий OSPF	512
Входящий IS-IS	512
Входящий LDP	512
Входящий ARP	128
Входящий Multicast IGMP	128
Входящий ICMP	128
Входящий SSH	512
Входящий BGP	512
Входящий Multicast	512
Входящий L2	256
Входящий прочий трафик	1024
Исходящий ICMP	128
Исходящий прочий трафик	1024

В CLI EcoRouterOS пользователь может ограничить полосу пропускания трафика для протоколов, перечисленных в таблице, в СР маршрутизатора. Настройки защиты от DoS и DDoS атак доступны на интерфейсах и портах, а также и глобально на СР устройства. Переход в режим конфигурирования СР осуществляется по команде **control-plane** в конфигурационном режиме. Пользователь может одновременно настроить защиту в разных режимах (на разных элементах устройства). Команды ограничения полосы пропускания (количество пакетов в секунду) для различных протоколов представлены в таблице.

Таблица 119

Команда	Режимы	Описание
rate-limit dhcp-discovery <0-262144>	(config-cp), (config-port), (config-port-channel), (config-int)	Общее ограничение полосы пропускания сообщений DHCP discover от всех клиентов
rate-limit dhcp-discovery per-interface <0-262144>	(config-int)	Общее ограничение полосы пропускания сообщений DHCP discover на интерфейсе от всех клиентов
rate-limit dhcp-discovery per-subscriber <0-15>	(config-int)	Ограничение полосы пропускания сообщений DHCP discover от одного клиента

Команда	Режимы	Описание
rate-limit arp <0-524288>	(config-cp), (config-port), (config-port-channel), (config-int)	Общее ограничение полосы пропускания сообщений ARP request от всех клиентов
rate-limit arp per-interface <0-524288>	(config-int)	Общее ограничение полосы пропускания сообщений ARP request на интерфейсе от всех клиентов
rate-limit arp per-subscriber <0-524288>	(config-int)	Ограничение полосы пропускания сообщений ARP request от одного клиента
rate-limit multicast-other <0-262144>	(config-cp)	Общее ограничение входной полосы пропускания мультикастного трафика
rate-limit multicast-igmp <0-262144>	(config-cp)	Общее ограничение входной полосы пропускания IGMP трафика
rate-limit ldp <0-2048>	(config-cp)	Общее ограничение входной полосы пропускания LDP трафика
rate-limit isis <0-2048>	(config-cp)	Общее ограничение входной полосы пропускания IS-IS трафика
rate-limit ospf <0-2048>	(config-cp)	Общее ограничение входной полосы пропускания OSPF трафика
rate-limit ssh <0-2048>	(config-cp)	Общее ограничение входной полосы пропускания SSH трафика
rate-limit bgp <0-2048>	(config-cp)	Общее ограничение входной полосы пропускания BGP трафика
rate-limit dhcp-other <0-4096>	(config-cp)	Общее ограничение входной полосы пропускания всех сообщений DHCP от всех клиентов
rate-limit icmp <0-512> (in out)	(config-cp)	Общее ограничение полосы пропускания для ICMP трафика в различных направлениях
rate-limit unicast-other <0-524288> (in out)	(config-cp)	Общее ограничение полосы пропускания для юникастового трафика в различных направлениях
rate-limit non-ip <0-2048>	(config-cp)	Общее ограничение входной полосы пропускания для любого не IP трафика от всех клиентов
rate-limit all <0-524288> (in out)	(config-cp)	Общее ограничение полосы пропускания для любого трафика в различных направлениях

31.1 Команды просмотра

Для просмотра текущего состояния счетчиков политики уровня управления необходимо в режиме администрирования выполнить команду **show counters copp**.

ecorouter#show counters copp
Received

packets bytes dropped

Total 196886 21667244 0
OSPF 29645 2439234 0
ISIS 0 0 0
LDP 0 0 0
ARP 3 180 0

IGMP	63300	3804506	0
SSH	143	18324	0
ICMP	17	1770	0
BGP	17534	1340980	0
MCAST	85009	13987600	0
L2	1078	64680	0
Other	157	9970	0
Transmitted			

	-----	packets	bytes
	-----		dropped
Total	312702	18649358	0
ICMP	14759	1033130	340
Other	297943	17616228	0

В данном выводе отображено количество входящих/исходящих пакетов, входящих/исходящих байтов, а также количество сброшенных пакетов (из-за превышения порога полосы пропускания).

Для очистки текущих значений счетчиков необходимо выполнить команду **clear counters copp** в режиме конфигурирования.

```
ecorouter(config)#clear counters copp
```

32 Поток Е1

E1 – цифровой метод передачи данных и голоса, основанный на временном разделении канала. Кадр потока E1 состоит из 32 временных интервалов с 0 по 31, называемых таймслотами (timeslot). Каждый таймслот, в свою очередь, содержит 8 бит информации. За одну секунду передается 8000 кадров, следовательно, скорость передачи данных по каналу E1 может достигать 2048 Кбит/с.

Нулевой таймслот служит для сигнализации. В нем передается управляющая информация. Таким образом для передачи данных используется 31 таймслот (с 1 по 31). Такой режим работы называется структурированным режимом (framed). Однако нулевой таймслот также может быть задействован под передачу данных, – такой режим работы называется неструктурированным режимом работы (unframed). При структурированном режиме необходимо указать, какие таймслоты будут использоваться для передачи данных. В случае использования всех оставшихся доступных таймслотов запись будет иметь вид – 1-31. Значение используемых таймслотов на устройствах, соединенных одной линией передачи, должно совпадать.

Для тестирования потока существуют два режима: **loopback local** и **loopback networkline**. Первый режим служит для тестирования локального порта E1, второй – для магистрали между оборудованием.

Существует режим отслеживания ошибок, называемый CRC-4. Если данный режим включен, происходит расчет контрольной суммы при отправлении и на удаленной стороне. Если принятая и рассчитанная сумма совпадают, то кадр считается целым. Бит контрольной суммы находится в нулевом таймслоте. Для того, чтобы посчитать контрольную сумму, устройство группирует 16 таймслотов, эта группа называется мультикадром. Данный режим включается опционально. На обоих сторонах магистрали режимы должны совпадать.

Маршрутизатор использует два типа инкапсуляции в потоке E1: HDLC и PPP. Тип инкапсуляции на обоих сторонах должен совпадать.

32.1 Порты и каналы Е1

Некоторые модели маршрутизаторов EcoRouter поддерживают передачу данных через цифровые интерфейсы первичного уровня европейского стандарта плезиохронной цифровой иерархии (PDH), известные как E1. Технические характеристики интерфейса E1 соответствуют рекомендации МСЭ-Т G.703/6. Битовая скорость потока E1 – 2048 Кбит/с. В качестве физического канала передачи используется симметричная витая пара с импедансом 100–120 Ом, в качестве разъемов – коннекторы 8P8C, известные также как RJ45. На рисунке ниже приведена разводка линий по контактам разъема.

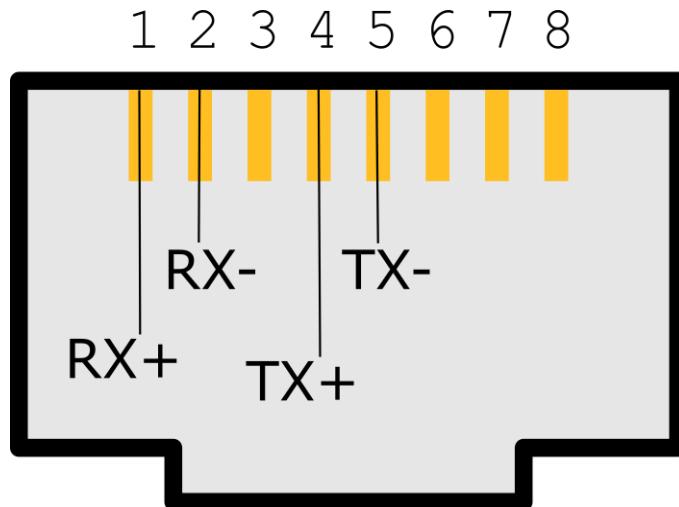


Рисунок 53

Поддерживаются как неструктурированные потоки E1, так и структурированные (framed, structured, channelised) в соответствии с рекомендацией МСЭ-Т G.704. В последнем случае нулевой канальный интервал (тайм-слот) используется для синхронизации, и максимальная пропускная способность снижается до 1984 Кбит/с. Выделение отдельных канальных интервалов для формирования канальных групп не поддерживается.

32.1.1 Настройка контроллера

В EcoRouterOS с интерфейсом E1 связаны два объекта конфигурации: контроллер (**controller**) и порт (**port**). Контроллеры создаются в конфигурации автоматически при подключении интерфейсной карты E1. Если в данной модели EcoRouter отсутствует интерфейсная карта E1, то контроллеры будут недоступны для конфигурирования.

Имена контроллеров E1, заданные системой: **e1.1** и **e1.2**.

Для настройки контроллеров используется команда конфигурационного режима **controller e1.<NUM>**, где <NUM> - номер контроллера, соответственно. После этого в режиме конфигурирования контроллера будут доступны команды настройки параметров, приведенные в таблице ниже.

Таблица 120

Команда	Описание
clocking {internal remote}	Выбор источника синхронизации: internal – внутренний источник синхронизации, remote – удаленный источник синхронизации
framing {crc4 nocrc4 unframed}	Настройка структуры кадров: crc4 – включен режим CRC-4, nocrc4 – выключен режим CRC-4, unframed – включен неструктурированный режим
loopback {local remote}	Включение режима петли: local – петля на локальном оборудовании, remote – петля на удаленном оборудовании

Пример настройки контроллера.

```
ecorouter#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#controller e1.1
ecorouter(config-contr-e1)#framing nocrc4
```

```
|ecorouter(config-contr-e1)#clocking internal
```

Для диагностики контроллеров используются команды административного режима **show controller** (для вывода информации обо всех контроллерах) и **show controller e1.<NUM>** (для вывода информации о конкретном контроллере).

```
ecorouter#show controller e1.1
Controller e1.1
Clocking source: internal
Framing: no-crc4
Loopback mode: off
 1-32      free
```

32.1.2 Настройка порта E1

Порты, связанные с контроллерами E1, создаются пользователем, а имена портов указывают на тип инкапсуляции, которая будет использоваться для передачи кадров. EcoRouter поддерживает два типа инкапсуляции: HDLC и PPP, поэтому имена портов будут иметь вид **hdlc.<NUM>** для инкапсуляции HDLC и **ppp.<NUM>** – для PPP, где <NUM> – номер порта.

Подробнее о создании и настройке порта можно прочитать в разделе "Виды интерфейсов. Порт". Специфичные для портов E1 настройки приведены в таблице ниже. Все они выполняются в контекстном режиме конфигурирования порта.

Таблица 121

Команда	Описание
timeslots controller e1.<NUM> (1-31)	Выделение таймслотов с контроллера E1, где <NUM> – номер контроллера. Для режима unframed диапазон таймслотов не указывается
service instance <NAME>	Задание сервисного интерфейса
encapsulation untagged	Задание нетегированной инкапсуляции. Обязательная команда
connect ip interface <NAME>	Привязывание IP-адреса интерфейса к данному порту. Интерфейс, который привязывается к порту с инкапсуляцией HDLC, должен иметь MTU не более 1486 байт

Пример настройки порта PPP.

```
ecorouter#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#interface ppp0
ecorouter(config-contr-e1)#ip address 10.1.1.1/30
ecorouter(config)#interface ppp0
ecorouter(config)#port ppp.0
ecorouter(config-port-ppp)#timeslots controller e1.1 1-31
ecorouter(config-port-ppp)#service-instance unit0
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface ppp0
```

Для диагностики портов используются команды административного режима **show port** (для вывода информации обо всех портах) и **show port <NAME>** (для вывода информации о конкретном порте).

```
|ecorouter#show port ppp.0
```

```

PPP port ppp.0 is up [10.1.1.1/30]
PPP authentication is off
MTU: 17940
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
Service instance ppp.0.unit0 is up
ingress encapsulation untagged
ingress rewrite none
egress encapsulation untagged
egress none
Connect interface mppp0 symmetric
Input packets 6, bytes 588
Output packets 26, bytes 1484

```

32.1.3 Настройка аутентификации

Для инкапсуляции PPP можно задать аутентификацию для идентификации удаленной стороны. В EcoRouter для аутентификации используется протокол CHAP. Режим аутентификации задается контекстной командой настройки порта **ppp** или **mppp** (Multilink ppp). Для порта **mppp** аутентификация конфигурируется на объединенном порту Multilink.

Задание аутентификации по протоколу CHAP выполняется при помощи команды **authentication chap hostname <LOCAL-NAME> username <REMOTE-NAME> password <PASS>**. Здесь <LOCAL-NAME> – имя локальной машины (hostname маршрутизатора или любое другое имя), <REMOTE-NAME> – имя удаленной машины, <PASS> – пароль для данного подключения.

Пример настройки порта PPP.

```

ecorouter#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#interface ppp0
ecorouter(config-contr-e1)#ip address 10.1.1.1/30
ecorouter(config)#interface ppp0
ecorouter(config)#port ppp.0
ecorouter(config-port-ppp)#timeslots controller e1.1 1-31
ecorouter(config-port-ppp)#authentication chap hostname Bob username
Clara password supersecret
ecorouter(config-port-ppp)#service-instance unit0
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface ppp0

```

Для диагностики портов используются команды административного режима **show port** (для вывода информации обо всех портах) и **show port <NAME>** (для вывода информации о конкретном порте).

```

ecorouter#show port ppp.0
PPP port ppp.0 is up [10.1.1.1/30]
PPP authentication is on
protocol: chap
hostname: Bob
username: Clara
MTU: 17940
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0

```

```
Service instance ppp.0.unit0 is up
ingress encapsulation untagged
ingress rewrite none
egress encapsulation untagged
egress none
Connect interface mppp0 symmetric
Input packets 6, bytes 588
Output packets 26, bytes 1484
```

32.2 Настройка Multilink PPP

Для увеличения пропускной способности и обеспечения отказоустойчивости можно объединить два порта **ppp** в один логический порт – Multilink PPP. Такой порт будет называться **mppp.<NUM>**, где <NUM> – номер порта. Для создания **mppp** порта, необходимо сконфигурировать два **ppp** порта и добавить их в один **mppp** порт.

Для создания порта для Multilink PPP используется команда конфигурационного режима **port mppp.<NUM>**, где <NUM> – номер порта. Далее в режиме конфигурирования созданного порта необходимо добавить порты **ppp** в Multilink при помощи команды **bind ppp.<NUM>**, где <NUM> – номер порта.

Пример настройки Multilink PPP.

```
ecorouter(config)#interface mppp0
ecorouter(config-if)#ip address 10.3.3.2/30
ecorouter(config-if)#exit
ecorouter(config)#port ppp.0
ecorouter(config-port-ppp)#timeslots controller e1.1
ecorouter(config-port-ppp)#port ppp.1
ecorouter(config-port-ppp)#timeslots controller e1.2
ecorouter(config-port-ppp)#exit
ecorouter(config)#port mppp.0
ecorouter(config-port-mppp)#bind ppp.0
ecorouter(config-port-mppp)#bind ppp.1
ecorouter(config-port-mppp)#service-instance unit0
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect ip interface mppp0
```

Для диагностики портов используется команда административного режима **show port mppp.<NUM>**, где <NUM> – номер порта.

```
ecorouter#show port mppp.0
Multilink PPP port mppp.0 is up [10.3.3.2/30]
PPP authentication is off
PPP port ppp.0
PPP port ppp.1
MTU: 17940
Input packets 0, bytes 0, errors 0
Output packets 0, bytes 0, errors 0
Service instance mppp.0.unit0 is up
ingress encapsulation untagged
ingress rewrite none
egress encapsulation untagged
egress none
```

```
Connect interface mppp0 symmetric
Input packets 0, bytes 0
Output packets 3, bytes 126
```

33 Виртуальные машины и контейнеры

33.1 Виртуальные машины и контейнеры. Общие сведения

На платформе маршрутизатора кроме встроенного программного обеспечения EcoRouterOS может быть запущено программное обеспечение сторонних производителей. Для этого используются технологии виртуализации двух типов:

- полная виртуализация на базе QEMU/KVM;
- контейнерная виртуализация на базе Docker.

Полная виртуализация позволяет запускать операционные системы и эмулировать платформы, поддерживаемые QEMU/KVM. Если стороннее программное обеспечение работает на Linux и не требует эмуляции дополнительного оборудования, то более подходящим вариантом будет контейнерная виртуализация на основе одной ОС.

Функционал виртуальных машин и контейнеров позволяет отказаться от приобретения и поддержки дополнительных серверов и разместить непосредственно на маршрутизаторе программное обеспечение для различных сетевых сервисов.

При конфигурировании виртуальных машин и контейнеров необходимо различать два варианта взаимодействия:

управление виртуальной машиной, которое производится внешними средствами (создание, запуск, остановка, уничтожение);

конфигурирование подключения интерфейсов виртуальной машины к портам EcoRouter, которое делается из командной строки EcoRouterOS.

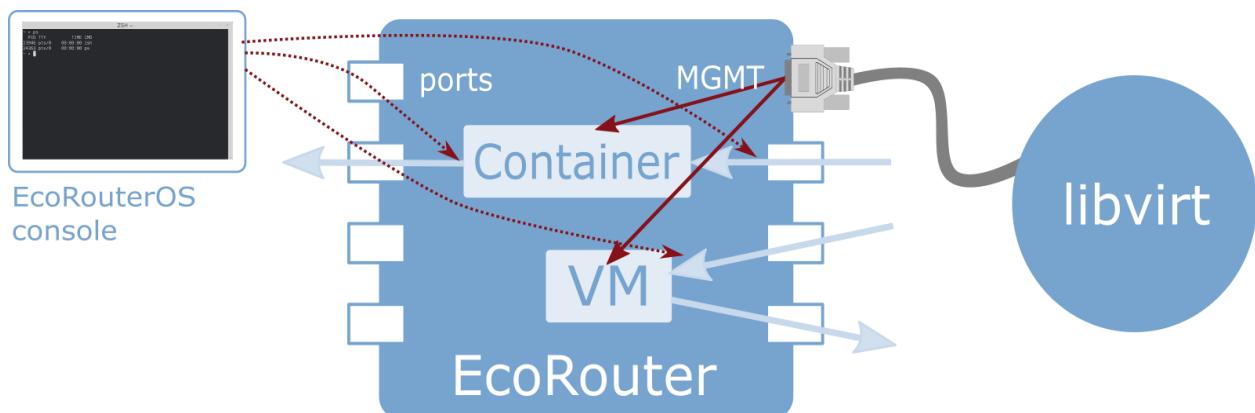


Рисунок 54

Внимание! При использовании сетевых интерфейсов с драйвером **virtio** необходимо отключить **TCP offload engine**, так как на данный момент существует ошибка при подсчете контрольной суммы в TCP-заголовке.

Отключить **TCP offload engine** можно следующими способами:

1. В ОС на виртуальной машине выполнить следующую команду:

```
ethtool --offload eth0 tx off
```

2. В **virsh** отредактировать свойства сетевого интерфейса, добавив следующие строки:

```
<host csum='off' gso='off' tso4='off' tso6='off' ecn='off' ufo='off'
mrg_rxbuf='off' />
<guest csum='off' tso4='off' tso6='off' ecn='off' ufo='off' />
```

Для этого необходимо выполнить следующие действия:

2.1. подключиться к удаленному хосту:

```
virsh -c qemu+tls://admin@ecorouter/system
```

2.2. остановить виртуальную машину:

```
shutdown virt_name
```

2.3. войти в режим редактирования xml-файла настроек для этой машины:

```
edit virt_name
```

2.4. в секцию **interface** добавить следующие строки:

```
<driver>
  <host csum='off' gso='off' tso4='off' tso6='off' ecn='off' ufo='off'
mrg_rxbuf='off' />
  <guest csum='off' tso4='off' tso6='off' ecn='off' ufo='off' />
</driver>
```

2.5. сохранить файл и выйти;

2.6. запустить виртуальную машину и проверить применение данных опций:

```
ethtool -k ifname
```

33.2 Конфигурирование подключения интерфейсов виртуальной машины к EcoRouter

Маршрутизатор EcoRouter предоставляет для виртуальных машин виртуальные порты, которые могут быть отображены в физические, либо к ним могут подключаться маршрутизуемые L3 интерфейсы.

Команда конфигурационного режима **enable container** включает функционал работы с виртуальными машинами или контейнерами.

Просмотр существующих виртуальных сетей, используемых виртуальными машинами или контейнерами, осуществляется при помощи команды административного режима **show virtual-network vm** или **show virtual-network container** для контейнеров.

Порты виртуальных машин создаются и конфигурируются при помощи команды конфигурационного режима **port virt.<NUM>**, где <NUM> - номер виртуального порта.

В режиме конфигурации порта виртуальной машины можно связать виртуальный порт с виртуальной сетью при помощи команды **virtual-network vm <IDENTIFIER>**, где указывается идентификатор виртуального интерфейса из вывода команды **show virtual-network vm**. Для контейнеров, соответственно, используется контекстная команда **virtual-network container <IDENTIFIER>**, где указывается идентификатор виртуального интерфейса из вывода команды **show virtual-network container**.

В режиме конфигурации порта виртуальной машины также можно настроить сервисные интерфейсы командой **service-instance <NAME>**.

Дальнейшее конфигурирование средствами сервисных интерфейсов делается аналогично обычным портам (см. раздел «Сервисные интерфейсы»).

33.3 Конфигурирование доступа внешних средств управления контейнерами

Управление контейнерами осуществляется при помощи внешних менеджеров, поддерживающих API кластеров docker-контейнеров. Например, может использоваться стандартный клиент docker версии 1.12 и выше. Доступ внешних средств управления контейнерами возможен только через management-порт. Аутентификация и защита соединения обеспечивается с помощью TLS и токена кластера.

Чтобы управление контейнерами было возможно, необходимо включить EcoRouter в кластер (известный также как "swarm"). Для этого в CLI EcoRouter используется команда административного режима **virtual-container join-swarm <TOKEN> <IP> <PORT>**, где:

- <TOKEN> – 85-символьный токен кластера,
- <IP> – IP-адрес менеджера,
- <PORT> – TCP-порт менеджера.

Необходимые параметры выводятся командой **docker swarm join-token worker** на менеджере кластера.

После включения маршрутизатора в кластер дальнейшее управление осуществляется стандартными командами клиента docker режима **swarm mode**. TLS-соединение формируется автоматически и не требует конфигурирования.

При необходимости выйти из кластера используется команда административного режима **no virtual-container join-swarm**.

33.4 Копирование виртуальных дисков

В EcoRouterOS есть возможность копирования виртуальных дисков для виртуальных машин. Для этого используется команда конфигурационного режима **copy <ftp | tftp> virtual-disk <АДРЕС> <mgmt | vr default | vr <VR NAME>>**.

```
ecorouter#copy ftp virtual-disk  
ftp://ftpuser:ftpuser@192.168.255.2:/ubuntu-14.04.qcow2 mgmt  
Download of virtual disk ubuntu-14.04.qcow2 complete
```

В таблице ниже описаны варианты данной команды для FTP и TFTP-серверов.

Таблица 122

Команда	Описание
copy ftp virtual-disk ftp://user:password@xxx.xxx.xxx.xxx/filename mgmt	С FTP-сервера будет скачан указанный файл виртуального диска, FTP-сервер доступен через менеджмент-порт (mgmt)
copy ftp virtual-disk ftp://user:password@xxx.xxx.xxx.xxx/filename vr default	С FTP-сервера будет скачан указанный файл виртуального диска. Доступ к FTP-

Команда	Описание
	серверу осуществляется через интерфейс виртуального маршрутизатора, выбранного по умолчанию
copy tftp virtual-disk tftp://xxx.xxx.xxx.xxx/filename vr vrname	С TFTP-сервера будет скачан указанный файл виртуального диска. Доступ к TFTP-серверу осуществляется через интерфейс виртуального маршрутизатора с именем vrname
copy tftp virtual-disk tftp://xxx.xxx.xxx.xxx/filename mgmt	С TFTP-сервера будет скачан указанный файл виртуального диска. Доступ к TFTP-серверу осуществляется через менеджмент-порт (mgmt)

33.5 Распределение ядер между виртуальными машинами и data-plane

В EcoRouterOS предусмотрена возможность выделения ядер для виртуальных машин. Возможное количество выделенных ядер: 0 или 4.

Для этого используется команда конфигурационного режима **hw reserved-cores {0 | 4}**, где 0 означает, что ядра не будут выделены; 4 означает, что будет выделено 4 ядра.

ВНИМАНИЕ: Результат выполнения данной команды будет доступен только после сохранения конфигурации и перезагрузки маршрутизатора.

```
ecorouter(config)#hw reserved-cores 4
Changes will be applied after reboot. Please save config and reload.
ecorouter(config)#write
ecorouter(config)#reload
reboot system? (y/n) : y
```

В результате после выполнения команды **hw reserved-cores**, сохранения конфигурации и перезагрузки маршрутизатора для виртуальных машин будет выделено 4 ядра.

Проверить количество выделенных для виртуальных машин ядер можно при помощи команды **show platform cpu detail**.

33.6 Подключение к виртуальной машине

33.6.1 Подготовка клиентской машины

Для подключения к встроенной в EcoRouter системе виртуализации QEMU/KVM необходимо корректно подготовить клиентскую машину на базе Linux/Unix. Инструкция составлена и проверена на базе клиента под CentOS 7.

Для управления машиной необходимо установить библиотеку LibVirt и OpenSSL.

```
yum install libvirt openssl
```

Для управления машиной при помощи графического интерфейса также необходимо установить virt-manager и его зависимости.

```
yum install qemu-kvm python-virtinst libvirt libvirt-python virt-manager
libguestfs-tools
```

Для установки графического интерфейса в CentOS 7 используется следующая последовательность команд.

```
yum -y groups install "GNOME Desktop"
startx
```

33.6.2 Конфигурирование доступа внешних средств управления виртуальной машиной

Для управления виртуальными машинами используется программа **libvirt**. Доступ внешних средств управления виртуальными машинами возможен только через management-порт. Аутентификация и защита соединения обеспечивается с помощью протокола TLS и инфраструктуры открытых ключей (PKI). Получение сертификата Центра сертификации (CA), пользовательского сертификата и закрытого ключа пользователя описано в разделе «Авторизация в системе». Их необходимо сохранить в файлы с названиями **cacert.pem**, **clientcert.pem** и **clientkey.pem** соответственно и поместить эти файлы в директории на управляющей машине, предназначенные для их хранения. Ниже приведен пример конфигурирования для операционных систем Unix/Linux.

```
#mv cacert.pem /etc/pki/CA/
#mv clientcert.pem /etc/pki/libvirt/
#mv clientkey.pem /etc/pki/libvirt/private/
#chmod 444 /etc/pki/CA/cacert.pem
#chmod 440 /etc/pki/libvirt/clientcert.pem
/etc/pki/libvirt/private/clientkey.pem
```

Также необходимо обеспечить разрешение доменного имени маршрутизатора, прописанного в сертификатах **Subject: CN=ecorouter**. Для этого задействуется система DNS или имя прописывается в файл **/etc/hosts**.

Если ранее настройки хостов на машине не выполнялись, то файл будет выглядеть подобным образом:

```
127.0.0.1 localhost localhost.localdomain localhost4
localhost4.localdomain4
::1      localhost localhost.localdomain localhost6
localhost6.localdomain6
127.2.2.2 ecorouter
```

33.6.3 Управление гипервизором

Подключение к гипервизору устанавливается с клиентской машины при помощи средств управления, поддерживающих **libvirt**, например, **virsh** или **virt-manager**:

```
virsh -c qemu+tls://admin@ecorouter/system
```

Для примера данной командой осуществляется запрос состояния виртуального процессора виртуальной машины **show_debian**.

```
[root@localhost ~]# virsh -c qemu+tls://admin@ecorouter/system vcpuinfo
show_debian | grep State
State:      running
```

Доступ непосредственно к рабочему столу или командной строке виртуальной машины осуществляется, например, с помощью **virt-manager** или **virt-viewer**:

```
$virt-viewer -c qemu://ecorouter/system <имя_BM> &
```

В случае если используется графическая оболочка, то необходимо открыть консоль **Virtual Machine Manager**. Перейти в раздел **File – Add Connection**, заполнить появившуюся форму, как показано на рисунке ниже, и нажать **Connect**.

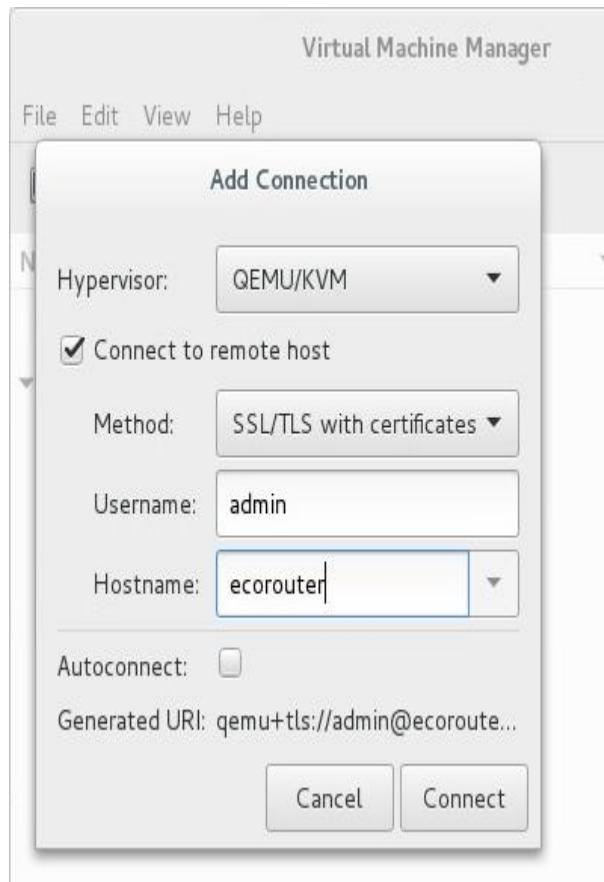


Рисунок 55

33.7 Быстрая настройка виртуальных машин

Для быстрой настройки виртуальных машин в EcoRouter необходимо произвести действия, описанные ниже.

1. Включить поддержку виртуальных машин в EcoRouter при помощи команды конфигурационного режима **enable vm**.

По умолчанию, для всех виртуальных машин используется одно ядро. В случае, если необходимо загрузить виртуальную машину с ресурсоемкими приложениями, то количество ядер может быть увеличено до 4.

Для этого используется команда конфигурационного режима **hw reserved-cores <N>**, где N - количество ядер, резервируемых под виртуальные машины.

Пример:

```
ecorouter(config)#hw reserved-cores 4
```

2. Скопировать образ виртуальной машины на EcoRouter при помощи команды режима администрирования **copy {ftp | tftp} virtual-disk**.

```
ecorouter#copy ftp virtual-disk ftp://user:password@xxx.xxx.xxx.xxx/filename
```

```
ecorouter#copy tftp virtual-disk tftp://xxx.xxx.xxx.xxx/filename
```

3. Убедиться, что на локальном компьютере, с которого будет производиться управление виртуальными машинами, установлены libvirt и openssl.

Для подключения к виртуальным машинам на EcoRouter используется утилита командной строки virsh или графический аналог virt-manager. Версия virt-manager должна быть не меньше 1.3.

4. Экспортировать на локальную машину сертификаты пользователя для подключения к libvirt на EcoRouter. Пример экспорта для Linux машин приведен в таблице ниже.

Таблица 123

Вывод команды на EcoRouter	скопировать в файл на локальном компьютере
cryptp ca export	/etc/pki/CA/cacert.pem
cryptp certificate export	/etc/pki/libvirt/clientcert.pem
cryptp key export	/etc/pki/libvirt/private/clientkey.pem

Все команды, указанные в таблице, вводятся в режиме администрирования.

Для корректной работы необходимо установить следующие права доступа на файлы:

```
chmod 444 /etc/pki/CA/cacert.pem
chmod 440 /etc/pki/libvirt/clientcert.pem
/etc/pki/libvirt/private/clientkey.pem
```

5. Добавить в файл **/etc/hosts** запись об IP-адресе EcoRouter с именем хоста - **ecorouter**.

6. Подключиться к libvirt на EcoRouter. В консоли для этого необходимо ввести команду **virsh -c qemu+tls://admin@ecorouter/system**.

В случае, если используется графическая оболочка, то необходимо открыть консоль **Virtual Machine Manager**. Перейти в раздел **File – Add Connection**, заполнить появившуюся форму, как показано на рисунке ниже, и нажать **Connect**.

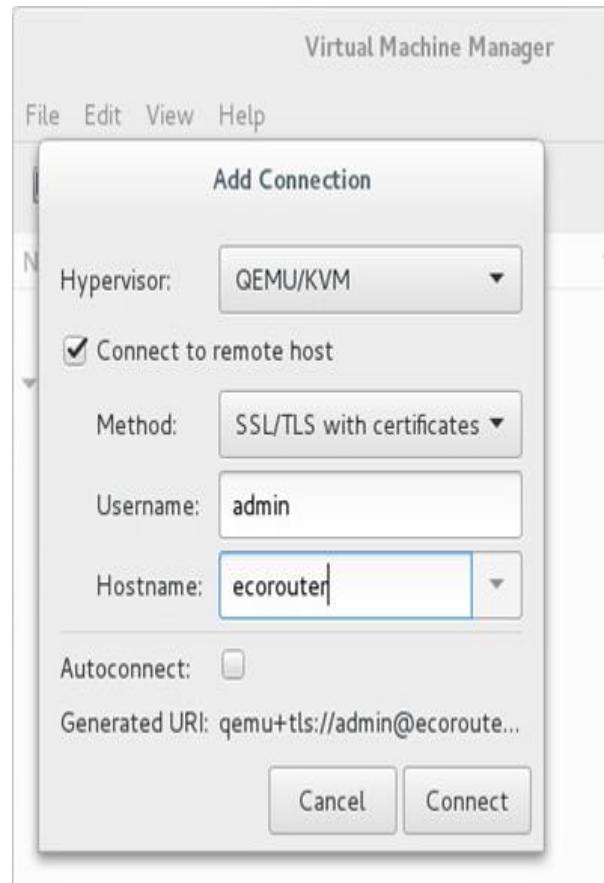


Рисунок 56

7. Создать новую виртуальную машину, используя образ жесткого диска, скопированный ранее наEcoRouter (см. шаг 2).

8. Сетевые интерфейсы виртуальных машин необходимо подключать к изолированным сетям. Для создания такой сети необходимо перейти в детали подключения к EcoRouter и создать виртуальную сеть с типом **Isolated virtual network** (изолированная виртуальная сеть).

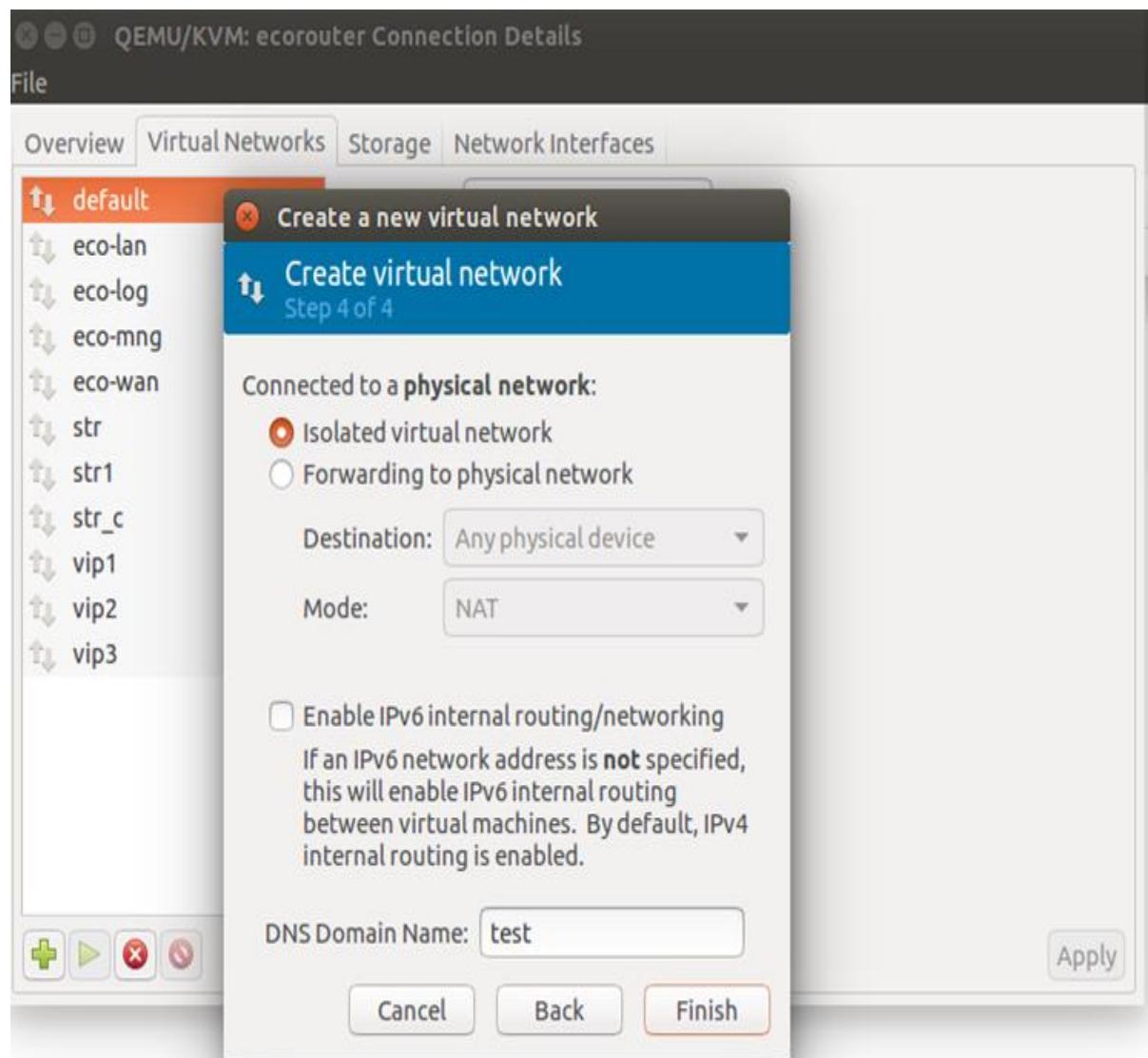


Рисунок 57

9. При необходимости добавить сетевые интерфейсы. Каждый интерфейс подключается к одной из ранее созданных виртуальных сетей.

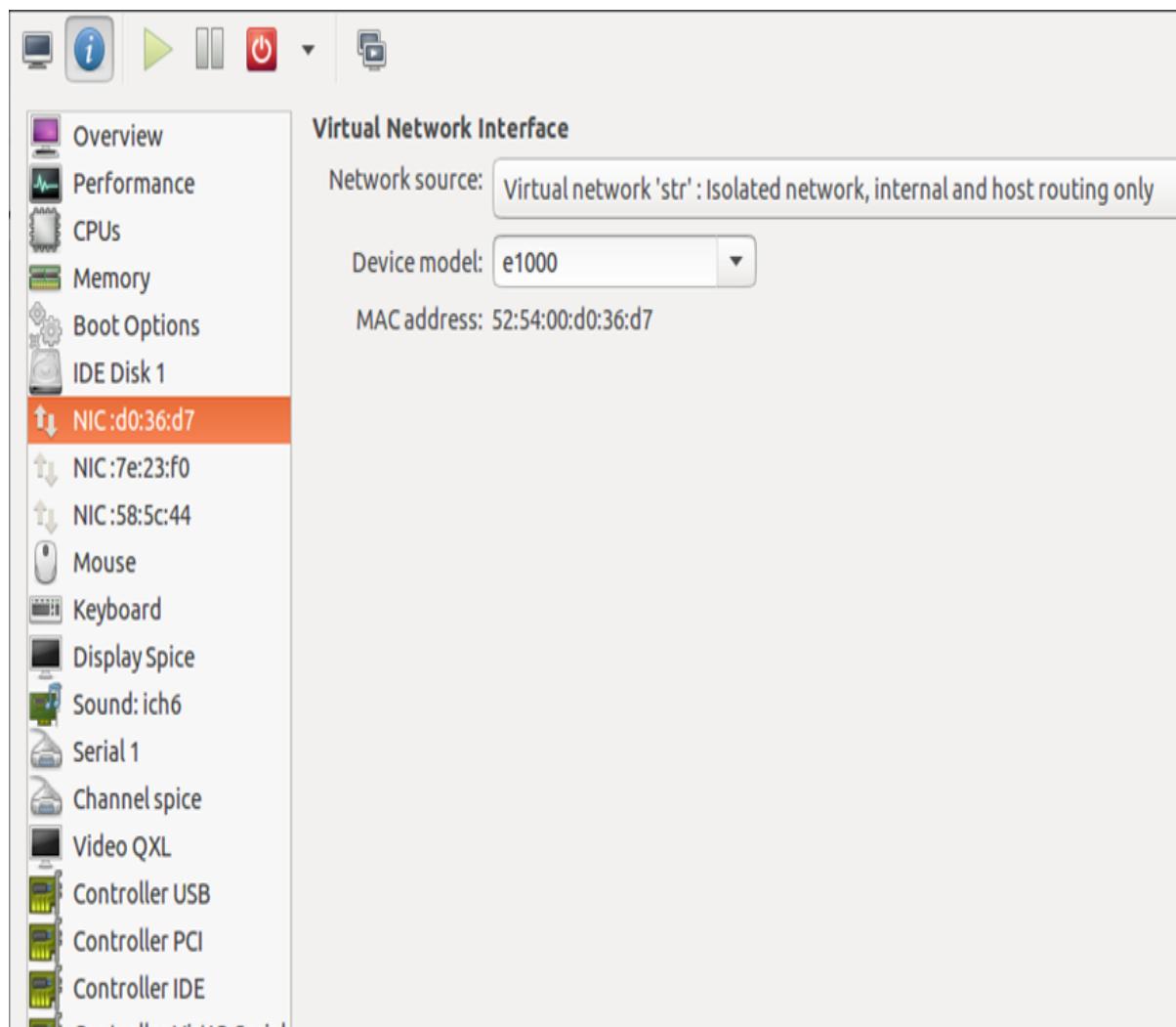


Рисунок 58

10. В пункте **Display Spice** в поле **Address** выбрать **All interfaces**.

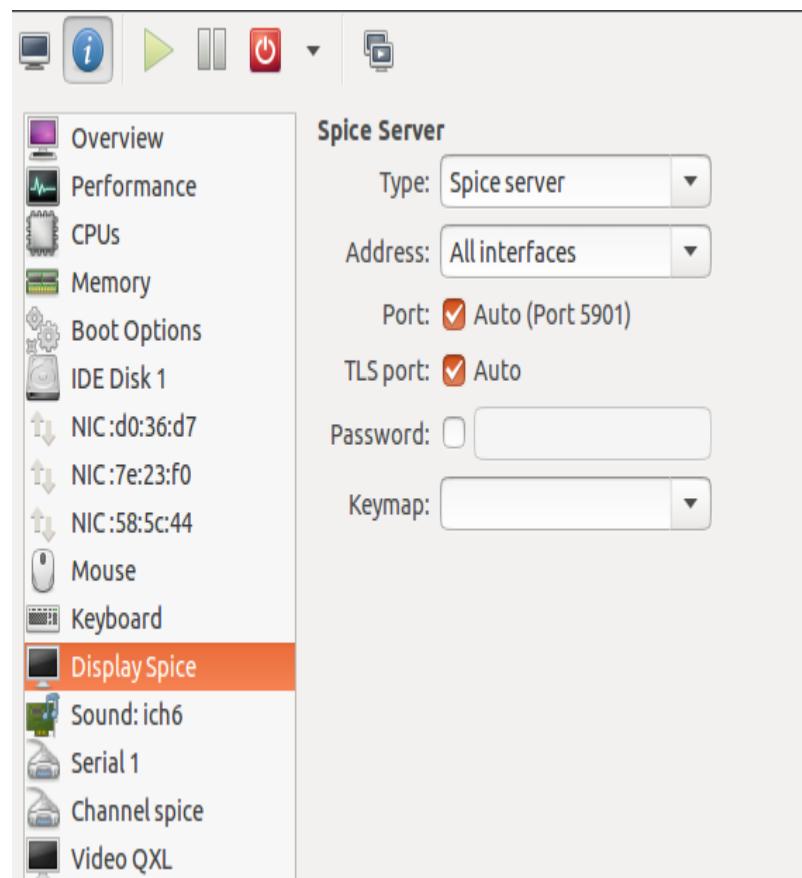
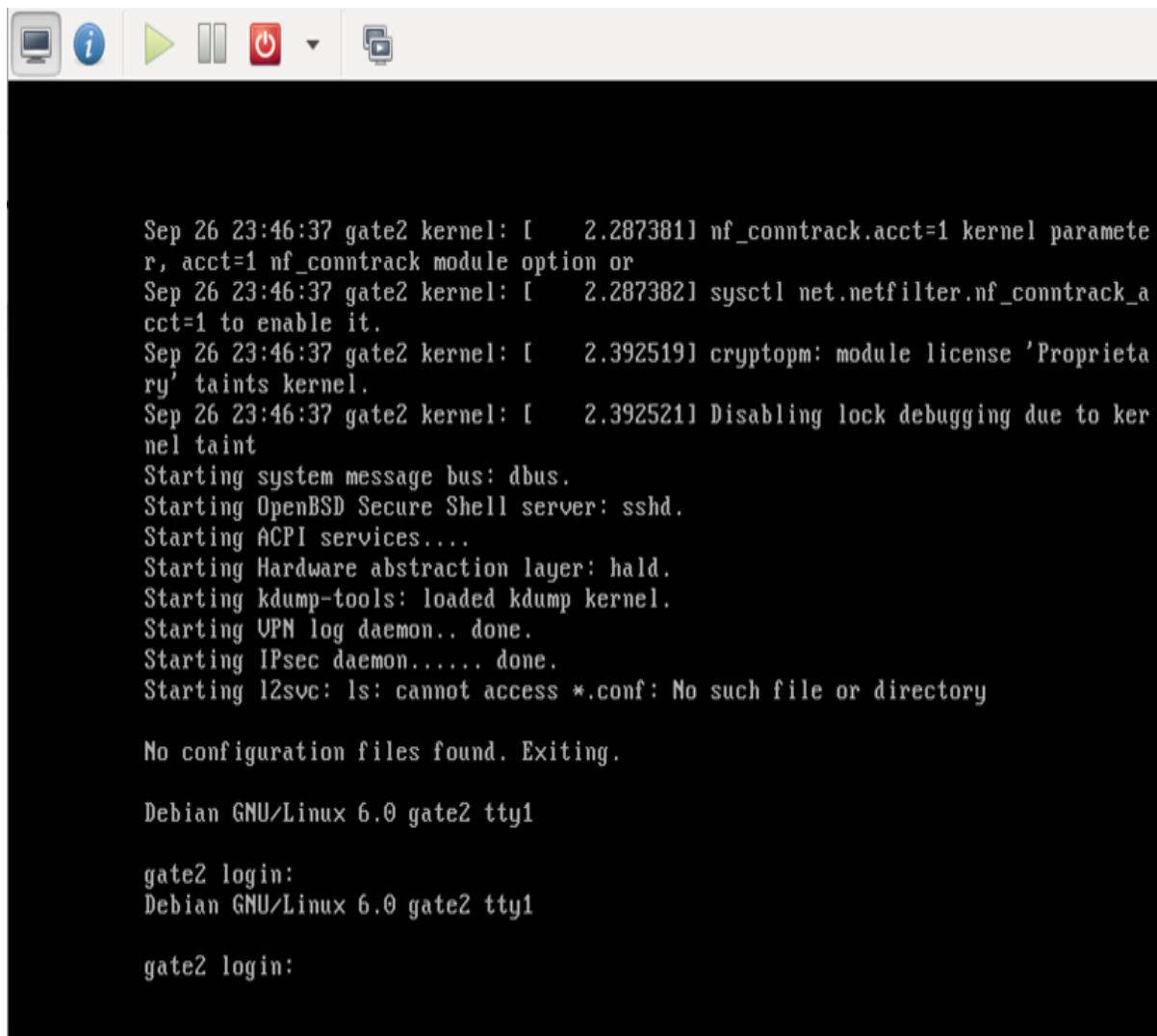


Рисунок 59

11. Включить машину и убедиться, что на виртуальном мониторе появилась загрузка операционной системы.



```

Sep 26 23:46:37 gate2 kernel: [    2.287381] nf_conntrack.acct=1 kernel parameter, acct=1 nf_conntrack module option or
Sep 26 23:46:37 gate2 kernel: [    2.287382] sysctl net.netfilter.nf_conntrack_acct=1 to enable it.
Sep 26 23:46:37 gate2 kernel: [    2.392519] cryptom: module license 'Proprietary' taints kernel.
Sep 26 23:46:37 gate2 kernel: [    2.392521] Disabling lock debugging due to kernel taint
Starting system message bus: dbus.
Starting OpenBSD Secure Shell server: sshd.
Starting ACPI services.....
Starting Hardware abstraction layer: hal.
Starting kdump-tools: loaded kdump kernel.
Starting VPN log daemon.. done.
Starting IPsec daemon..... done.
Starting l2svc: ls: cannot access *.conf: No such file or directory

No configuration files found. Exiting.

Debian GNU/Linux 6.0 gate2 tty1

gate2 login:
Debian GNU/Linux 6.0 gate2 tty1

gate2 login:

```

Рисунок 60

12. Для соединения виртуальной машины с EcoRouter используются виртуальные порты. На маршрутизаторе необходимо создать виртуальный порт командой конфигурационного режима **port virt.0**. Данный порт присоединить к одной из виртуальных сетей, созданных через virt-manager. Тогда интерфейс виртуальной машины и виртуальный порт маршрутизатора будут связаны через виртуальную сеть. После этого с данным портом можно работать как с обычным портом маршрутизатора. Например, можно настроить поток, который будет на уровне L2 связывать реальный порт маршрутизатора и виртуальный, тем самым все пакеты виртуальной машины будут проходить через реальный порт маршрутизатора.

Пример:

Конфигурирование виртуального порта.

```

ecorouter#conf t
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#port virt.0
ecorouter(config-port-virt)#service-instance virt0
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect port gel

```

Конфигурирование внешнего порта EcoRouter.

```
ecorouter#conf t
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#port gel1
ecorouter(config-port-virt)#service-instance gel1
ecorouter(config-service-instance)#encapsulation untagged
ecorouter(config-service-instance)#connect port virt.0
```

После указанных настроек в конфигурации маршрутизатора появятся следующие записи.

```
ecorouter#show running-config
!
...
!
port gel1
lacp-priority 32767
mtu 9728
service-instance gel1
  encapsulation untagged
!
...
!
port virt.0
  virtual-network vm uplink
  service-instance virt0
    encapsulation untagged
!
...
!
flow port gel1 service-instance gel1 port virt.0
!
flow port virt.0 service-instance virt0 port gel1
!
end
```

Для того, чтобы проверить правильность настройки соединения между внешним и виртуальным портом EcoRouter необходимо ввести команду административного режима **show virtual-network vm**.

```
ecorouter#show virtual-network vm
Virtual network uplink
bridge virbr1
port virt.0
```

13. Далее все настройки IP-адресации будут производиться на виртуальной машине.

34 Логирование и отладка

34.1 Логирование

В системе EcoRouter ведется запись обо всех происходящих событиях (выполненных операциях, изменениях конфигурации) - логирование. По умолчанию журнал событий (лог) ведется на самом устройстве.

Сообщения о событиях пишутся в двух форматах, описанных ниже.

Формат системных сообщений - действий, производимых сервисами (демонами) системы:

><DATE> <TIME> [VERBOSE] [SERVICE] <MESSAGE>

Формат сообщений об операциях, производимых пользователями (аккаунтинга):

<DATE> <TIME> [VERBOSE] [IMISH] AUDIT [USER] <MESSAGE>

Параметры приведенной условной записи форматов сообщений описаны в таблице ниже.

Таблица 124

Параметр	Описание
DATE	дата события в формате ГГГГ-ММ-ДД
TIME	время события в формате ЧЧ:ММ:СС.CCCCCC
VERBOSE	уровень события: FATAL – критические сообщения, ERROR – ошибки, WARN – предупреждения, INFO – информация
SERVICE	системный сервис (демон)
MESSAGE	сообщение о событии
USER	пользователь EcoRouter, который выполнил операцию

Для просмотра и записи журнала в файл используется команда административного режима **show log**.

Общий синтаксис команды: **show log (all |) (excessive |) (lines <NUM> |) (follow |reverse)**). Как и для других команд группы **show**, здесь также доступны модификаторы.

Чтобы отправить вывод команды в указанный файл, необходимо добавить к команде **show log** модификатор **| redirect <FILE>** или его краткую форму:

```
ecorouter#show log > Text1.log
```

Команда **show log** без параметров выводит на консоль все сообщения из системного журнала с момента загрузки устройства.

```
ecorouter#show log
>2016-10-26 13:55:28.490128 [info] [ecolog] writer thread started
>2016-10-26 13:55:28.490128 [info] [ecolog] reader thread started
>2016-10-26 13:55:28.490128 [info] [ecolog] listener thread started
>2016-10-26 13:55:28.490128 [info] [ecolog] watchdog thread started
>2016-10-26 13:55:28.490128 [info] [ecolog] Ecolog v1.0 connection
request[0]: 1
>2016-10-26 13:55:28.490128 [info] [ecolog] Ecolog v1.0 connection
request[0]: OK
```

```
>2016-10-26 13:55:28.490128 [info] [ecolog] [0] reader thread started
>2016-10-26 13:55:28.490128 [info] [ecobus] reader thread started
>2016-10-26 13:55:28.490128 [info] [ecobus] listener thread started
>2016-10-26 13:55:28.490128 [info] [ecobus] watchdog thread started
...

```

Команда **show log** с параметром **all** выводит на консоль все сообщения из *journalctl*.

Команда **show log** с параметром **excessive** выводит на консоль сообщения из системного журнала с дополнительной информацией о файле, функции и строке исходного файла.

```
ecorouter#show log excessive
>2016-10-27 12:25:10.571110 [info] [ecolog]
[src/writer.c:263,ecolog_writer_thread_proc] writer thread started
>2016-10-27 12:25:10.571110 [info] [ecolog]
[src/reader.c:295,ecolog_reader_thread_proc] reader thread started
>2016-10-27 12:25:10.571110 [info] [ecolog]
[src/listener.c:380,ecolog_listener_thread_proc] listener thread started
>2016-10-27 12:25:10.571110 [info] [ecolog]
[src/watchdog.c:197,ecolog_watchdog_thread_proc] watchdog thread started
>2016-10-27 12:25:12.571112 [info] [ecolog]
[src/listener.c:212,ecolog_listener_accept] Ecolog v1.0 connection
request[2]: 1
>2016-10-27 12:25:12.571112 [info] [ecolog]
[src/listener.c:225,ecolog_listener_accept] Ecolog v1.0 connection
request[2]: OK
>2016-10-27 12:25:12.571112 [info] [ecolog]
[src/reader.c:155,ecolog_reader_session_thread_proc] [2] reader thread
started
>2016-10-27 12:25:12.571112 [info] [IMI] [log.c:311,openzlog] trace
started
>2016-10-27 12:25:12.571112 [info] [IMI] [imi_ercp.c:488,imi_ercp_init]
-> imi_ercp_init []
>2016-10-27 12:25:12.571112 [info] [IMI]
[imi_ercp.c:750,imi_ercp_platform_init] -> imi_ercp_platform_init []
>2016-10-27 12:25:12.571112 [info] [IMI]
[imi_ercp_snmp.c:318,imi_ercp_snmp_init] -> imi_ercp_snmp_init
[snmp_config 0x00000000]
>2016-10-27 12:25:12.571112 [info] [IMI]
[imi_ercp_snmp.c:382,imi_ercp_snmp_init] <- imi_ercp_snmp_init: 0x0
...

```

Команда **show log** с параметром **lines <NUM>** выводит на консоль несколько последних сообщений, где <NUM> - количество сообщений.

```
ecorouter#show log lines 10
>2016-10-27 12:25:29.571129 [info] [OSPF] OSPFd (3.2.1) starts
>2016-10-27 12:25:29.571129 [info] [IMI] imi_server_send_config called
(PM 4)
>2016-10-27 12:25:29.571129 [info] [IMI] imi_server_send_config called
(PM 44)
>2016-10-27 12:25:29.571129 [info] [BGP] BGPd 3.2.1 starting: vty@2605,
bgp@179
>2016-10-27 12:25:29.571129 [info] [IMI] imi_server_send_config called
(PM 44)
>2016-10-27 12:25:30.571130 [info] [ecolog] Ecolog v1.0 connection
request[11]: 1

```

```
>2016-10-27 12:25:30.571130 [info] [ecolog] Ecolog v1.0 connection
request[11]: OK
>2016-10-27 12:25:30.571130 [info] [ecolog] [11] reader thread started
>2016-10-27 12:25:30.571130 [info] [PIM] trace started
>2016-10-27 12:25:30.571130 [info] [IMI] imi_server_send_config called
(PM 11)
```

Команда **show log** с параметром **follow** выводит на консоль непрерывный поток логов. Для непрерывного просмотра логов необходимо отключить pager: **show log follow | nopager**.

Команда **show log** с параметром **reverse** выводит на консоль поток логов в обратном порядке.

Можно задать несколько параметров и модификатор одновременно.

```
ecorouter#show log excessive lines 2
>2016-10-27 14:14:20.577660 [info] [ecobus]
[src/listener.c:351,ecobus_listener_accept] Ecobus v1.0 connection
request[7109]: 0/2/0
>2016-10-27 14:14:20.577660 [info] [ecobus]
[src/listener.c:366,ecobus_listener_accept] Ecobus v1.0 connection
request[7109]: OK
```

Например, для того чтобы вывести только те сообщения, которые относятся к действиям пользователя, необходимо ввести команду:

```
ecorouter#show log all | include IMISH
2016-10-27 12:25:43.571143 [info] [IMISH] AUDIT Logged in user
2016-10-27 12:25:43.571143 [info] [IMISH] AUDIT [admin] logged in
>2016-10-27 12:25:43.571143 [info] [IMISH-1648] trace started
2016-10-27 12:25:46.571146 [info] [IMISH] AUDIT ER user
2016-10-27 12:25:46.571146 [info] [IMISH] AUDIT [admin] logged in
2016-10-27 12:25:48.571148 [info] [IMISH] AUDIT [admin] en
2016-10-27 12:26:29.571189 [info] [IMISH] AUDIT [admin] terminal monitor
2016-10-27 12:26:47.571207 [info] [IMISH] AUDIT [admin] conf t
2016-10-27 12:26:58.571218 [info] [IMISH] AUDIT [admin] port te0
2016-10-27 12:28:11.571291 [info] [IMISH] AUDIT [admin]
2016-10-27 12:28:42.571322 [info] [IMISH] AUDIT [admin] service-
instance 100
2016-10-27 12:29:02.571342 [info] [IMISH] AUDIT [admin] ex
2016-10-27 12:29:05.571345 [info] [IMISH] AUDIT [admin] ex
```

Для дополнительного контроля за производимыми действиями предусмотрена возможность вывода сообщений логов на консоль в режиме реального времени.

Для включения данной функции используется команда административного режима **terminal monitor**. Для отключения вывода сообщений на консоль используется команда административного режима **no terminal monitor**.

34.2 Включение/выключение отладки

Для каждого компонента системы действуют отладочные команды, описанные в этом разделе.

Для включения отладки отдельных подсистем используются команда **debug <SUBSYSTEM>**, где **SUBSYSTEM** - имя подсистемы. Данная команда доступна и в

административном, и в конфигурационном режиме. При использовании данной команды в конфигурационном режиме она будет записана в конфигурацию маршрутизатора.

Кроме подсистем можно включить отладку для отдельных опций, например, **debug nsm packet recv detail**.

В таблице ниже приведен список доступных подсистем и параметров данной команды.

Таблица 125

Подсистема/ параметр команды	Описание	Режим
bgp	Border Gateway Protocol (BGP)	Административный и конфигурационный
bgp all	all debugging	
bgp dampening	BGP Dampening	
bgp events	BGP events	
bgp filters	BGP filters	
bgp fsm	BGP Finite State Machine	
bgp keepalives	BGP keepalives	
bgp mpls	BGP MPLS	
bgp nht	NHT message	
bgp nsm	NSM message	
bgp updates	BGP updates	
data-plane	Data Plane	Административный и конфигурационный
data-plane all	Enable all debugging	
data-plane bridge	Bridge subsystem	
data-plane cp	Control Plane subsystem	
data-plane fastpath	Fastpath subsystem	
data-plane general	General subsystem	
data-plane integrator	Integrator subsystem	
data-plane mac check	Mac check	
data-plane packetflow	Packetflow subsystem	
data-plane print	Print subsystem	
data-plane slowpath	Slowpath subsystem	
data-plane test	Test subsystem	
igmp	Internet Group Management Protocol (IGMP)	Административный и конфигурационный
igmp all	All IGMP debugging	
igmp decode	IGMP decode	
igmp encode	IGMP encode	
igmp events	IGMP events	
igmp fsm	IGMP FSM	
igmp tib	IGMP Tree-Info-Base (TIB)	
igmp vrf	VPN Routing/Forwarding instance	
isis	Intermediate System - Intermediate System (IS-IS)	Административный и конфигурационный
isis all	Enable all debugging	
isis authentication	IS-IS Authentication	
isis checksum	IS-IS Check-Sum	
isis events	IS-IS Events	
isis hello	IS-IS Hello Debug	

Подсистема/ параметр команды	Описание	Режим
isis ifsm	IS-IS Interface Finite State Machine	
isis local-updates	IS-IS Local Updates	
isis lsp	IS-IS Link State PDU	
isis mpls	Multi-Protocol Label Switching (MPLS)	
isis nfsm	IS-IS Neighbor Finite State Machine	
isis nsm	IS-IS NSM information	
isis pdu	IS-IS Protocol Data Unit	
isis protocol-errors	IS-IS Protocol Errors	
isis rib	IS-IS RIB information	
isis spf	IS-IS SPF Calculation	
ldp	Label Distribution Protocol (LDP)	Административный и конфигурационный
ldp advertise-labels	List IP access lists of advertise-labels	
ldp all	Enable all debugging	
ldp dsm	LDP Downstream SM	
ldp events	LDP events	
ldp fsm	LDP FSM	
ldp graceful-restart	LDP Graceful Restart Debugging	
ldp hexdump	LDP HEXDUMP	
ldp nsm	NSM messages	
ldp packet	LDP packet	
ldp qos	LDP QoS	
ldp rib	RIB messages	
ldp tsm	LDP Trunk SM	
ldp usm	LDP Upstream SM	
ldp vc	LDP VC Info	
mrib	Multicast Routing Information Base (MRIB)	Административный и конфигурационный
mrib all	All MRIB debugging	
mrib event	MRIB events	
mrib fib-msg	MRIB FIB messages	
mrib mrib-msg	MRIB MRIB IPC messages	
mrib mrt	MRIB route	
mrib mtrace	MRIB traceroute	
mrib mtrace-detail	MRIB traceroute detailed debugging	
mrib nsm-msg	MRIB NSM IPC messages	
mrib register-msg	MRIB PIM Register messages	
mrib stats	MRIB statistics	
mrib vif	MRIB interface	
mrib vrf	VPN Routing/Forwarding instance	
nsm	Network Service Module (NSM)	Административный и конфигурационный
nsm all	Enable all debugging	
nsm events	NSM events	
nsm packet	NSM packets	
ospf	Open Shortest Path First (OSPF)	Административный и конфигурационный
ospf all	Enable all debugging	
ospf database-timer	OSPF Database Timers	
ospf events	OSPF events information	
ospf graceful-restart	OSPF graceful-restart	
ospf ifsm	OSPF Interface State Machine	

Подсистема/ параметр команды	Описание	Режим
ospf lsa	OSPF Link State Advertisement	
ospf fsm	OSPF Neighbor State Machine	
ospf nsm	OSPF NSM information	
ospf packet	OSPF packets	
ospf policy	OSPF policy information	
ospf redistribute	OSPF redistribute information	
ospf retransmission	OSPF Debug retransmission information	
ospf rib	OSPF RIB information	
ospf route	OSPF route information	
pim	Protocol Independent Multicast (PIM)	Административный и конфигурационный
pim all	All PIM debugging	
pim events	PIM events	
pim mfc	PIM MFC updates	
pim mib	PIM mib	
pim mtrace	Mtrace messages	
pim nexthop	PIM nexthop	
pim nsm	NSM message	
pim packet	PIM packet	
pim state	PIM state	
pim timer	PIM timers	
pim vrf	VPN Routing/Forwarding instance	
rib	Routing Information Base (RIB)	Административный и конфигурационный
rib all	Enable all debugging	
rib events	RIB events	
rib nsm	NSM messages	
rib packet	RIB packets	
rib routing	Enable debugging for routing events	
security-profile	Security profile	Административный и конфигурационный
vrrp	Virtual Router Redundancy Protocol (VRRP)	Административный и конфигурационный
vrrp all	Enable all debugging	
vrrp events	VRRP events	
vrrp packet	VRRP packets	
aaa	AAA	Конфигурационный
aaa 1	critical	
aaa 2	error	
aaa 3	warning	
aaa 4	notice	
aaa 5	info	
aaa 6	debug	

Для отключения отладки используется команда **no debug <SUBSYSTEM>**, которая также работает в двух режимах. Предусмотрена также команда **un debug <SUBSYSTEM>**, однако, она работает только для подсистем и доступна только в административном режиме.

Для отключения отладки сразу для всех доступных подсистем используются команды **no debug all** и **undebug all**.

Для вывода на консоль информации об отладке подсистем используется команда административного режима **show debugging <SUBSYSTEM>**, где **SUBSYSTEM** - имя

подсистемы. Данная команда доступна для подсистем: **bgp, data-plane, igmp, isis, ldp, mrib, nsm, ospf, pim, rib, security-profile, vrrp**.

34.3 Архив логов

34.3.1 Просмотр архива логов

В EcoRouterOS в случае непредвиденных ситуаций собирается архив с логами и со всеми необходимыми для диагностики данными. Эти файлы имеют префикс "report" в названии. В название каждого такого архива также включается дата и точное время создания. Все отчеты хранятся локально на маршрутизаторе. Для их просмотра следует воспользоваться командой **show reports**. В результате ее выполнения выводится список архивов логов с указанием размеров архивов и даты и времени их создания.

```
ecorouter#show reports
report-20171107T143644UTC-3.2.3.9.11254-develop-68fb7f7.tar.xz: 181 KB
2017-10-07 14:36:45
report-20171107T143606UTC-3.2.3.9.11254-develop-68fb7f7.tar.xz: 174 KB
2017-10-07 14:36:07
```

34.3.2 Удаление архива логов

Ненужные или старые архивы можно удалить при помощи команды **delete report <REPORT_NAME>**, где <REPORT_NAME> - имя удаляемого архива. Для удаления всех архивов следует использовать команду **delete report all**.

```
ecorouter#show reports
report-20171107T143644UTC-3.2.3.9.11254-develop.tar.xz: 181 KB 2017-10-
07 14:36:45
report-20171107T143606UTC-3.2.3.9.11254-develop: 174 KB 2017-10-07
14:36:07
ecorouter#delete report report-20171107T143644UTC-3.2.3.9.11254-
develop.tar.xz
ecorouter#show reports
report-20171107T143606UTC-3.2.3.9.11254-develop.tar.xz: 174 KB 2017-10-
07 14:36:07
ecorouter#delete report all
ecorouter#show reports
No reports found!
ecorouter#
```

34.3.3 Копирование архива логов на внешний сервер

При необходимости архив логов можно скопировать на внешние FTP/TFTP-сервера. Общий вид команды для копирования следующий:

```
copy report {ftp | tftp} <REPORT_NAME> <URL>[<NEW_FILENAME>] {mgmt | vr
default | vr <VRNAME>}
```

Здесь <REPORT_NAME> - имя копируемого архива логов, <URL> - адрес сервера с указанием имени пользователя и пароля, <NEW_FILENAME> - новое имя файла архива логов (если возникла необходимость сохранить его на сервере под исходным именем, отличным от исходного).

Различные варианты применения команды **copy report** представлены в таблице.

Таблица 126

Команда	Описание
copy report ftp REPORT_NAME ftp://user:password@xxx.xxx.xxx.xxx/ mgmt	Архив логов с именем REPORT_NAME будет выгружен на FTP-сервер, FTP-сервер доступен через менеджмент-порт (mgmt)
copy report ftp REPORT_NAME ftp://user:password@xxx.xxx.xxx.xxx/filename vr default	Архив логов с именем REPORT_NAME будет выгружен на FTP-сервер. Доступ к FTP-серверу осуществляется через интерфейс виртуального маршрутизатора, выбранного по умолчанию. Архив логов будет сохранен на сервере под именем filename
copy report tftp REPORT_NAME tftp://xxx.xxx.xxx.xxx/ vr vrname	Архив логов с именем REPORT_NAME будет выгружен на TFTP-сервер. Доступ к TFTP-серверу осуществляется через интерфейс виртуального маршрутизатора с именем vrname .
copy report tftp REPORT_NAME tftp://xxx.xxx.xxx.xxx/filename mgmt	Архив логов с именем REPORT_NAME будет выгружен на TFTP-сервер. Доступ к TFTP-серверу осуществляется через менеджмент-порт (mgmt). Архив логов будет сохранен на сервере под именем filename

ПРИЛОЖЕНИЕ А

Справочник команд

В таблице ниже представлен справочник по командам EcoRouter.

В таблице содержится описание команды, режим консоли, в котором данная команда доступна, роли, для которых команда доступна.

В столбце "Режим консоли" используются следующие обозначения:

Польз - пользовательский режим,

Админ - режим администрирования,

Конф - режим конфигурации.

Команды, доступные только для роли **admin** и запрещенные для любых других ролей, отмечены буквой **d** (access denied).

Таблица 127

Команда	Описание	Режим консоли	Роли		
			admin	noc	helpdesk
bgp	Border Gateway Protocol (BGP)	Польз	+		
clear	Reset functions	Польз	+		
crypto	Security specific commands	Польз			
debug	Debugging functions (see also 'undebug')	Польз	+		
disable	Turn off privileged mode command	Польз	+	+	+
enable	Turn on privileged mode command	Польз	+	+	+
exit	End current mode and down to previous mode	Польз	+	+	+
help	Description of the interactive help system	Польз	+	+	+
logout	Exit from the EXEC	Польз	+	+	+
no	Negate a command or set its defaults	Польз	+	+	+
ping	Send echo messages	Польз	+		
quit	Exit current mode and down to previous mode	Польз	+	+	
show access-group	Show access group	Польз	+	+	
show access-list	Show access list configuration	Польз	+	+	
show banner motd	Show current motd banner message	Польз	+	+	
show bgp	Border Gateway Protocol (BGP)	Польз	+	+	
show bridge	Bridge status and configuration	Польз	+	+	
show bridge mac-table	Bridge mac-table	Польз	+	+	
show cli	Show CLI tree of current mode	Польз	+	+	
show clns	Connectionless-Mode Network Service (CLNS)	Польз	+	+	
show controller	Controller status and configuration	Польз	+	+	
show counters	Counters	Польз	+	+	
show debugging	Debugging information outputs	Польз	+	+	
show dhcp-profile	DHCP profile configuration	Польз	+	+	
show filter-map	Filterring rules	Польз	+	+	

Команда	Описание	Режим консоли	Роли		
			admin	noc	helpdesk
bgp	Border Gateway Protocol (BGP)	Польз	+		
show flow-export-profile	Flow export profile configuration	Польз	+	+	
show hostname	Hostname	Польз	+	+	
show hw	EcoRouter platform	Польз	+	+	
show interface	Interface configuration	Польз	+	+	
show ip	Internet Protocol (IP)	Польз	+	+	
show isis	Intermediate System - Intermediate System (IS-IS)	Польз	+	+	
show lacp	LACP	Польз	+	+	
show ldp	Label Distribution Protocol (LDP)	Польз	+	+	
show list	Show command lists	Польз	+	+	
show users localdb	Display users database information	Польз	+	d	d
show log	Display log	Польз	+	+	
show mirror-session	Mirror session status and configuration	Польз	+	+	
show mpls	Show MPLS specific data	Польз	+	+	
show platform	Show platform information	Польз	+	+	
show port	Port status and configuration	Польз	+	+	
show pppoe	Point-to-Point over Ethernet (PPPoE)	Польз	+	+	
show privilege	Show current privilege level	Польз	+	+	
show reports	Show existing reports	Польз	+	+	
show role	Display information about role	Польз	+	d	d
show running-config	Current Operating configuration	Польз	+	+	
show security-profile	Security profile	Польз	+	+	
show traffic-classifier	Traffic classifier status and configuration	Польз	+	+	
show traffic-limiter	Traffic limiter status and configuration	Польз	+	+	
show traffic-scheduler	Traffic scheduler status and configuration	Польз	+	+	
show transceiver	Transceiver information	Польз	+	+	
show uptime	Show system uptime	Польз	+	+	
show users connected	Display information about terminal lines	Польз	+	+	
show version	Display version	Польз	+	+	
show virtual-router	Virtual Router information	Польз	+	+	
show vrrp	VRRP information	Польз	+	+	
terminal	Set terminal line parameters	Польз	+	+	+
undebbug	Disable debugging functions (see also 'debug')	Польз	+	+	+
virtual-container	Virtual container settings	Польз			
boot	Boot options of EcoRouterOS	Админ	+		
clear	Reset functions	Админ	+		
configure terminal	Enter configuration mode	Админ	+		
copy	Copy from one place to another	Админ	+		
copy report	Upload report to remote server	Админ	+		
crypto ca export	Certification Authority settings	Админ	+		
crypto certificate export	Display security information	Админ	+		
crypto key export	User private key	Админ	+		
debug	Debugging functions (see also 'undebbug')	Админ	+		

Команда	Описание	Режим консоли	Роли		
			admin	нос	helpdesk
bgp	Border Gateway Protocol (BGP)	Польз	+		
delete report	Delete existing reports	Админ	+		
develop	Debug command	Админ	+		
disable	Turn off privileged mode command	Админ	+		
enable	Turn on privileged mode command	Админ	+		
exit	End current mode and down to previous mode	Админ	+	+	+
faults	Fault management command	Админ	+		
help	Description of the interactive help system	Админ	+	+	+
image	Image of EcoRouterOS	Админ	+		
login	Login as a particular user	Админ	+	+	+
logout	Exit from the EXEC	Админ	+	+	+
mstat	show statistics after multiple multicast traceroutes	Админ	+	+	+
mtrace	Trace multicast path from source to destination	Админ	+	+	+
no	Negate a command or set its defaults	Админ	+		
ping	Send echo messages	Админ	+	+	+
poweroff	Turn system off	Админ	+		
quit	Exit current mode and down to previous mode	Админ	+	+	+
reload	Halt and perform a cold restart	Админ	+		
restart	Restart process	Админ	+		
show access-group	Access group	Админ	+	+	
show access-list	Access list configuration	Админ	+	+	
show arp	ARP table	Админ	+	+	
show banner motd	Show current motd banner message	Админ	+	+	
show bgp	Border Gateway Protocol (BGP)	Админ	+	+	
show boot	Boot configuration of EcoRouterOS	Админ	+	+	
show bridge	Bridge status and configuration	Админ	+	+	
show bridge mac-table	Bridge mac-table	Админ	+	+	
show cli	Show CLI tree of current mode	Админ	+	+	
show clns	Connectionless-Mode Network Service (CLNS)	Админ	+	+	
show controller	Controller status and configuration	Админ	+	+	
show counters	Counters	Админ	+	+	
show debugging	Debugging functions (see also 'undebug')	Админ	+	+	
show develop	Debug output	Админ	+	+	
show dhcp-profile	DHCP profile configuration	Админ	+	+	
show faults	Show recorded faults	Админ	+	+	
show filter-map	Filtering rules	Админ	+	+	
show flow-export-profile	Flow export profile configuration	Админ	+	+	
show hostname	Hostname	Админ	+	+	
show hw	EcoRouter platform	Админ	+	+	
show images	Images that can be used to upgrade EcoRouterOS	Админ	+	+	
show interface	Interface configuration	Админ	+	+	
show ip	Internet Protocol (IP)	Админ	+	+	
show isis	Intermediate System - Intermediate System (IS-IS)	Админ	+	+	
show lacp	LACP	Админ	+	+	
show ldp	Label Distribution Protocol (LDP)	Админ	+	+	
show list	Show command lists	Админ	+	+	

Команда	Описание	Режим консоли	Роли		
			admin	noc	helpdesk
bgp	Border Gateway Protocol (BGP)	Польз	+		
show users localdb	Display users database information	Админ	+	+	
show log	Display log	Админ	+	+	
show mirror-session	Mirror session status and configuration	Админ	+	+	
show mpls	Show MPLS specific data	Админ	+	+	
show mrib	MRIB	Админ	+	+	
show nsm	NSM	Админ	+	+	
show ntp	Configuration NTP	Админ	+	+	
show platform	Show platform information	Админ	+	+	
show port	Port status and configuration	Админ	+	+	
show pppoe	Point-to-Point over Ethernet (PPPoE)	Админ	+	+	
show privilege	Show current privilege level	Админ	+	+	
show process	Process	Админ	+	+	
show process-group	Process	Админ	+	+	
show proc-names	Show process names	Админ	+	+	
show reports	Show existing reports	Админ	+	+	
show rib	RIB	Админ	+	+	
show role	Display information about role	Админ	+	+	
show route-map	Route-map information	Админ	+	+	
show router-id	Router ID	Админ	+	+	
show routing	Display routing information	Админ	+	+	
show running-config	Current Operating configuration	Админ	+	+	
show security-profile	Security profile	Админ	+	+	
show snmp	Display snmp settings	Админ	+	+	
show startup-config	Contents of startup configuration	Админ	+	+	
show tech-support	Show router technical information	Админ	+	+	
show tech-support-vr	Show technical information of non privileged	Админ	+	+	
show traffic-classifier	Traffic classifier status and configuration	Админ	+	+	
show traffic-limiter	Traffic limiter status and configuration	Админ	+	+	
show traffic-scheduler	Traffic scheduler status and configuration	Админ	+	+	
show transceiver	Transceiver information	Админ	+	+	
show uptime	Show system uptime	Админ	+	+	
show users connected	Display information about terminal lines	Админ	+	+	
show version	Display version	Админ	+	+	
show virtual-network	Virtual network	Админ	+	+	
show virtual-router	Virtual Router information	Админ	+	+	
show vrrp	VRRP information	Админ	+	+	
start-shell	Start shell	Админ	+		
telnet	Open a telnet connection	Админ	+	+	+
terminal	Set terminal line parameters	Админ	+	+	+
traceroute	Trace route to destination	Админ	+	+	+

Команда	Описание	Режим консоли	Роли		
			admin	noc	helpdesk
bgp	Border Gateway Protocol (BGP)	Польз	+		
undebug	Disable debugging functions (see also 'debug')	Админ	+		
virtual-container join-swarm	Virtual container settings. Join a swarm as a node	Админ	+		
write	Write running configuration to memory, file or terminal	Админ	+		
aaa	Authentication Authorization Accounting	Конф	+		
aaa-profile	AAA server-profile configuration	Конф	+		
arp	Address Resolution Protocol (ARP)	Конф	+		
bandwidth	Bandwidth configuration	Конф	+		
banner	Define a login banner	Конф	+		
bgp	Border Gateway Protocol (BGP)	Конф	+		
bridge	Bridge configuration	Конф	+		
class-map	Class-map configuration	Конф	+		
controller	Controller configuration	Конф	+		
cvlan	Configure C-VLAN parameters	Конф	+		
debug	Debugging functions (see also 'undebug')	Конф	+		
debug dns client	Display DNS debugging messages	Конф	+		
dhcp-profile	Select a DHCP profile to configure	Конф	+		
do	To run exec commands in config mode	Конф	+		
enable container	Enable containerization	Конф	+		
enable password	Assign the privileged level password	Конф	+		
enable vm	Enable libvirt/kvm virtualization	Конф	+		
exit	End current mode and down to previous mode	Конф	+		
fib	FIB information	Конф	+		
filter-map ethernet	Filter by L2 header	Конф	+		
filter-map ipv4	Filter by L3 header	Конф	+		
flow-export-profile	Flow export profile configuration	Конф	+		
help	Description of the interactive help system	Конф	+		
hostname	Set system's network name	Конф	+		
hw	EcoRouter platform	Конф	+		
interface	Select an interface to configure	Конф	+		
ip	Internet Protocol (IP)	Конф	+		
IP domain-list	Define a list of default domain names used to complete unqualified host names	Конф	+		
IP domain-lookup	Enable DNS host name-to-address translation	Конф	+		
IP domain-name	Set the default domain name used to complete unqualified host names	Конф	+		
IP host	Define static hostname-to-address mappings in DNS	Конф	+		
IP name-server	Add 1-3 DNS server addresses that are used to translate hostnames to IP addresses	Конф	+		
isis	Intermediate System - Intermediate System (IS-IS)	Конф	+		
key	Authentication key management	Конф	+		
l2vpn-vpws	Configure MPLS specific attributes	Конф	+		
line	Configure a terminal line	Конф	+		
mac-access-list	Add an access list entry	Конф	+		
max-fib-routes	Set maximum fib routes number	Конф	+		
maximum-paths	Set multipath numbers installed to FIB	Конф	+		
max-static-routes	Set maximum static routes number	Конф	+		
mirror-session	Select a mirror session to configure	Конф	+		
mpls	Configure MPLS specific attributes	Конф	+		

Команда	Описание	Режим консоли	Роли		
			admin	noc	helpdesk
bgp	Border Gateway Protocol (BGP)	Польз	+		
no	Negate a command or set its defaults	Конф	+		
ntp	Configuration NTP	Конф	+		
oep	Configure OVC endpoint map	Конф	+		
ospf	Open Shortest Path First (OSPF)	Конф	+		
platform sensor alarm	Enable sensor alarm notifications	Конф	+		
policy-filter-list	Add an access list entry	Конф	+		
port	Port configuration	Конф	+		
role	User role management	Конф	+		
route-map	Create route-map or enter route-map command mode	Конф	+		
router	Enable a routing process	Конф	+		
rsyslog	rsyslog options	Конф	+		
security	Set security profile	Конф	+		
security-profile	Security profile	Конф	+		
service	Setup miscellaneous service	Конф	+		
service-policy	Service-policy configuration	Конф	+		
show cli	Show CLI tree of current mode	Конф	+		
show list	Show command lists	Конф	+		
show running-config	Current Operating configuration	Конф	+		
show hosts	Display the DNS name servers and domain names	Конф	+	+	+
show running-config dns	Show the DNS settings the running configuration	Конф	+	+	+
snmp	snmp	Конф	+		
snmp-server	Configure snmp server	Конф	+		
traffic-class	Select a traffic class to configure	Конф	+		
traffic-classifier	Select a traffic classifier to configure	Конф	+		
traffic-limiter	Select a traffic limiter to configure	Конф	+		
traffic-profile	Select a traffic profile to configure	Конф	+		
traffic-scheduler	Select a traffic scheduler to configure	Конф	+		
username	Establish User Name Authentication	Конф	+		
virtual-router	Virtual-router configuration	Конф	+		
vlan	Configure VLAN parameters	Конф	+		
vrrp	VRRP configuration	Конф	+		

<http://rdp.ru>
 Телефон: +7(495)204-9-204
 E-Mail: sales@rdp.ru

