

EcoRouter OS

**Руководство по установке
и конфигурированию**

Редакция: январь 2026 г.



Оглавление

Оглавление	2
Введение	10
Условные обозначения	12
Список сокращений	14
Подключение к EcoRouter	24
1 Оборудование	25
1.1 Сетевые порты	28
1.2 Сетевые карты	29
1.3 SFP модули	31
1.4 Лицензионные программные расширения	33
1.5 Блоки питания, вентиляторы и сенсоры	33
2 Общие сведения о работе с CLI	37
2.1 Подключение к EcoRouter	37
2.2 Виртуальные интерфейсы удалённого доступа	39
2.3 Подключение к другим сетевым устройствам	41
2.4 Интерфейс командной строки	42
2.5 Режим администрирования	44
2.6 Работа с конфигурацией	44
2.7 Подсказки и горячие клавиши	45
2.8 Команды группы show	47
2.9 Утилита ping	48
2.10 Утилита traceroute	51
2.11 Приветствие и баннер	53
3 Локальная аутентификация, авторизация и аккаунтинг	55
3.1 Вход в систему	55
3.2 Пользователи и их роли	56
3.3 Настройка учётных записей пользователей	58
3.4 Команды группы show	60
3.5 Локальный аккаунтинг	60

3.6 Служебные пользователи	61
3.7 Приоритет способов авторизации и аутентификации	61
3.8 Уровни логирования служб AAA	61
3.9 Внешние процедуры AAA по протоколу TACACS+	62
3.10 Внешние процедуры AAA по протоколу RADIUS	69
3.11 Профили безопасности	71
4 Виды портов и интерфейсов	77
4.1 Порт	77
4.2 Агрегирование каналов	78
4.3 Интерфейс	78
4.4 ICMP параметры интерфейса	80
4.5 Максимальный передаваемый элемент данных (MTU)	81
4.6 Интерфейс loopback	82
4.7 Bridge domain	83
4.8 Интерфейс bridge domain	84
4.9 Service Instance	84
4.10 Команды просмотра состояний интерфейсов	85
4.11 Команды просмотра SFP модулей	87
5 Сервисные интерфейсы	91
5.1 Виды инкапсуляции	91
5.2 Операции над метками	92
5.3 Операции над метками в сервисных интерфейсах	94
5.4 Просмотр настроек сервисных интерфейсов	101
6 Бриджинг с поддержкой L3	104
6.1 Настройка	106
6.2 Создание BDI	107
6.3 Команды просмотра	107
7 Экспорт и импорт конфигурации	110
7.1 Подключение к серверу	110
7.2 Путь копирования	110
7.3 Архив конфигурации	111
7.4 Выбор интерфейса	111

7.5 Экспорт конфигурации.....	112
7.6 Импорт конфигурации	113
7.7 Пример резервного копирования конфигурации	114
8 Операции с прошивкой	116
8.1 Скачивание образа прошивки	116
8.2 Установка образа прошивки	118
8.3 Действия после установки образа прошивки.....	120
8.4 Удаление образа прошивки	122
8.5 Выгрузка образа прошивки	122
8.6 Проверка целостности системных файлов	123
8.7 Сброс до заводской версии ПО	123
8.8 Мягкий сброс	124
9 ARP	126
10 LLDP	128
11 Dynamic Host Configuration Protocol.....	131
11.1 Работа DHCP.....	133
11.2 Настройка DHCP-функционала на EcoRouter	141
11.3 Настройка DHCP-сервера.....	144
12 VRRP	153
12.1 Базовая настройка	153
12.2 Дополнительные функции	154
12.3 Поддерживаемые версии протокола.....	155
12.4 Пример конфигурации	156
12.5 Известные особенности взаимодействия EcoRouter с оборудованием других производителей	159
13 IP SLA	161
14 Агрегирование каналов	168
14.1 Вычисление хэш-функции	168
14.2 LACP	169
14.3 ECMP.....	173
14.4 Настройка Link aggregation	173
15 Настройки зеркалирования SPAN/RSPAN	178



15.1 Mirror-session	178
15.2 Пример настройки зеркалирования	180
15.3 Приостановка зеркалирования	185
15.4 Просмотр правил зеркалирования	186
16 SNMP	188
16.1 Запуск и остановка сервиса SNMP	188
16.2 Настройка SNMP community	189
16.3 Настройка представлений (SNMP views)	190
16.4 Настройка отправки асинхронных сообщений	191
16.5 Внешняя установка параметров	192
16.6 SNMPv3	193
17 NTP	198
17.1 Базовая настройка	199
17.2 Команды просмотра NTP	200
18 PTP	202
18.1 Команды просмотра	205
19 CoPP	206
19.1 Команды просмотра	208
20 Маршрутизация Unicast	210
20.1 Введение в маршрутизацию	210
20.2 Настройка статических маршрутов	212
20.3 Протокол RIP	214
20.4 Протокол OSPF	219
20.5 Протокол IS-IS	235
20.6 Протокол BGP	242
20.7 Функция защиты BGP сессий	269
20.8 Настройка uRPF	272
21 Списки доступа	274
21.1 Policy-filter-list	274
21.2 Префиксные списки (prefix-list)	283
21.3 Filter-map	286
22 Карты маршрутов	308



22.1 Настройка карт маршрутов	308
22.2 Обработка записей в картах маршрутов	309
23 Настройка туннелирования	311
23.1 GRE	311
23.2 IP in IP	316
24 IPsec.....	320
24.1 Введение в IPsec	320
24.2 Статический IPsec.....	326
24.3 Настройка протокола IKEv2	332
24.4 Настройка GRE over IPsec.....	339
25 Встроенный NAT	348
25.1 NAT port forwarding.....	351
25.2 Пример конфигурации static source NAT	353
25.3 Пример конфигурации static source PAT	354
26 Коммутация по меткам MPLS	356
26.1 Настройка статического MPLS.....	357
26.2 LDP	361
26.3 Virtual Private Wire Service	371
26.4 Совместная работа BGP и MPLS	384
26.5 MPLS L3 VPN.....	390
26.6 Virtual Private LAN Service	414
26.7 RSVP-TE	430
27 BFD	452
27.1 Протокол BFD.....	452
27.2 Пример настройки single-hop BFD-OSPF	457
27.3 Настройка Multi-Hop BFD по BGP	459
28 Маршрутизация Multicast.....	461
28.1 IGMP	461
28.2 IGMP SSM Mapping.....	464
28.3 Proxy-IGMP.....	466
28.4 PIM-SM/SSM	468
28.5 PIM-DM и смешанный режим Sparse-Dense	475



29 BRAS	476
29.1 IPoE абоненты	477
29.2 Настройки PPPoE	492
29.3 Аутентификация, авторизация и аккаунтинг	504
29.4 Фильтрация и HTTP перенаправление	514
29.5 Удалённая аутентификация, авторизация и аккаунтинг	520
29.6 Таймеры абонентских сессий	528
29.7 Команды группы show для BRAS	528
29.8 Функционал ARP Proxu	535
29.9 Рекомендации и тонкости настройки	536
29.10 Логирование абонентских сессий	538
29.11 Общие сервисы.....	541
29.12 Удалённые абонентские сети в среде MPLS	543
30 L2TP	546
30.1 Установление управляющего туннеля L2TP.....	546
30.2 Подготовка пользовательской сессии PPP.....	547
30.3 Пользовательская сессия PPP	549
30.4 Передача данных.....	553
30.5 Настройка L2TP.....	554
30.6 Конфигурирование L2TP LNS.....	558
30.7 Конфигурирование L2TP LNS + LAC 1	564
30.8 Конфигурирование L2TP LNS + LAC 2.....	568
31 Экспорт данных о трафике.....	590
31.1 Пример настройки	592
31.2 Команды просмотра	593
32 QoS.....	595
32.1 Архитектура QoS	595
32.2 Классификация трафика.....	596
32.3 RED	600
32.4 Планировщик (Scheduler)	602
32.5 Счётчики.....	607
32.6 Ограничение скорости	609

32.7 Маркировка трафика	610
32.8 Перемаркировка трафика	612
32.9 Сервисные политики	613
32.10 Профиль трафика	616
32.11 Карты классов	617
32.12 Ограничение входящего трафика по классам	619
32.13 Пример классификации трафика	620
32.14 Пример управления трафиком	624
33 IPv6	630
33.1 Введение в IPv6	630
33.2 Маршрутизация IPv6	638
34 Виртуальные маршрутизаторы	680
34.1 Команды настройки виртуальных маршрутизаторов	680
34.2 Пример настройки виртуального маршрутизатора	682
34.3 Команды просмотра	683
35 Виртуальные машины и контейнеры	685
35.1 Общие сведения	685
35.2 Конфигурирование подключения интерфейсов виртуальной машины к EcoRouter	687
35.3 Конфигурирование доступа внешних средств управления контейнерами	687
35.4 Копирование виртуальных дисков	688
35.5 Распределение ядер между виртуальными машинами и data-plane	689
35.6 Подключение к виртуальной машине	689
35.7 Быстрая настройка виртуальных машин	692
35.8 Инфраструктура открытых ключей	700
36 Поток E1	704
36.1 Порты и каналы E1	704
36.2 Настройка Multilink PPP	709
37 Логирование и отладка	711
37.1 Локальное логирование	711
37.2 Включение и выключение отладки	714
37.3 Архив логов	720

37.4 Сниффинг.....	722
38 Справочник команд.....	724

Введение

Маршрутизатор EcoRouter создавался как программное решение для работы на универсальной x86 платформе. Преимуществами реализации является использование выделенных высокоскоростных драйверов и обработчиков для уровней управления и передачи данных.

Маршрутизатор использует собственные процессы плоскости управления. В качестве примера можно провести такие протоколы и приложения как BGP, OSPF, процессы сервера широкополосного доступа и другие.

Передача данных между физическими сетевыми адаптерами и операционной системой маршрутизатора производится напрямую с помощью драйверов Date Plane Development Kit, минуя обработку на уровне ядра Linux.

Подобным же образом, благодаря набору eBPF, производится высокоскоростная и безопасная обработка управляющих данных.

Драйверы EcoRouterOS управляют основными компонентами программно-аппаратной платформы, взаимодействующими с центральным процессором. Что позволяет передавать обработку сетевого трафика на одном порте маршрутизатора одновременно нескольким ядрам процессора.

Важной особенностью работы EcoRouter, является разгрузка центральных процессоров за счёт использования аппаратных возможностей сетевых адаптеров, включая расчёт контрольных сумм, приоритезацию, балансировку и другие возможности.



Работа с оперативной памятью (RAM) осуществляется с помощью средств прямого доступа к памяти DMA.

В настоящем руководстве описан порядок установки, первичной настройки и работы с маршрутизатором 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://ecorouter.ru/> или в службу технической поддержки.

Рекомендации по настройке, сопровождающиеся словами «ВНИМАНИЕ», «ВАЖНО», 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:

- Полужирный шрифт — названия элементов пользовательского интерфейса (команды, кнопки клавиатуры, символы консоли, рекомендуемые значения вводимых параметров), значения параметров команд.

Пример:

"Для изменения значения в контекстном режиме конфигурирования порта необходимо вызвать команду `l2cp-priority <NUM>`, где **NUM** — приоритет порта, изменяемый в пределах от 0 до 65535."

- Шрифт `Courier New` на светло-сером фоне — отдельные строки и вставки по ходу текста примеров кода, команд, вывода консоли. В примере выше: `l2cp-priority <NUM>`.
- Блок текста шрифтом `Courier New` на светло-сером фоне — блоки вводимых команд и вывода консоли.

Пример:

```
ecorouter#show counters l2cp
Port channel: ae.01
Port          LACPDU recv pkts    LACPDU sent pkts    Unknown recv pkts    Illegal
recv pkts
te1            0            1648            0            0
```

Условные обозначения при описании консоли:

- `< >` — значения параметров вводимые пользователем.
Пример: `<часть команды>?`.
- `[]` — необязательный элемент или набор необязательных элементов.
Требуется выбрать один из них. Пример: `<часть команды>[TAB]`.
- `{ }` — обязательный элемент или набор обязательных элементов.
Требуется выбрать один из них.
- `|` — Разделитель для взаимоисключающих элементов.
Пример: `ptp mode {transparent|boundary} {e2e|p2p} {ethernet|udp}`

Шаблоны:

- `A.B.C.D` — адрес IPv4.

- `x:x::x:x` — адрес IPv6.

Список сокращений

- AAA** — *Authentication, Authorization, Accounting* — аутентификация, авторизация, учёт;
- ABR** — *Area Border Router*, граничный маршрутизатор зоны;
- AC** — *Alternating Current*, переменный ток;
- ACK** — *Acknowledgement*, оповещение о получении сообщения;
- ACL** — *Access control list*, списки контроля доступа;
- ADV** — *Advertising Router*, маршрутизатор, который инициировал или сгенерировал LSA;
- AES** — *Advanced Encryption Standard*, расширенный стандарт шифрования;
- API** — *Application Programming Interface*, программный интерфейс приложения;
- ARP** — *Address Resolution Protocol*, протокол определения MAC-адреса;
- AS** — *Autonomous System*, совокупность сетей под одним административным управлением;
- ASBR** — *Autonomous System Boundary Router*, граничный маршрутизатор автономной системы;
- ASN** — *Autonomous System Number*, уникальный номер автономной системы (AS);
- AUTH** — *Authorization*, авторизация;
- BDI** — *Bridge Domain Interface*, логический интерфейс, обеспечивающий двунаправленный поток данных между L2 ширококестельным доменом и L3 маршрутизируемой сетью;
- BFD** — *Bidirectional Forwarding Detection*, обнаружение двунаправленной переадресации — протокол для быстрого обнаружения разрыва соединений;
- BGP** — *Border Gateway Protocol*, протокол граничного шлюза;
- BMC** — *Baseboard Management Controller*, управляющий контроллер в составе серверной материнской платы;
- BMI** — *Broadband Multiple Instance*, L3 интерфейс для обработки IPoE/PPP/MPLS трафика от абонентов;
- BNG** — *Broadband Network Gateway*, шлюз широкополосной сети, служит для подключения абонентов к широкополосной сети;
- BOOTP** — *Bootstrap Protocol*, сетевой протокол прикладного уровня, используемый для автоматического получения клиентом IP-адреса;
- BRAS** — *Broadband Remote Access Server*, маршрутизатор широкополосного удалённого доступа;

BSR — *Bootstrap Router*, протокол для управления распределением многоадресных потоков данных в сети;

BVI — *Bridge Virtual Interface*, виртуальный мостовой интерфейс;

CHAP — *Challenge Handshake Authentication Protocol*, протокол аутентификации с косвенным согласованием;

CIR — *Committed Information Rate*, гарантированная средняя скорость передачи данных;

CLI — *Command Line Interface*, интерфейс командной строки;

CLNS — *Connectionless Network Service*, сетевая служба без установления соединения;

CNTL — *CoNTroL*, клавиша — модификатор ввода в компьютерах архитектуры x86;

COM — *COMmunications port*, коммуникационный (последовательный) порт;

CP — *Control Plane*, управляющая плоскость - набор протоколов для передачи управляющих данных в сетях LTE;

CPU — *Central Processing Unit*, центральный процессор компьютера;

CSNP — *Complete Sequence Number PDU*, список всех состояний каналов (LSP) в базе данных состояний маршрутизатора;

DC — *Direct Current*, постоянный ток;

DDM — *Digital Diagnostics Monitoring*, функция цифрового контроля параметров производительности SFP трансивера;

DES — *Data EncrypTion Standard*, стандарт шифрования данных;

DF — *Do not Fragment*, флаг (бит) в IP-пакете, запрещающий фрагментацию пакета;

DHCP — *Dynamic Host Configuration Protocol*, протокол динамической настройки узла;

DNA — *Do No Age*, неустаревающий бит;

DNAT — *Destination Network Address Translation*, изменение адреса и порта назначения пакета;

DNS — *Domain Name System*, система доменных имён;

DORA — *Discover Offer Request Acknowledgement*, процесс получения IP-адреса в протоколе DHCP;

DOT1Q — Опция тегирование данных по стандарту IEEE 802.1Q;

DR — *Designated Router*, назначенный маршрутизатор;

DSCP—*Differentiated Services Code Point*, дифференцированный код услуг в сетях с управлением качеством обслуживания (Qos);



DST—*Destination*, пункт назначения;

EBGP—*External Border Gateway Protocol*, внешний протокол граничного шлюза;

ECMP — *Equal-Cost Multi-path routing*, стратегия маршрутизации, при которой пересылка пакетов в один пункт назначения может осуществляться по нескольким лучшим путям с равным приоритетом маршрутизации;

EGP — *Exterior Gateway Protocol*, протокол внешней (между автономными системами) маршрутизации;

ESP — *Encapsulating Security Payload*, протокол безопасной инкапсуляции полезной нагрузки в стандарте IPSec;

EVC—*Ethernet Virtual Connection*, технология виртуальных соединений, позволяющая организовать внутри физического коммутатора несколько широковещательных L2 доменов;

EXP — *Experimental*, поле заголовка MPLS кадра для определения качества обслуживания (QoS);

FCS — *Frame Check Sequence*, последовательность для проверки кадра;

FEC — *Forwarding Equivalence Class*, класс эквивалентности продвижения — все пакеты одного класса, передаются через MPLS-сеть по одному LSR;

FIB — *Forwarding Information Base*, таблица для ускоренной пересылки пакетов;

FIP — *Framed IP*, атрибут RADIUS, который указывает на IP-адрес, назначенный клиенту;

FSM — *Finite State Machine*, система с конечным числом состояний;

FTN — *FEC To Next hop*, таблица сопоставлений классов FEC с заданными маршрутами;

FTP — *File Transfer Protocol*, протокол передачи файлов по сети;

GRE — *Generic Routing Encapsulation*, протокол туннелирование с общей инкапсуляции маршрутов;

GTSM — *Generalized TTL Security Mechanism*, метод обеспечения безопасности передачи данных, основанный на времени жизни пакета;

HDLC — *High-Level Data Link Control*, высокоуровневый контроль канала передачи данных - протокол второго уровня модели OSI, обеспечивает передачу данных между устройствами в режиме точка-точка или точка-многоточка;

HEX — *Hexadecimal*, шестнадцатеричная система счисления;

HTTP — *HTTP (HyperText Transfer Protocol*, протокол передачи гипертекстовых данных;

IA — *Inter Area*, обозначение межобластного маршрута OSPF;

IANA — *Internet Assigned Numbers Authority*, Администрация адресного



пространства Интернет;

ICMP — *Internet Control Message Protocol*, протокол управляющих сообщений Интернета;

ID — *Identifier*, идентификатор;

IEEE — *Institute of Electrical and Electronics Engineers*, Институт инженеров электротехники и электроники США;

IETF — *Internet Engineering Task Force*, целевая группа по интернет-инжинирингу;

IGMP — *Internet Group Membership Protocol*, протокол управления группами Интернета;

IGP — *Interior Gateway Protocol*, внутренний (внутри автономной системы) протокол маршрутизации;

IKE — *Internet Key Exchange*, протокол установления и управления безопасными криптографическими соединениями в рамках IPsec;

ILM — *Incoming Label Map*, таблица входящих меток в протоколе MPLS;

IMISH — *Integrated Management Interface Shell*, интегрированная оболочка интерфейса управления;

IP — *Internet Protocol*, межсетевой протокол;

IPC — *inter-process communication*, обмена данными и синхронизацией в рамках распределённой системы;

IPFIX — *internet protocol flow information export*, стандартизированный протокол, для сбора и экспорта данных о сетевых потоках;

IPMI — *Intelligent Platform Management Interface*, стандартизированный интерфейс удалённого управления;

IPv4 — *Internet Protocol version 4*, четвёртая версия интернет-протокола (IP);

IS-IS — *Intermediate System to Intermediate System*, протокол маршрутизации промежуточных систем;

ISO — *International Organization for Standardization*, международная организация по стандартизации;

KVM — *Kernel-based Virtual Machine*, виртуализация в среде Linux на платформе x86;

L2VPN — *Layer 2 Virtual Private Network*, виртуальные частные сети канального уровня;

LACP — *Link Aggregation Control Protocol*, стандартный протокол для объединения нескольких физических соединений в одно логическое;

LAN — *Local Area Network*, локальная вычислительная сеть;

LDP — *Label Distribution Protocol*, протокол распределения меток;



LER — *Label Edge Router*, граничный маршрутизатор сети MPLS;

LFA — *Loop-Free Alternate*, технология быстрого восстановления маршрутов в случае сбоя;

LIB — *Label Information Base*, таблица всех меток в сетях MPLS известных маршрутизатору;

LLDP — *Link Layer Discovery Protocol*, протокол канального уровня для обнаружения и обмена данными с соседними устройствами;

LSA — *Link State Advertisement*, сообщение с описанием локального состояния маршрутизатора или сети;

LSP — *Label Switched Path*, предопределённый путь в сети MPLS;

LSR — *Label Switch Router*, маршрутизатор осуществляющий коммутацию по меткам внутри сети MPLS;

LSU — *Link-State Update*, сообщение об изменениях в топологии сети в протоколе OSPF;

MAC — *Media Access Control*, подуровень управления доступом к среде в сетевой модели OSI;

MD5 — *Message Digest Algorithm 5*, криптографическая хеш-функция;

MED — *Multi-Exit Discriminator*, атрибут протокола BGP для выбора пути к автономной системе;

MF — *More Fragments*, бит, используемый при обработке фрагментированных IP-сегментов;

MFC — *Multi-Field Classification*, метод классификации сетевого трафика по нескольким полям в заголовке;

MGMT—*Management*, порт внешнего управления устройством;

MIB — *Master Information Block*, блок системной информации;

MMF — *Multi-mode optical fiber*, многомодовое оптическое волокно;

MPLS — *Multiprotocol Label Switching*, механизм передачи данных по меткам;

MPO — *Multi-Fiber Push-On*, тип оптического разъёма, используемый в SFP модулях;

NAK — смотри NACK;

NAT — *Network Address Translation*, механизм преобразования адресов в сетях TCP/IP;

NBMA — *Non-broadcast Multiple Access*, нешироковещательный множественный доступ;

NHT — *Next Hop Tracking*, функция уведомления протокол BGP об изменении маршрутизации для следующего перехода BGP;



NLRI — *Network Layer Reachability Information*, информация сетевого уровня о доступности сети в протоколе BGP;

NMS — *Network Management System*, система управления сетью;

NSM — *Networking Services Manager*, компонент, отвечающий за управление сетевыми сервисами, такими как маршрутизация, QoS, безопасность;

NSSA — *Not-So-Stubby Area*, тупиковая зона OSPF из которой объявляются внешние маршруты;

NTP — *Network Time Protocol*, протокол сетевого времени;

OID — *Object Identifier*, уникальный идентификатор в протоколе SNMP;

ORF — *Outbound Route Filtering*, функция динамического управления фильтрацией исходящих маршрутов в протоколе BGP;

OSI — *Open Systems Interconnection model*, сетевая модель стека сетевых протоколов;

OSPF — *Open Shortest Path First*, протокол динамической маршрутизации по кратчайшему маршруту;

OVC — *Operator Virtual Connection*, соединение ассоциированных внешних интерфейсов одного оператора;

P2P — *peer-to-peer*, одноранговая децентрализованная или пиринговая сеть;

PADI — *PPPoE Active Discovery Initiation*, сообщение от клиента серверу в протоколе PPPoE с запросом о доступности сервиса;

PADO — *PPPoE Active Discovery Offer*, ответное сообщение сервера на сообщение PADI в протоколе PPPoE о готовности предоставить сервис;

PAP — *Password Authentication Protocol*, протокол простой проверки подлинности;

PAT — *Port Address Translation*, технология трансляции сетевого адреса в зависимости от TCP/UDP-порта получателя;

PBR — *Policy-Based Routing*, маршрутизация на основе политик;

PC — *Personal Computer*, персональный компьютер (ПК);

PDH — *plesiochronous digital hierarchy*, плезиохронная цифровая иерархия;

PDU — *protocol data unit*, базовая единица обмена данными между устройствами, в рамках определённого сетевого протокола;

PE — *Provider Edge*, граничный маршрутизатор со стороны оператора;

PID — *Process Identifier*, идентификатор процесса;

PIM — *Protocol Independent Multicast*, протоколнезависимая многоадресная передача данных;

PING — *Packet Internet Groper*, утилита для тестирования доступности узла в сети IP;



PIR — *Peak Information Rate*, максимальная средняя скорость передачи данных;

PM — *Post Meridien*, время после полудня;

PMTUD — *Path MTU Discovery*, стандартизированная технология для определения MTU на пути между двумя узлами IP-сети;

PPP — *Point-to-Point Protocol*, протокол вида точка-точка канального уровня;

PRC — *Partial Route Calculation*, механизм в протоколах OSPF или IS-IS для вычисления маршрутов только для части сети, а не для всей топологии;

PSH — *Push*, флаг в заголовке TCP инструктирует получателя протолкнуть данные, накопившиеся в приёмном буфере, в приложение пользователя;

PSU — *Power supply unit*, блок питания устройства;

PTP — *Precision Time Protocol*, протокол точного времени;

PW — *Pseudowire*, виртуальный канал для эмуляции L2 услуг через сети уровня L3;

QEMU — *Quick Emulator*, ПО для запуска ОС и программ на компьютерах разной архитектуры, а также для создания и управления виртуальными машинами.;

QSFP — *Quad Small Form-factor Pluggable*, стандарт модульных компактных сетевых трансиверов со скоростью передачи данных 40 Гбит/с и более;

QSFP+ — *QSFP Plus*, см. QSFP;

RD — *Route Distinguisher*, уникальный идентификатор, используемый в MPLS VPN для различения маршрутов с одинаковыми IP-адресами, но принадлежащих разным VPN;

RED — *Random Early Detection*, произвольное раннее обнаружение;

RFC — *Random early detection*, произвольное раннее обнаружение;

RIB — *Route Information Base*, таблица маршрутизации;

RID — *RouterID*, идентификатор маршрутизатора;

RIP — *Routing Information Protocol*, протокол маршрутной информации;

RP — *Rendezvous Point*, точка распределения многоадресных данных;

RPF — *Reverse path forwarding*, пересылка по обратному пути — метод для беспетельной многоадресной передачи;

RPM — *revolutions per minute*, единица частоты вращения — обороты в минуту;

RR — *Route Reflector*, сервер отражения маршрутов;

RSA — *Rivest, Shamir и Adleman*, криптографический алгоритм с открытым ключом;

RSPAN — *Remote Switch Port Analyzer*, это функция на сетевых коммутаторах, которая позволяет зеркалировать трафик с одного порта (или VLAN) на удалённый порт в другой части сети для анализа и мониторинга;

RST — *Reset*, флаг в заголовке TCP для немедленного завершения TCP-соединения;



RSVP — *Resource ReSerVation Protocol*, протокол резервирования сетевых ресурсов;

RT — *Route Target*, атрибут, используемый в VRF для управления импортом и экспортом маршрутов между различными VPN;

RX — *Receive*, приём данных;

SCP — *Secure Copy Protocol*, протокол безопасного копирования файлов между компьютерами;

SEL — *Selector*, селектор;

SFF — *Small Form Factor*, малый форм-фактор (как правило речь о модулях SFP);

SFP — *Small Form-factor Pluggable*, промышленный стандарт модульных компактных приёмопередатчиков (трансиверов);

SI — *Service Instance*, логический субинтерфейс, работающий между L2 и L3 уровнями;

SLA — *Service Level Agreement*, соглашение об уровне обслуживания;

SMF — *Single-mode optical Fiber*, одномодовое оптическое волокно;

SNAT — *Source Network Address Translation*, изменение адреса и порта источника пакета;

SNMP — *Simple Network Management Protocol*, стандартный интернет-протокол для управления устройствами в IP-сетях;

SPAN — *Switch Port Analyzer*, это функция на сетевых коммутаторах, которая позволяет зеркалировать трафик с одного порта (или VLAN) на другой порт для анализа и мониторинга;

SPF — *Shortest Path First*, алгоритм протоколов OSPF и IS-IS;

SPT — *Shortest Path Tree*, дерево кратчайшего пути для вычисления кратчайшего пути от одного узла сети до всех остальных узлов;

SRC — *Source*, источник данных;

SSH — *Secure Shell*, безопасная оболочка — протокол удалённого управления операционной системой;

SSM — *Source-Specific Multicast*, многоадресная передача данных с привязкой к источнику;

SVI — *Switched Virtual Interface*, логический интерфейс третьего уровня на коммутаторе;

SYN — *Synchronization*, синхронизация;

TAB — *Tabulation*, клавиша табуляции на клавиатуре компьютера;

TCP — *Transmission Control Protocol*, протокол управления передачей;

TFTP — *Trivial File Transfer Protocol*, простой протокол передачи файлов;

TIB — *Tag Information Base*, таблица в сетях MPLS для сопоставления меток с маршрутами;

TLS — *transport layer security*, протокол защиты транспортного уровня;

TOS — *Type of Service*, тип сервиса, поле - второй байт заголовка IPv4;

TPID — *Tag Protocol Identifier*, идентификатор протокола тегирования;

TTL — *Time To Live*, время жизни пакета данных;

TX — *Transmit*, передача (радиосигнала);

UDP — *User Datagram Protocol*, протокол пользовательских датаграмм;

UP — *User Plane*, абонентская плоскость - набор протоколов для передачи абонентских данных в сетях LTE;

URL — *Uniform Resource Locator*, единообразный указатель местонахождения ресурса;

USB — *Universal Serial Bus*, универсальная последовательная шина;

UTC — *Coordinated Universal Time*, всемирное координированное время;

VC — *Virtual circuit*, интерфейс от одного граничного маршрутизатора в сторону другого граничного маршрутизатора;

VCORE — *Voltage Core*, напряжение, подаваемое на ядро процессора (CPU) для его работы;

VLAN — *Virtual Local Area Network*, виртуальная локальная сеть;

VM — *Virtual Machine*, виртуальная машина — программная эмуляция физического компьютера, которая позволяет запускать операционные системы и приложения в изолированной среде на одном физическом устройстве;

VPLS — *Virtual Private LAN Services*, класс VPN с поддержкой подключения нескольких узлов в едином мостовом домене сети IP/MPLS;

VPN — *Virtual Private Network*, виртуальная частная сеть — обобщённое название обеспечивающих сетевые соединения поверх другой сети;

VPWS — *Virtual Private Wire Service*, тип услуги VPN, который эмулирует поведение выделенной линии точка-точка или физического провода в общей пакетной сети;

VR — *Virtual Router*, виртуальный роутер;

VRF — *Virtual routing and forwarding*, несколько виртуальных таблиц маршрутизации и пересылки на одном физическом маршрутизаторе;

VRRP — *Virtual Router Redundancy Protocol*, протокол резервирования виртуального маршрутизатора;



VSI — *Virtual Switch Instance*;

VTY — *Virtual Switch Instance*, логический коммутатор, который эмулирует работу Ethernet-коммутатора в виртуальной частной сети;

WAN — *Wide Area Network*, глобальная вычислительная сеть;

WRED — *Weighted Random Early Detection*, взвешенное произвольное раннее обнаружение;

WWW — *World Wide Web*, информационная система, для обмена данными через сеть Интернет;

БА — база данных;

ВМ — виртуальная машина;

МСЭ-Т — Международный союз электросвязи;

ОС — операционная система;

ПК — персональный компьютер;

ПО — программное обеспечение.

Подключение к EcoRouter

Пароль и логин по умолчанию для входа в EcoRouterOS: `admin`, `admin`.
IP-адрес MGMT-порта по умолчанию: 192.168.255.1/24.

Команда для входа по SSH: `ssh <LOGIN>@<IP>`, где **LOGIN** логин по умолчанию **"admin"** или заданный вами логин и **IP** — заданный адрес интерфейса на маршрутизаторе. После подключения система попросит вас ввести пароль.

Команда для входа через Telnet: `Telnet <IP>`, где **IP** также заданный адрес интерфейса. После подключения система также попросит вас ввести логин и пароль.

Подключение через консольный порт

Консольный порт (порт 8P8C, он же RJ45, обычно расположен в левой части передней панели маршрутизатора) имеет стандартный порядок контактов и совместим с консольными кабелями Cisco и других вендоров. Настройка порта: 115200 8N1 No flow control.

Подключите консольный кабель и воспользуйтесь любым клиентским приложением с функцией подключения к соответствующему последовательному (COM) порту.

При соединении через консольный порт не потребуется вводить логин и пароль.

Подключение через порт управления MGMT

Порт управления `mgmt` (обычно расположен первым слева в группе встроенных гигабитных ethernet-портов и имеет маркировку MNG или GEO) имеет IP-адрес по умолчанию 192.168.255.1/24.

Интерфейс порта управления (MGMT) принадлежит специальному **VRF management** (Virtual Routing and Forwarding instances) которому назначен **Security profile none** разрешающий любые подключения.

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>`.

Физически подключите кабель и в консоли управляющего устройства воспользуйтесь SSH или Telnet подключением.

1 Оборудование

Модели EcoRouter представлены в следующем порядке:

- ER-110, ER-116, ER-406, ER-T406,
- ER-1004 NP1, ER-1004 NP2, ER-1004 L, ER-T1004,
- ER-1004 S, ER-2008, ER-2008 S.

Таблица 1 — Младшая линейка устройств EcoRouter

Платформа	ER-110	ER-116	ER-406	ER-T406
Производительность	до 6 Gbps	до 26 Gbps	до 46 Gbps	до 44 Gbps
Форм-фактор	Desktop	Desktop	1U	1U
Сетевые интерфейсы:				
- 1 GE Copper	6	2	6	4
- 2,5 GE Copper	—	4	—	—
- 1 GE Fiber (SFP)	—	2	—	—
- 10 GE Fiber (SFP+)	—	2	4	4
Консольный порт	Mini-USB	USB Micro B / RJ45	RJ45	RJ45
Блок питания	12 VDC	12 VDC	Dual 300W AC/DC	Dual 300W AC/DC
Охлаждение	Passive	Passive	2 Fans	2 Fans
Размеры (ДхШхВ), мм	90×90×44	256×200×44	292×430×44	250×450×44

Таблица 2 — Старшая линейка устройств EcoRouter

Платформа	ER-1004 NP1	ER-1004 NP2	ER-1004 L	ER-T1004
Производительность	До 56 Gbps	До 88 Gbps	до 200 Gbps	до 200 Gbps
Форм-фактор	1U	1U	1U	1U
Модульные слоты	2	2	4	4
Сетевые интерфейсы				
- 1 GE Copper	8	8	1	1
- 1 GE Fiber (SFP)	8	—	—	—
- 10 GE Fiber (SFP+)	4	8	2	1

Платформа	ER-1004 NP1	ER-1004 NP2	ER-1004 L	ER-T1004
Интерфейс управления	RJ45 + IPMI	RJ45 + IPMI	RJ45 + IPMI	RJ45 + IPMI
Консольный порт	RJ45	RJ45	RJ45	RJ45
Блок питания	Dual 300W AC/DC	Dual 300W AC/DC	Dual 650W AC/DC	Dual 550W AC/DC
Охлаждение	3 Hot Swap, Smart Control	3 Hot Swap, Smart Control	4 Hot Swap, Smart Control	4 Hot Swap, Smart Control
Размеры (ДхШхВ), мм	500×430×44	500×430×44	650×430×44	500×430×44

Платформа	ER-1004 S	ER-2008	ER-2008 S
Производительность	до 400 Gbps	до 400 Gbps	до 800 Gbps
Форм-фактор	1U	2U	2U
Модульные слоты	4	8	8
Сетевые интерфейсы			
- 1 GE Copper	—	2	2
- 1 GE Fiber (SFP)	—	—	—
- 10 GE Fiber (SFP+)	—	—	—
Интерфейс управления	RJ45	RJ45 (IPMI)	RJ45 (IPMI)
Консольный порт	RJ45	RJ45	RJ45
Блок питания	Dual 550W AC/DC	Dual 1300(1600)W AC/DC	Dual 1600(2000)W AC/DC
Охлаждение	4 Hot Swap, Smart Control	4 Hot Swap, Smart Control	4 Hot Swap, Smart Control
Размеры (ДхШхВ), мм	610×438×44	720×430×88	760×430×88

Таблица 3 — Перечень сетевых интерфейсных карт для старшей линейки устройств EcoRouter

Модель карты	Интерфейсы
NIC-8GE-TX	8xRJ45
NIC-4GE-SFP	4xSFP
NIC-4GE-TX-4GE-SFP	4xRJ45 + 4xSFP

Модель карты	Интерфейсы
NIC-8GE-SFP	8xGbE SFP
NIC-4XGE-SFPP	4x10GbE SFP+
NIC-8XGE-SFPP	8x10GbE SFP+
NIC-2x25GE-SFP28	2x25GbE SFP28
NIC-4x25GE-SFP28	4x25GbE SFP28
NIC-2x40GE-QSFP	2x40GbE QSFP+
NIC-2x100GE-QSFP28	2x100GbE QSFP28

У всех устройств серии на передней панели расположены:

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

На передней панели "младших" моделей серии также расположен разъём кабеля питания. В случае если питание производится от сети переменного тока, там же расположена кнопка включения питания.

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

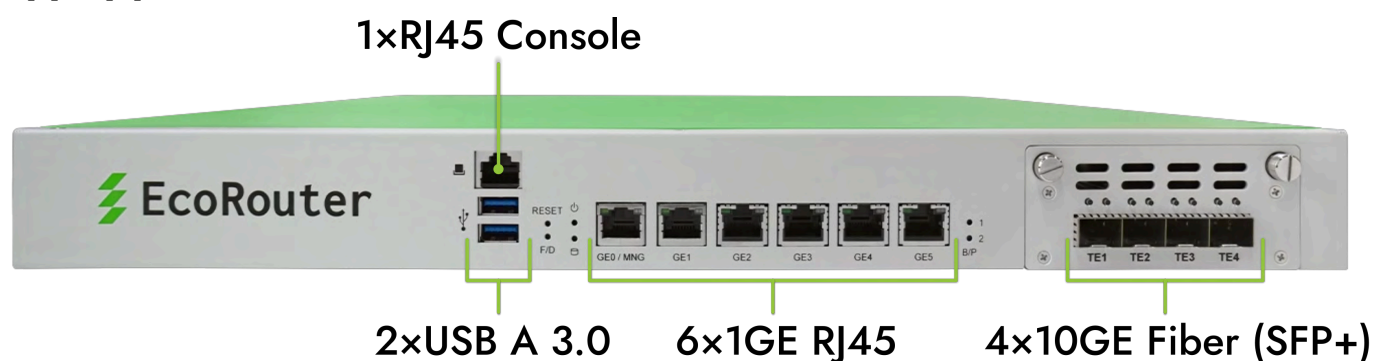


Рисунок 1

У "старших" моделей серии разъёмы 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.

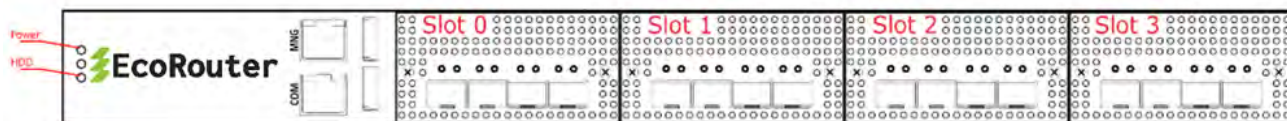


Рисунок 2

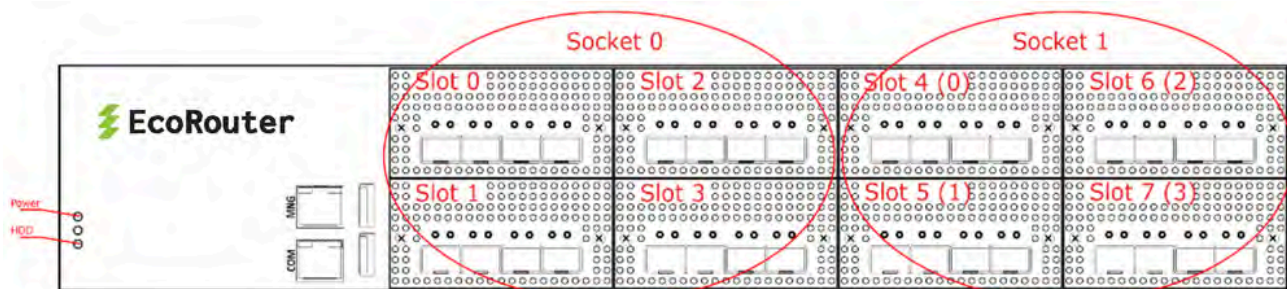


Рисунок 3

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 Сетевые порты

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.

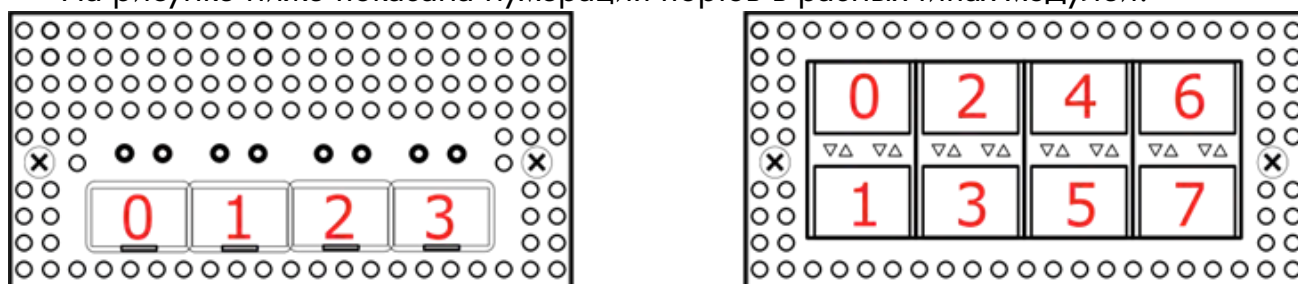


Рисунок 4

1.2 Сетевые карты

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
slot0	NIC-8GE-TX		
slot1	NIC-4XGE-SFP		
	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
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 SFP модули

Модули EcoRouter могут быть снабжены разным набором сетевых интерфейсов (10/100/1000 МбE, 1, 10, 25, 40, 100 GbE). Поддерживается горячая замена оптических модулей, модули могут быть подключены или отключены после старта системы.

Изготовитель гарантирует корректную работу устройств EcoRouter с рекомендованными для использования SFP-модулями.

Изготовитель не ограничивает возможность использования модулей сторонних производителей, совместимых с сетевыми адаптерами Intel.

Таблица 4 — Рекомендованные интерфейсные модули SFP/SFP+/SFP28/QSFP+/QSFP28

EcoSFP-1GE-T	Трансивер EcoSFP 1GbE SFP T (Copper Transceiver Module)
EcoSFP-1GE-SX	Трансивер EcoSFP 1GbE SFP SX (850nm for up to 300m Transmission on Multi-Mode Fiber)
EcoSFP-1GE-LX	Трансивер EcoSFP 1GbE SFP LX (1310nm for up to 10km Transmission on Single Mode Fiber)
EcoSFP-1GE-ZX	Трансивер EcoSFP 1GbE SFP ZX (1550nm for up to 80km Transmission on Single Mode Fiber)
EcoSFP-10GE-SR	Трансивер EcoSFP 10-GbE SFP+ SR (850nm for up to 300m Transmission on Multi-Mode Fiber)
EcoSFP-10GE-LR	Трансивер EcoSFP 10-GbE SFP+ LR (1310nm for up to 10km Transmission on Single Mode Fiber)
EcoSFP-10GE-ER	Трансивер EcoSFP 10-GbE SFP+ ER (1550nm for up to 40km Transmission on Single Mode Fiber)
EcoSFP-10GE-ZR	Трансивер EcoSFP 10-GbE SFP+ ZR (1550nm for up to 80km Transmission on Single Mode Fiber)
EcoQSFP-40G-SR4	Трансивер EcoQSFP+ 40GBASE-SR4 MPO/UPC (850nm for up to 300m Transmission on Multimode Fiber)
EcoQSFP-40G-IR4	Трансивер EcoQSFP+ 40GBASE-IR4 (1310nm for up to 2km Transmission on Single Mode Fiber)
EcoQSFP-40G-LX4	Трансивер EcoQSFP+ 40GBASE-LX4 (1310nm 150m over MMF up to 2km Transmission on Single Mode Fiber)

EcoSFP-1GE-T	Трансивер EcoSFP 1GbE SFP T (Copper Transceiver Module)
EcoQSFP-40G-LR4	Трансивер EcoQSFP+ 40GBase-LR4 (1310nm for up to 10km Transmission on Single Mode Fiber)
EcoQSFP-PSM-LR4	Трансивер EcoQSFP+ 40GBase-LR4 MPO/APC (1310nm for up to 10km Transmission on Single Mode Fiber)
EcoQSFP-100G-LR4	Трансивер EcoQSFP28 100GBase-LR4 (1310nm for up to 10km Transmission on Single Mode Fiber)
EcoCBL-10G-DAC-1	Кабель SFP+ 10 Gigabit Ethernet Direct Attach Copper (Twinax Copper Cable), 1M
EcoCBL-10G-DAC-3	Кабель SFP+ 10 Gigabit Ethernet Direct Attach Copper (Twinax Copper Cable), 3M
EcoCBL-10G-DAC-5	Кабель SFP+ 10 Gigabit Ethernet Direct Attach Copper (Twinax Copper Cable), 5M
EcoCBL-10G-DAC-7	Кабель SFP+ 10 Gigabit Ethernet Direct Attach Copper (Twinax Copper Cable), 7M
EcoCBL-40G-PSM-2	Кабель 40G QSFP+ to 4xSFP+10G Direct Attach Copper (Twinax Copper Cable), 2м
EcoCBL-QSFP-2	Кабель 40G QSFP+ to QSFP+ Direct Attach Copper (Twinax Copper Cable), 2м
EcoCBL-QSFP28-2	Кабель 100G QSFP28 to QSFP28 Direct Attach Copper (Twinax Copper Cable), 2м
EcoCBL-MPO/APC-4xDLC-2M	Кабель оптический MPO/APC to 4xDLC, SM, 2м

Маршрутизатор поддерживает работу некоторых SFP-модулей с меньшей производительностью (1 GbE в порте 10 GbE). При вставке модуля в порт он может быть сразу включен в работу без перезагрузки устройства. Однако, если не удастся поднять порт в состояние UP, то может потребоваться повторная инициализация порта при помощи команды **port-reload** в режиме конфигурации L2 порта. Если и это не помогло, значит данный SFP-модуль не поддерживается.

Примечание: Если порт находится в группе LAG, то для повторной инициализации порта необходимо сначала вывести порт из LAG (команда **no bind <имя порта>** в режиме конфигурирования LAG-порта, см. раздел "Агрегирование каналов"), а затем уже ввести команду **port-reload**.

Если в порт вставить модуль большей производительности (например, 10 GbE в порт 1 GbE), то работать он не будет, хотя может определяться системой.

1.4 Лицензионные программные расширения

Помимо стандартного ПО EcoRouterOS, возможности маршрутизаторов EcoRouter могут быть расширены с помощью специальных лицензий.

Так, для маршрутизаторов EcoRouter, предназначенных для обеспечения технологии широкополосного доступа BNG (BRAS) могут быть подключены следующие лицензии, рассчитанные на разные степени нагрузок:

- BRAS для 1 000 активных абонентских сессий;
- BRAS для 2 000 активных абонентских сессий;
- BRAS для 4 000 активных абонентских сессий;
- BRAS для 8 000 активных абонентских сессий;
- BRAS для 16 000 активных абонентских сессий;
- BRAS для 32 000 активных абонентских сессий;
- BRAS для 64 000 активных абонентских сессий;
- BRAS для 128 000 активных абонентских сессий.

Для каждой модели роутеров EcoRouter может быть подключена лицензия, обеспечивающая функционал сервера широкополосного доступа (BRAS), L2TP сервера (LNS), виртуальных роутеров (VR), виртуальных машин и контейнеров (VM).

1.5 Блоки питания, вентиляторы и сенсоры

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 5 — Вывод параметров электропитания

Параметр	Описание
CPU VCORE	Напряжение на процессоре. Предупреждение (ALARM) не выдается, потому что значение может сильно



Параметр	Описание
	варьироваться от процессора, значение может завышать сама плата. Выводится для информации
+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
CPU _n	Температура процессора. Предупреждение (ALARM) выдается, если температура превышает 90°C
MB	Температура материнской платы. Предупреждение (ALARM) выдается, если температура превышает 70°C
FAN _n	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 определённого 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 справочной информации.

2.1 Подключение к EcoRouter

Подключиться к маршрутизатору можно следующими способами:

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

Пароль и логин по умолчанию для входа в EcoRouterOS: `admin`, `admin`.

Команда для входа по SSH: `ssh <LOGIN>@<IP>`, где **LOGIN** логин по умолчанию "**admin**" или заданный вами логин и **IP** — заданный адрес интерфейса на маршрутизаторе. После подключения система попросит вас ввести пароль.

Команда для входа через Telnet: `Telnet <IP>`, где **IP** также заданный адрес интерфейса. После подключения система также попросит вас ввести логин и пароль.

Обратите внимание, что подключение через линейные Ethernet-порты ограничено профилями безопасности (см. подраздел "Профили безопасности" раздела "Локальная авторизация").

Для успешного подключения алгоритмы шифрования на стороне клиента должны соответствовать алгоритмам поддерживаемым в EcoRouterOS:

- Команда `show ssh cipher info` выводит список поддерживаемых алгоритмов шифрования передаваемых данных.
- Команда `show ssh hash info` выводит список поддерживаемых алгоритмов хеширования.
- Команда `show ssh key-exchange info` выводит список поддерживаемых алгоритмов обмена ключами шифрования. Методы обмена ключами шифрования соответствуют перечню представленному в разделе четвёртом рабочего предложения IETF RFC 9142.

2.1.1 Консольный порт

Консольный порт (порт 8P8C, он же RJ45, обычно расположен в левой части передней панели маршрутизатора) имеет стандартный порядок контактов и совместим с консольными кабелями Cisco и других вендоров. Настройка порта: 115200 8N1 No flow control.

Подключите консольный кабель и воспользуйтесь любым клиентским приложением с функцией подключения к соответствующему последовательному (COM) порту.

При соединении через консольный порт не потребуется вводить логин и пароль.

2.1.2 Порт MGMT

Порт управления mgmt (обычно расположен первым слева в группе встроенных гигабитных ethernet-портов и имеет маркировку MNG или GEO) имеет IP-адрес по умолчанию 192.168.255.1/24.

Интерфейс порта управления (MGMT) принадлежит специальному **VRF management** (Virtual Routing and Forwarding instances) которому назначен **Security profile none** разрешающий любые подключения.

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>`.

Физически подключите кабель и в консоли управляющего устройства воспользуйтесь SSH или Telnet подключением.

2.1.3 Линейные порты

Подключение через линейные Ethernet-порты ограничено профилями безопасности (см. подраздел "Профили безопасности" раздела "Локальная авторизация"). Все интерфейсы линейных портов по умолчанию принадлежат **VRF default** (Virtual routing and forwarding default) которому назначен профиль **Security profile default** запрещающий подключение по протоколам SSH, Telnet и SNMP.

Для быстрого снятия ограничений достаточно в режиме конфигурации ввести команду `no security default`. Профиль безопасности **default** будет снят с **VRF default** и вместо него назначен профиль **none**. Следовательно, будут сняты ограничения со всех интерфейсов, по умолчанию принадлежащих **VRF default**.

Для интерфейсов не из **VRF default** для подключения возможности SSH-доступа необходимо войти в контекстный режим соответствующего VRF `ip vrf <VRF_NAME>` и дать команду **transport input ssh**.

Обратите внимание, что снятие ограничений в виде профиля безопасности **Security profile default** со всех интерфейсов принадлежащих **VRF default** делает их открытыми для несанкционированного доступа. Такое решение можно рассматривать лишь как временное, для первичного подключения, в будущем будет необходимо перенастроить профили безопасности и привязку интерфейсов к VRF в соответствии с политикой безопасности вашей сети.

Для подключения необходимо настроить порт и интерфейс с IP-адресом. Подробно работа с портами и интерфейсами описана в разделе 4 "Виды портов и интерфейсов". Пример настройки порта и интерфейса с адресом 192.168.255.88:

```
configure
interface te1
ip address 192.168.255.88/24
exit
port te1
service-instance te1
encapsulation untagged
connect ip interface te1
```

Результаты настройки можно проверить командой `do show ip interface brief`.

На управляющей машине необходимо настроить адрес из той же подсети, что был выбран для интерфейса роутера.

Физически подключите кабель и в консоли управляющего устройства воспользуйтесь SSH или Telnet подключением.

Обратите внимание что доступ с помощью Telnet будет работать только для портов в **VRF default**. В пользовательских VRF, доступ с помощью Telnet запрещён в целях информационной безопасности.

2.2 Виртуальные интерфейсы удалённого доступа

VTY — VirtualTeletype, виртуальный интерфейс, который обеспечивает удаленный доступ к устройству.

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

EcoRouterOS поддерживает до 872 одновременных сессий по протоколам Telnet и SSH через менеджмент-порт 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-сессий.

Таким образом, команда `line vty 0 871` указывает маршрутизатору, что следующие за ней настройки будут применены ко всем 872 виртуальным линиям, а команда `line vty 7` служит для настройки 7 линии.

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

Таблица 6 — Команды режима конфигурирования консоли

Команда	Описание
<code>exec-timeout <0-35791> <0-2147483></code>	<p>Время ожидания. Если за указанный интервал времени в данной сессии на данной виртуальной линии (консоли) не будет произведено никаких действий, система автоматически завершит сессию с сообщением типа "User is logged out by timeout" или "Vty connection is timed out". Для возобновления сессии пользователю необходимо будет снова ввести свой логин и пароль.</p> <p>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</p>
<code>history max <0-2147483647></code>	<p>Количество команд, которое будет сохраняться в буфере команд. Буфер доступен по нажатию клавиши стрелка вверх «↑». По умолчанию значение равно максимально возможному</p>

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

Пример вывода информации о подключённых пользователях:

```
ecorouter#show users connected
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.

2.3 Подключение к другим сетевым устройствам

Находясь в интерфейсе командной строки EcoRouter вы можете совершать подключения к другим устройствам в сети по протоколам SSH и Telnet.

Для такого подключения может потребоваться расширенный набор параметров и команда примет следующий вид:

`ssh <USER>@<DEST_IP> ip <SOURCE_IP> vrf <VRF_NAME>`, где **USER** — имя пользователя на устройстве адресата, **DEST_IP** — IP-адрес назначения, **SOURCE_IP** — IP-адрес источника, **VRF** — VRF к которому принадлежит адрес источника.

Ниже два примера команд: одна для соединения через управляющий порт и вторая для соединения через интерфейс привязанный к VRF с названием "MNG".

```
ssh admin@192.168.123.88 vrf management
ssh admin@192.168.123.99 ip 192.168.123.88 vrf MNG
```

В случае повторного подключения по SSH к тому же устройству (по тому же IP), существует возможность, что идентификатор удалённого устройства мог измениться и будет получено предупреждение "REMOTE HOST IDENTIFICATION HAS CHANGED!". Для решения данной проблемы нужно выполнить команду `ssh remove-keys <remote_ip>`, которая очистит сохранённые идентификаторы связанные с указанным ip. После этого можно будет заново подключиться к устройству.

Для соединения через Telnet, также нужно указать VRF источника:

```
telnet 192.168.123.99 vrf MNG .
```

Обратите внимание что подключение с помощью Telnet будет работать только при обращении к интерфейсам в **VRF default** и **VRF management**. В созданных пользователем VRF, доступ с помощью Telnet запрещён в целях информационной безопасности.

После подключения внешнему устройству работающему под управлением OS Linux может возникнуть ситуация, когда при нажатии клавиш на экране командной строки не будет происходить никаких видимых изменений. На самом деле ввод команд продолжает работать, но вывод в консоль не происходит или происходит с огромным запозданием.

Для решения этой проблемы, нажмите **Enter**, чтобы начать ввод с новой строки, введите команду Linux `reset` и снова нажмите **Enter**. После этого вводимые данные должны отображаться в консоли в реальном времени.

2.4 Интерфейс командной строки

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

EcoRouter даёт доступ к нескольким уровням командной строки. Каждый уровень характеризуется разным набором возможных команд.

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

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

Таблица 7 — Режимы работы консоли

Режим и приглашение командной строки	Описание	Как попасть в режим
Пользовательский режим (user-exec) <code>ecorouter></code>	Этот режим позволяет просматривать текущее состояние устройства, соединений, использовать сетевые утилиты	Подключиться к устройству
Режим администрирования (enable-exec) <code>ecorouter#</code>	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
Режим конфигурирования (config) <code>ecorouter(config)#</code>	В режиме конфигурирования можно изменять и задавать настройки, которые повлияют на работу устройства в целом	Ввести команду configure terminal , находясь в режиме администрирования

Режим и приглашение командной строки	Описание	Как попасть в режим
Контекстный режим (context-config) <code>ecorouter(config-КОНТЕКСТ)#</code>	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 определённых команд в режиме конфигурирования

При входе на устройство пользователь оказывается в режиме просмотра и видит приглашение командной строки в следующем виде: `ecorouter>`.

Чтобы переключиться в режим администрирования, нужно ввести команду `enable`, после чего приглашение командной строки изменит вид на `ecorouter#`. Чтобы вернуться в режим просмотра, нужно ввести команду `disable`.

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

```
EcoRouterOS version 3.0.0 EcoRouter 04/01/16 17:28:12
ecorouter>enable
ecorouter#configure 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 сессии 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**.

2.5 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#
```

2.6 Работа с конфигурацией

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 стартовом конфигурационном файле до тех пор, пока не будет введена команда **write**. Если команда **write** не была дана, после перезагрузки устройства текущие изменения будут сброшены до состояния на момент предыдущей загрузки и не будут применены.



У команды **write** есть несколько аргументов:

- **write memory** — сохранение текущей конфигурации в стартовый конфигурационный файл, равнозначный эффект даёт ввод команды **write** без аргументов;
- **write terminal** — вывод текущей конфигурации на экран, аналог команды **show running-config**;
- **write file <FILENAME>** — сохранение текущей конфигурации в локальный файл с произвольным именем.

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

Также, при работе с файлами конфигурации, будут полезны следующие команды:

- **show config file <FILENAME>** — выводит содержимое конфигурационного файла с указанным именем в консоль;
- **show files config** — выводит все сохранённые конфигурационные файлы;
- **copy file <FILENAME> startup config** — заменяет стартовой конфигурационный файл файлом с указанным именем. **Внимание!** Чтобы конфигурация заменённого файла вступила в силу, необходимо перезагрузить роутер.

2.7 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 'undebug')
develop      Debug command
```




```
disable    Turn off privileged mode command
enable     Turn on privileged mode command
```

Как было сказано выше, в разных режимах командной строки перечень команд будет различаться.

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
```

Команды также можно давать в сокращённом виде — по начальным буквам. Количество начальных букв команды должно быть достаточным, чтобы можно было отличить одну команду от других с теми же начальными буквами. Например, короткой записью для команды **show** будет **sh**. При такой записи также можно дополнить команду с начальных букв до полного вида 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 8 — Использование подсказок CLI

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> [TAB]	Пытается выполнить автозаполнение
стрелка вверх [↑]	Возврат к ранее введённой команде (история)
стрелка вниз [↓]	Возврат к команде, введённой позднее (история)

2.8 Команды группы show

Для просмотра параметров конфигурации 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 используйте клавишу **Q**.

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

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

Поддерживаемые модификаторы описаны в таблице ниже.

Таблица 9 — Модификаторы вывода

Команда	Описание
begin	Выводит строки, начинающиеся с заданного символа или группы символов
exclude	Выводит строки, исключаящие заданный символ или группу символов
grep	Выводит строки, соответствующие введённому регулярному выражению
include	Выводит строки, включающие заданный символ или группу символов
monitor	Выводит все строки и продолжает отслеживать изменения в течение указанного времени
nopager	Выводит на экран все строки сразу, без разбивки на страницы

Рассмотрим пример работы модификаторов.

Вывод команды со статусами всех существующих интерфейсов:

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

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

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

2.9 Утилита ping

Утилита **ping** является общим способом поиска неисправностей в сетях. Команда использует протокол ICMP для отправки серии эхо-пакетов для определения, является ли удалённое 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 mgmt 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**.

В случае запуска **ping** с маршрутизатора подключённого к сети через пользовательский VRF, будет необходимо указать адрес источника и к каком VRF он принадлежит. В таком случае команда примет следующий вид

```
ping 10.10.10.11 source 10.10.10.22 vrf VRFNAME
```

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 10 — Расширенные параметры утилиты ping

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

Поле	Описание
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.

2.10 Утилита **traceroute**

Утилита **traceroute** используется для обнаружения путей следования пакета до адресов удалённых 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 удалённому 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 удалённый 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, с которого была запущена 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 11 — Аргументы расширенной версии утилиты ***traceroute 12**

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

Поле	Описание
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 символическое, так и цифровое отображение; тем не менее можно отменить символическое 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#
```

2.11 Приветствие и баннер

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 приведён 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 Локальная аутентификация, авторизация и аккаунтинг

AAA (от англ. Authentication, Authorization, Accounting) — это концепция управления доступом в компьютерных системах и сетях, которая включает три основных процесса:

- Аутентификация (Authentication) — процесс проверки подлинности пользователя или устройства. Например, подтверждение личности с помощью логина и пароля, биометрических данных или цифровых сертификатов.
- Авторизация (Authorization) — определение прав и разрешений для доступа к ресурсам системы после успешной аутентификации.
- Учёт (Accounting) — фиксация и отслеживание действий пользователя или устройства в сети. Например: логирование времени доступа, объема использованных ресурсов или других данных для анализа и отчётности.

3.1 Вход в систему

При соединении с консолью управления EcoRouter система просит пользователя ввести логин и пароль, соответствующие одной из учётных записей пользователей в системе.

Изначально в EcoRouterOS задана учётная запись с ролью администратора. Для входа воспользуйтесь именем **admin** и паролем **admin**.

После аутентификации на консоль выводится версия системы и приглашение командной строки в виде заданного имени (hostname) устройства (по умолчанию имя устройства — "ecorouter") и значок пользовательского режима консоли (">").

Пример:

```
<<< EcoRouter 3.2.0.0.xxxxxxxxxxxxxxxxxx (x86_64) - ttyS0 >>>
ecorouter login: admin
Password:|
User Access Verification
EcoRouterOS version 3.2.0 EcoRouter 06/29/16 15:35:53
ecorouter>
```

Подключение к маршрутизатору может быть ограничено, профилями безопасности. Настройка профилей безопасности описана ниже в конце данного раздела.

3.2 Пользователи и их роли

Для разграничения уровней доступа в EcoRouter используются роли пользователей.

Следующие варианты ролей являются предопределёнными:

Таблица 13 — Разграничения уровней доступа пользователя

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

Для каждой роли доступен свой набор из общего состава команд.

Список команд для каждой роли указан в разделе Справочник команд.

Чтобы вывести на консоль подробные данные по всем ролям существует команда `show role`. Команда доступна только для роли администратора. По каждой роли будут выведены все доступные команды для каждого режима.

Три предопределённые роли нельзя изменить, но можно создать новую роль с нужным набором доступных команд и режимов.

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 предопределённых, тогда все команды и режимы, доступные для предопределенной роли, будут автоматически скопированы в новую роль. Например команда `role Administrator based-on admin` создаст новую роль **Administrator** с полными правами администратора.

Первый вариант создания роли более удобен, если нужно создать роль с небольшим набором команд. Второй вариант (на основе предопределенных) более удобен, если необходимо создать роль с большим набором команд или набором команд, незначительно отличающимся от одной из предопределённых ролей.

Для настройки созданной роли используется аналогичная команда конфигурационного режима `role <NAME>`.

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

командам. Обратите внимание, что запись в **DESCRIPTION** без кавычек не допускает использование пробелов, а в кавычках — допускает.

Чтобы добавить (сделать доступной) команду в определённом режиме, нужно использовать команду `permit {user-exec | enable-exec | config | context-config} <COMMAND>`. Чтобы удалить команду из роли потребуется та же команда с приставкой `no`. Режимы:

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

Второй обязательный параметр **COMMAND** — имя команды. Если название команды состоит из нескольких слов, например, **banner motd**, допускается указывать только первое слово (**banner**). Если команда добавлена, она будет работать с префиксами **no** и **do** (обратная команда и использования данной команды в конфигурационном режиме). Специально вносить команды с префиксами не нужно!

Обратите внимание, что ввод имени команды, в данном случае, никак не контролируется системой, так что вам следует самому позаботиться о правильном написании команды и о том в какой из режимов вы её добавляете. Напомним, что перечень команд доступных для каждой роли можно вывести командой `show role`.

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

Пример:

```
ecorouter(config)# role myrole
ecorouter(config-role)# permit user-exec enable
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`). Какие именно указано в разделе Справочник команд.

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

Внимание! Все изменения и добавления ролей и пользователей вступают в силу только после сохранения конфигурации командой `write`.

3.3 Настройка учётных записей пользователей

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

Далее в контекстном режиме задаются параметры учётной записи пользователя. Команды, управляющие этими параметрами, описаны ниже.

- `description <DESCR>` — добавить описание пользователя.
- `no description` — удалить описание пользователя.
- `password <PASS>` — задать пароль пользователя.
- `no password` — удалить пароль пользователя.
- `role {admin | noc | helpdesk | <NAME>}` — назначить пользователю роль. Указывается одно из значений: **admin**, **noc**, **helpdesk**, либо имя роли созданной пользователем.
- `no role {admin | noc | helpdesk | <NAME>}` — лишить пользователя роли.
- `vr <NAME>` — Разрешить пользователю доступ к виртуальному маршрутизатору.
- `no vr <NAME>` — Запретить пользователю доступ к виртуальному маршрутизатору.
- `activate` — активировать учётную запись пользователя.
- `deactivate` — отключить учётную запись пользователя.
- `lifetime <120-86400>` — "время жизни" учётной записи пользователя в секундах по истечении которого учётной записи будет присвоен статус `deactivate`.
- `lifetime day <1-131072>` — "время жизни" учётной записи пользователя в днях по истечении которого учётной записи будет присвоен статус `deactivate`.
- `ssh-key <SSH_KEY>` — позволяет установить SSH-ключ длиной до 500 символов и производить подключение к системе по ключу без ввода пароля.

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.

Следует пояснить поведение системы при совместном использовании команд `lifetime` и `activate` / `deactivate`.



- При установке `lifetime` для активной учётной записи начинается обратный отсчёт времени её действия, в течение которого возможности учётной записи доступны пользователю.
- Отсчитывается календарное время, не зависящее от присутствия пользователя в системе и других факторов.
- По истечению времени, учётная запись будет переведена в неактивное состояние равнозначное вводу команды `deactivate`. Активная сессия пользователя будет прерван, пользователь более не сможет войти в систему по логину и паролю своей учётной записи.
Подобное поведение системы должно побудить пользователя обратиться за новым паролем и продлением действия учётной записи к администратору сети, что положительно влияет на информационную безопасность.
- По команде `activate` учётная запись снова приходит в активное состояние, отсчёт `lifetime` начинается с начала, пользователь получает доступ к системе.
- Для изменения сроков жизни учётной записи достаточно повторно ввести команду `lifetime` или `lifetime day` с новыми сроками.

Для удаления учётной 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 `role <NAME>`.

Для удаления пользовательской роли используется команда `no 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 и приоритетом заданным командой `aaa precedence` (см. ниже).

3.4 Команды группы show

Для просмотра подключений, а также ролей 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_test

Для просмотра учётных записей пользователей, имеющих 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, как и для других команд **show**.

3.5 Локальный аккаунтинг

По команде `show log all | grep AUDIT` можно отфильтровать из общего журнала событий и вывести на консоль все действия пользователей, включая вход и выход из системы, и все введенные команды.



3.6 Служебные пользователи

По умолчанию в системе также существует служебный пользователь `daemon`, его невозможно удалить из системы. Он требуется для корректной локальной аутентификации, а также для работы служб AAA при взаимодействии с удалёнными RADIUS / TACACS+.

3.7 Приоритет способов авторизации и аутентификации

Для установки приоритета способов аутентификации и авторизации используется команда `aaa precedence <local | radius | tacacs>`.

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

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

В примере выше авторизация RADIUS-сервера будет доступна только в случае если ни один сервер TACACS+ не доступен. Локальная авторизация будет доступна только если не удалось установить связь ни с одним внешним AAA-сервером.

Ниже мы подробно разберём протоколы TACACS+ и RADIUS.

3.8 Уровни логирования служб AAA

В EcoRouterOS ведётся постоянное логирование (ведение журналов) всех действий системы и пользователей, в частности действий по аутентификации, авторизации и аккаунтингу. Подробнее о логировании вы можете прочесть в разделе "Логирование и отладка".

Здесь же следует упомянуть о настройке уровней логирования служб AAA, то есть о типах событий, а следовательно и их количестве, которые будут зарегистрированы системой и записаны в журнал.

Настройка уровня логирования производится командой `aaa logging level <NUM>`, где **NUM** уровень логирования от меньшего числа регистрируемых событий к большему. Каждый уровень включает в себя сообщения о событиях всех нижестоящих уровней.



Всего существует шесть уровней логирования. Установленное по умолчанию значение равно трём (Warning).

```
ecorouter(config)#aaa logging level ?
 1 critical
 2 error
 3 warning (default)
 4 notice
 5 info
 6 debug
```

Для более детального анализа и поиска неполадок при работе служб AAA в EcoRouter можно воспользоваться повышенным уровнем логирования.

ВНИМАНИЕ! При использовании уровня 6 (Debug) и наличии большого количества событий может значительно возрасти нагрузка на центральный процессор маршрутизатора.

Используйте максимальный уровень логирования с осторожностью, включая его лишь на время поиска неисправностей.

Для возврата к настройкам по умолчанию, воспользоваться командой ``aaa logging level 3` или `**` по `aaa logging level`**`.

3.9 Внешние процедуры AAA по протоколу TACACS+

TACACS+ (англ. Terminal Access Controller Access Control System plus) — сессионный 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 сессий. В рамках TACACS+ возможно установить три различных типа сессий AAA (англ. authentication, authorization, accounting). Установление одного типа сессии в общем случае не требует предварительного успешного установления какого-либо другого. Спецификация протокола не требует для открытия сессии авторизации открыть сначала сессию аутентификации. Сервер TACACS+ может потребовать аутентификацию, но протокол этого не оговаривает.

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

Синтаксис команды: `aaa tacacs-server <IP> [port <NUM>] [secret <PASS>] [vrf <NAME>] {account | auth} [cmd-authorize] [timeout <0-300>]`, где:

- `server <IP>` — IP-адрес TACACS-сервера.



- `port <NUM>` — номер порта на сервере.
- `secret <PASS>` — пароль для соединения с TACACS-сервером (должен совпадать с установленным на TACACS).
- `vrf <NAME>` — наименование VRF, в котором задан IP-адрес сервера (значение по умолчанию — VRF текущего виртуального маршрутизатора)
- `source <IP2>` — исходящий адрес.
- `account | auth` — выбор режима взаимодействия с сервером: **account** — только отправка учёта действий пользователя, **auth** — только аутентификация и авторизация. Для обеспечения всех функций AAA будет необходимо задать две команды `aaa radius-server`.
- `cmd-authorize` — включить покомандную авторизацию.
- `timeout <0-300>` — время в секундах в диапазоне от 0 до 300. Время ожидания до повторного запроса в случае отсутствия ответа от TACACS.

Пример:

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

Если используется несколько серверов TACACS+, то по умолчанию запросы будут отправляться первому доступному серверу из списка. На все серверы отправляется только информация о моменте входа и выхода пользователя, учётные данные передаются только на первый сервер из списка.

Если ни к одному серверу TACACS+ или RADIUS доступа нет, включается режим локальной аутентификации и авторизации.

Команда `aaa tacacs-config acct-all` включает отправку всех введённых команд пользователей (accounting) на все доступные TACACS-серверы.

3.9.1 Конфигурирование TACACS+

Для конфигурирования сервера TACACS+ использована библиотека **"tac_plus"**. С полной документацией по **"tac_plus"** можно ознакомиться по ссылке https://projects.pro-bono-publico.de/event-driven-servers/doc/index.html?projects/tac_plus.html.

Для простоты, мы предполагаем, что маршрутизатор, сервер TACACS+ и устройство с которого производится SSH-подключение находятся в одной подсети.

Все настройки для аутентификации и авторизации пользователей сервера TACACS+ содержатся в файле `/etc/tacacs+/tac_plus.conf`.

Мы рассмотрим лишь несколько основных параметров групп `group = groupname { ... }` и пользователей `user = username { ... }`.

Действия на сервере TACACS+:

В файле `/etc/tacacs+/tac_plus.conf` создайте несколько групп и пользователей для дальнейшей работы:

```
# Создание группы с максимальными привилегиями
```

```
group = for_enable_group {  
    default service = deny  
    service = er-exec {  
        priv-lvl = 15  
        shell:roles=admin  
    }  
}
```

```
group = cool_user {  
    default service = deny  
    service = er-exec {  
        priv-lvl = 15  
        shell:roles=admin_test  
    }  
}
```

```
# Создание группы с минимальными привилегиями
```

```
group = operator {  
    default service = deny  
    service = er-exec {  
        priv-lvl = 0  
        shell:roles=noc  
    }  
}
```

```
user = ecouser {  
    member = cool_user  
    cmd = "show" {  
        permit ".*"  
    }  
    login = cleartext ecopass  
}
```

```
user = operator1 {  
    member = operator  
    login = cleartext justpass
```



```
}

user = for_enable_user {
    member = for_enable_group
    cmd = "show" {
        permit ".*"
    }
    cmd = "enable" {
        permit ".*"
    }
    login = cleartext for_enable_user_pass
}
```

Все роли заданные в группах пользователей должны присутствовать в локальной базе данных маршрутизатора.

group = operator — наименование группы.

- **default service = {permit | deny}** — разрешить или запретить сервис, явно не указанный в профиле пользователя.
- **service = er-exec** — командная строка ЭкоРоутера.
 - **priv-lvl = 0** — соответствует режиму user-exec консоли ЭкоРоутера.
 - **priv-lvl = 15** — соответствует режиму enable-exec консоли ЭкоРоутера.
 - **shell:roles=admin_test** — привязка роли настроенной на ЭкоРоутере.

login = {cleartext | crypt} <PASSWORD> :

- **cleartext** — параметр для нешифрованного текстового пароля.
- **crypt** — параметр для зашифрованного пароля (DES и MD5 - алгоритмы).
- **PASSWORD** — пароль, зашифрованный или нешифрованный текст.

member — членство пользователя в группе. Может быть задано несколько групп к которым будет принадлежать пользователь.

cmd = "show" { permit ".*" } — авторизация команд, разрешает данному пользователю использование команды **show** с любыми последующими параметрами.

Задайте IP-адрес для TACACS-сервера: **ip a a 10.10.10.10/24 dev eth0 .**

Проверьте конфигурацию, перезапустите и проверьте работу сервера.

```
service tacacs_plus check
service tacacs_plus --full-restart
service tacacs_plus status
```

Действия на маршрутизаторе:

- Установите пароль "111" для входа в административный режим.
- Настройте IP-адрес управляющего порта.
- Задайте приоритет сервера TACACS+ над локальной авторизацией.

```
enable secret 111
hw mgmt ip 10.10.10.1/24
aaa precedence tacacs local
```

- Настройте подключение к серверу TACACS+:

```
aaa tacacs-server 10.10.10.10 port 49 vrf management auth
```

- Далее, создадим двух локальных пользователей и роль:

```
username operator1
password justpass
role admin
exit

username log
password log
role admin
exit

role admin_test
permit user-exec enable
permit user-exec show
permit enable-eenxec show
end
```

Проверка пользователя заданного на TACACS+:

- Подключитесь к маршрутизатору `ssh ecouser@10.10.10.1` и введите пароль **ecopass**.
- Убедитесь что вы зашли как пользователь **ecouser** и вам присвоена роль **admin_test** командой `show users connected`.
- Перейдите в административный режим командой `enable`, введите заданный выше пароль **111**.
- Введите команду `configure` и убедитесь что у данного пользователя нет доступа в конфигурационный режим.

Обратите внимание, что пользователь **ecouser** не задан локально в маршрутизаторе, но при этом ему присвоена локально заданная роль **admin_test** — всё как и описано в конфигурации на сервере TACACS+.

Проверка локального пользователя не заданного на TACACS+:

- Подключитесь к маршрутизатору `ssh admin@10.10.10.1` и введите пароль **admin**.

Пароль не будет принят, поскольку на сервере TACACS+ не задан пользователь с таким именем.

Создание пользователя admin на TACACS+:

В конфигурационном файле `/etc/tacacs+/tac_plus.conf` создайте пользователя **admin**.

```
user = admin {  
    member = cool_user  
    login = cleartext admin  
}
```

Проверьте конфигурацию, перезапустите и проверьте работу сервера.

```
service tacacs_plus check  
service tacacs_plus --full-restart  
service tacacs_plus status
```

Повторите вход `ssh admin@10.10.10.1` с паролем **admin**. Теперь вход на маршрутизатор прошёл успешно.

Обратите внимание что права пользователя **admin** теперь отличаются (вход в конфигурационный режим закрыт) от локального пользователя с тем же именем, поскольку его права определяются ролью **admin_test** заданной в группе **cool_user** на сервере TACACS+.

Проверка приоритета пользователя:

Пользователь с именем **operator1** задан как на сервере TACACS+ так и локально.

- Подключитесь к маршрутизатору `ssh operator1@10.10.10.1` и введите пароль **justpass**.
- Войдите в административный режим командой `enable` с паролем **111**, попытайтесь войти в конфигурационный режим командой `configure` на что будет получен отказ.

Следовательно пользователь настроенный на сервере TACACS+ получил приоритет и пользователю **operator1** задана роль **noc** не позволяющая входить в

конфигурационный режим.

Проверка пользователя с ролью администратора:

- Подключитесь к маршрутизатору `ssh for_enable_user@10.10.10.1` и введите пароль **for_enable_user_pass**.
- Войдите в административный режим командой `enable` с паролем **111**, войдите в конфигурационный режим командой `configure`.
- Отключите пароль на вход в административный режим `no enable secret`.
- Отмените заданный приоритет авторизации `no aaa precedence tacacs local`.
- Задайте приоритет заново `aaa precedence tacacs local`.

Подобные действия доступны только для пользователя с ролью администратора, следовательно действует роль **admin**, заданная в группе **for_enable_group** в конфигурации сервера TACACS+, к которой принадлежит пользователь `for_enable_user`.

Проверка покомандной авторизации:

- На маршрутизатор задайте новое правило для связи с сервером:

```
no aaa tacacs-server 10.10.10.10 port 49 vrf management auth
aaa tacacs-server 10.10.10.10 port 49 vrf management auth cmd-authorize
```

Теперь пользователю будут доступны лишь те команды, которые авторизует сервер TACACS+.

- Подключитесь к маршрутизатору `ssh for_enable_user@10.10.10.1` и введите пароль **for_enable_user_pass**.
- Введите команду `sh users connecte | no` — команда выдаёт нужные данные, при этом в слове **connected** пропущена последняя буква, т.е. TACACS+ готов авторизовать команды даже по неполному их написанию.
- Введите команду `enable`, такая команда задана и будет авторизована TACACS+.
- Введите команду `configute`, будет выдано сообщение "% Authorization via: tacacs command deny". Не смотря на то что роль пользователя **for_enable_user** = **admin** мы не можем войти в конфигурационный режим поскольку команда не авторизована.
- Введите команду `pin mgmt 1.1.1.1`, снова сообщение об отказе в авторизации. Таким образом, мы подтвердили работу покомандной авторизации сервера TACACS+, в том числе по неполному написанию команды.

При этом команды `end` и `exit` не отправляются на авторизацию и доступны всем пользователям из любого режима, это позволяет избежать ситуаций, когда пользователь может "застрять" в одном из режимов.

3.10 Внешние процедуры AAA по протоколу RADIUS

RADIUS (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. Описан в стандартах RFC 2058, RFC 2059, RFC 2865 и RFC 2866.

RADIUS часто используется провайдерами для управления доступом пользователей. В данном подразделе мы рассмотрим RADIUS, как средство для процедур AAA в части доступа к маршрутизатору по протоколу SSH.

Для настройки подключения к RADIUS-серверу используется команда `aaa radius-server <IP> [port <NUM>] secret <PASS> [vrf <NAME>] [source <IP2>] {account | auth} [timeout <0-300>]`, где:

- `server <IP>` — IP-адрес RADIUS-сервера.
- `port <NUM>` — номер порта на сервере.
- `secret <PASS>` — пароль для соединения с RADIUS-сервером (должен совпадать с установленным на RADIUS).
- `vrf <NAME>` — наименование VRF, в котором задан IP-адрес сервера (значение по умолчанию — VRF текущего виртуального маршрутизатора)
- `source <IP2>` — исходящий адрес.
- `account | auth` — выбор режима взаимодействия с сервером: **account** — только отправка учёта действий пользователя, **auth** — только аутентификация и авторизация. Для обеспечения всех функций AAA будет необходимо задать две команды `aaa radius-server`.
- `timeout <0-300>` — время в секундах в диапазоне от 0 до 300. Время ожидания до повторного запроса в случае отсутствия ответа от TACACS.

Пример:

```
aaa radius-server 10.1.1.1 port 1812 secret pass1234 vrf management source 10.1.1.2 auth
```

3.10.1 Конфигурирование RADIUS

Для простоты, мы предполагаем, что маршрутизатор сервер RADIUS и устройство с которого производится SSH-подключение находятся в одной подсети.

Для подключения к RADIUS-серверу, в среде ОС Linux сервера следует задать IP-адрес: `ip a a 10.1.1.100/24 dev eth0`.

Для минимальной конфигурации RADIUS-сервера достаточно отредактировать два файла.

В файле **/etc/raddb/clients.conf** после строки **"# IPv6 Client"** следует ввести следующие строки:

```
client router {
    ipaddr = 10.10.10.1
    secret = pass1234
}
```

Здесь мы задаём IP-адрес и пароль клиентского устройства — нашего маршрутизатора и пароль для входа на сервер. Такой же пароль будет установлен на маршрутизаторе.

В самом начале файла **/etc/raddb/mods-config/files/authorize** следует ввести следующие строки соблюдая табуляции:

```
ecotest Cleartext-Password := "ecotestpass"
Cisco-AVPair = "shell:roles=noc"
```

Здесь мы задаём пользователя, его пароль, а также устанавливаем роль.

Проверьте конфигурацию RADIUS-сервера командой `radiusd -X`. Если нет выдачи об ошибках, RADIUS-сервер начнёт свою работу.

На маршрутизаторе мы настроим IP-адрес для управляющего порта, установим приоритет аутентификации и авторизации и настроим подключение к RADIUS-серверу.

```
hw mgmt ip 10.10.10.1/24
aaa precedence tacacs radius local
aaa radius-server 10.10.10.17 port 1812 secret pass1234 vrf management source
10.10.10.1 auth
```

После чего создадим локального пользователя с ролью администратора:

```
username ecotest
password ecotestpass
role admin
```

Проверьте доступность RADIUS-сервера командой `ping 10.10.10.17 source 10.10.10.1 vrf management`.

Отключите другие AAA-серверы или задайте на маршрутизаторе приоритет для RADIUS-сервера командой: `aaa precedence radius tacacs local`

Для проверки работы авторизации с помощью RADIUS-сервера зайдите со стороннего устройства на маршрутизатор с помощью утилиты SSH: `ssh ecotest@10.10.10.1` и введите пароль **ecotestpass**.

Командой `enable` перейдите в административный режим командной строки маршрутизатора. Введите команду `configure`. Если вам отказано в доступе с формулировкой "% User 'ecotest' has no permission to execute 'enable-execs>configure ' %", значит работает роль **noc** заданная на RADIUS-сервере для пользователя **ecotest**.

Повторите действия при отключенном RADIUS-сервере. У вас должен появиться доступ в конфигурационный режим, поскольку авторизация будет проходить согласно внутренним установкам роли.

3.11 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 задаётся 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 14 — Параметры команды задания правил профиля безопасности

Параметр	Описание
<code><0-1023></code>	Порядковый номер правила, от 0 до 1023. Правила применяются, начиная с 0 по 1023.
<code>permit</code> \ <code>deny</code>	Тип правила: разрешить (permit) или запретить (deny)
<code>PROTOCOL</code>	Пакеты какого протокола подпадают под это правило. Может быть указан номер протокола по спецификации IANA от 0 до 255 или одно

Параметр	Описание
	<p>из следующих обозначений:</p> <ul style="list-style-type: none"> - 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	<p>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></p>
DESTINATION	<p>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></p>
DEST PORT	<p>Вариант фильтрации. Указывается одно из следующих обозначений:</p> <ul style="list-style-type: none"> - eq — номер порта равен ..., - gt — номер порта больше, чем ..., - lt — номер порта меньше, чем ..., - range — номер порта находится в диапазоне ...
DP NUMBER	<p>Номер или обозначение порта.</p> <p>Возможные значения для TCP:</p> <ul style="list-style-type: none"> - номер порта от 0 до 65535, - ftp — FTP (21 порт), - ssh — SSH (22 порт), - telnet — Telnet (23 порт), - www — WWW (HTTP, 80 порт). <p>Возможные значения для UDP:</p> <ul style="list-style-type: none"> - номер порта от 0 до 65535, - bootp — BOOTP (67 порт), - tftp — TFTP (69 порт).

Параметр	Описание
	Если задается диапазон портов (range), то нижняя и верхняя граница диапазона указываются числами через пробел.

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

В EcoRouter существует жёстко 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
```

3.11.1 Management порт и виртуальные маршрутизаторы

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 (default), 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 порте 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**.

3.11.2 Пример настройки профиля безопасности

В примере все команды кроме команд группы **show** выполняются в конфигурационном режиме.

Создание профиля безопасности и правил:

```
security-profile 1
rule 0 permit tcp any any eq 23
rule 1 deny udp any any eq 67
rule 2 deny ospf host 127.0.0.12 any
rule 3 deny tcp any 192.168.10.2/24 range 21 23
do show security-profile
```

Прямо в процессе создания правил, можно удалить ненужное или ошибочное правило командой **no rule <NUM>**, где **NUM** — номер правила.

Команда **do show security-profile** выведет на консоль все профили безопасности включая только что созданный.

Создание VRF и привязка профиля безопасности:

```
# Создадим VRF с именем vrf0 и тут же завершим работу с VRF.
ip vrf vrf0
exit

# Привяжем профиль безопасности 1 к vrf0
security 1 vrf vrf0
end

# Проверим результат в выдаче команды show
show ip vrf

VRF default
...
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.

```
# Зайдём в редактирования профиля безопасности 1
security-profile 1
rule 4 permit any any any

# При попытке ввести правило, получим сообщение об ошибке.
# % Profile is set on 1 namespaces. Unset profile prior to change it.
# Для внесения изменений необходимо отвязать профиль безопасности от VRF.

# Выходим в конфигурационный режим
exit

# Отвязываем профиль безопасности от VRF,
# редактируем профиль и добавляем правило:
no security 1 vrf vrf0
security-profile 1
rule 4 permit any any any
exit

# Снова привязываем профиль безопасности к VRF:
ecorouter(config)#security 1 vrf vrf0
ecorouter(config)#end

# Проверим
ecorouter#show ip vrf
...
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
```



```
# К vrf0 привязан профиль безопасности 1 со всеми четырьмя правилами.
```

Удаление профиля безопасности (конфигурационный режим).

```
# Удаляем профиль безопасности и vrf0
no security 1 vrf
no ip vrf vrf0
end
```

```
# Проверяем результат
ecorouter#show ip vrf
VRF default
  Interfaces:
  Security profile default
...
  permit any any any

VRF management
```

3.11.3 Обработка ICMP echo request пакетов

Обработка ICMP echo request пакетов (ответ на ping) по умолчанию осуществляется в data-plane и не учитывает профили безопасности.

Для применения профилей безопасности к ICMP echo request пакетам необходимо выполнить следующую команду конфигурационного режима:

```
icmp-echo control-plane
```

После выполнения этой команды обработка ICMP echo request пакетов будет осуществляться в control-plane, правила профилей безопасности будут учтены.

Для исключения обработки ICMP echo request пакетов из действия профилей безопасности необходимо выполнить следующую команду конфигурационного режима:

```
console no icmp-echo control-plane
```

4 Виды портов и интерфейсов

4.1 Порт

Порт (port) — это стандартизированный разъём в составе EcoRouter для его подключения к физическим линиям связи. 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 в режим конфигурации определённого порта. Где **te1** — его имя:

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

Для административного (ручного) 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 соединения. Если порт выключен средствами системы, то в выводе статистики по портам его состояние обозначается как "administratively down".

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

Пример (конфигурационный режим):

```
do show port brief
```

Name	Physical	Admin	Lacp	Description
te0	UP	UP	*	
te1	DOWN	UP	*	
te2	DOWN	UP	*	
te3	DOWN	UP	*	

```
port te2
```

```
shutdown
```

```
do show port brief
```



Name	Physical	Admin	Lacp	Description
------	----------	-------	------	-------------

te0	UP	UP	*	
te1	DOWN	UP	*	
te2	DOWN	DOWN	*	
te3	DOWN	UP	*	

```
no shutdown
do show port brief
```

Name	Physical	Admin	Lacp	Description
------	----------	-------	------	-------------

te0	UP	UP	*	
te1	DOWN	UP	*	
te2	DOWN	UP	*	
te3	DOWN	UP	*	

4.2 Агрегирование каналов

Агрегирование каналов — объединение нескольких каналов в один логический канал для увеличения пропускной способности и резервирования. Чтобы добавить порты в объединённый канал они должны быть идентично настроены и параллельно соединять два устройства.

В один агрегированный порт могут быть объединены до 16 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 порте (см. раздел Сервисные интерфейсы).

4.3 Интерфейс

Интерфейс (interface) — это логический интерфейс для адресации L3. Название интерфейса задаётся 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 точка ".".



В EcoRouter существуют специальные L3-интерфейсы, которые служат для поддержки определённого функционала (bmi-интерфейсы, tunnel-интерфейсы, loopback-интерфейсы и т.д.) и называются соответственно. В качестве имени обычных логических интерфейсов для адресации L3 нельзя использовать названия специальных интерфейсов (ВСЕ ИМЕНА РЕГИСТРОЗАВИСИМЫЕ):

- **loopback.<номер>**,
- **pppoe.<номер>**,
- **Null**,
- **vlan**.

Базовая настройка интерфейса происходит в конфигурационном режиме:

Создание интерфейса, где **NAME** — произвольное имя с учётом ограничений описанных выше.

```
interface NAME
```

Общий вид командной строки при конфигурировании интерфейса (режим контекста конфигурирования интерфейса).

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

Назначение IP-адреса с маской подсети.

```
# Вариант 1
ecorouter(config-if)#ip address 10.10.10.1/24
# Вариант 2
ecorouter(config-if)# ip address 10.10.10.1 255.255.255.0
```

Назначение статического MAC-адреса.

```
ecorouter(config-if)# static-mac 1c87.7640.fa02
```

При этом базовый MAC-адрес сохраняется в памяти (его можно посмотреть при помощи команды **show interface <NAME>**). Для возврата к базовому MAC-адресу используется команда **no static-mac**.

Включение интерфейса.

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

Выключение интерфейса.

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

4.4 ICMP параметры интерфейса

Для каждого интерфейса можно задать ряд параметров влияющих на работу ICMP протокола. Все команды ниже вводятся в контекстном режиме настройки интерфейса `EcoRouter(config-if)#`.

- Команда `icmp redirects` включает, а команда `no icmp redirects` выключает отправку ICMP redirect-сообщений.

ICMP redirect-сообщения используются для информирования отправителя о том, что он должен отправлять пакеты не через текущий маршрутизатор, а по другому, более оптимальному маршруту. Это происходит в случае, если маршрутизатор обнаруживает, что существует более короткий путь к адресу получателя или следующему узлу и этот адрес находится за L3 интерфейсом через который пришёл пакет.

- Команда `icmp ttl-exceeded` включает, а команда `no icmp ttl-exceeded` отключает отправку "ttl-exceeded" — сообщений.

EcoRouter отбрасывает транзитные (не адресованный непосредственно данному маршрутизатору) пакеты с истёкшим "временем жизни" TTL (Time To Live) , т.е. $TTL \leq 1$ и по умолчанию отправляет источнику пакета сообщение о том, что TTL истёк.

Более гибкая настройка работы с TTL возможна с помощью правил списков доступа, как это описано в пункте "Настройка L3 filter-map" раздела "Списки доступа".

- Команда `icmp unreachable` включает, а команда `no icmp unreachable` отключает отправку сообщений о недоступности (unreachables).

Сообщения о недоступности отправляются маршрутизатором для уведомления отправителя о том, что пакет данных не может быть доставлен в указанное место назначения. Недоступными могут быть: сеть, конкретный хост, протокол или порт.

Готовность маршрутизатора отправлять сообщения даёт злоумышленникам возможность для атак с помощью массовой отправки на роутер пакетов с которыми ему нужно что-то сделать:

- пакетов по которым заранее известен более оптимальный маршрут, чтобы маршрутизатор отвечал сообщениями `redirects`,
- пакетов с $TTL \leq 1$, чтобы маршрутизатор отвечал сообщениями `ttl-exceeded` ("TTL expiry attack"),

- пакетов с недостижимым адресом назначения, чтобы маршрутизатор отвечал сообщениями unreachable.

Цель подобных "рассылок" проста — вынудить маршрутизатор тратить вычислительные мощности на обработку этих пакетов и на отправку сообщений. Тратить их в таком количестве, чтобы поглотить все ресурсы CPU и сделать невозможной дальнейшую эффективную работу маршрутизатора. Такие атаки получили название DoS-атак.

Самым простым и очевидным действием для предотвращения самой возможности подобных атак является отключение отправки маршрутизатором сообщений по вышеозначенным поводам.

Существуют и другие поводы для отключения ICMP сообщений:

- злоумышленник может отправить ICMP Redirect, чтобы перенаправить трафик через свое устройство
- уменьшение нагрузки на сеть путём уменьшения потоков ICMP данных,
- злоумышленники могут использовать сообщения TTL-exceeded для определения активных устройств в сети
- сообщения о недоступности могут раскрывать информацию о структуре и состоянии сети,
- и другие аспекты в основном связанные с вопросами безопасности.

Однако не стоит думать, что отправка перечисленных ICMP сообщений бесполезное дело. ICMP сообщения улучшают визуализацию процесса маршрутизации и уменьшают время устранения неисправностей в IP сети. Однако, в сегментах сети, которые находятся в непосредственной близости к конечным абонентам и направлены в сторону внешних сетей, желательно отправки таких ICMP сообщений отключить по завершению настроек IP сети, и в будущем включать лишь по необходимости.

4.5 Максимальный передаваемый элемент данных (MTU)

Стандартом де-факто для протокола Ethernet является размер кадра в 1514 байт. Но в ряде случаев данное значение может быть значительно превышено. EcoRouterOS реагирует на превышение MTU следующим образом:

- При превышении максимальных значений MTU, установленных для входящего порта и (или) интерфейса, полученный элемент данных отбрасывается без всяких условий.



- При превышении максимальных значений заданных для исходящего интерфейса возможны два варианта действий системы:
 - Пакет разбивается на более мелкие, в соответствии с ограничениями выходного интерфейса.
 - Если у входящего пакета в IP-заголовке установлен df-bit, роутер формирует и отправляет в адрес источника ICMP-сообщение "fragmentation needed" с указанием требуемого значения MTU. При этом предполагается, что в дальнейшем источник будет отправлять пакеты нужного размера (PMTUD).

Установка значения L2 MTU для кадров доступна в настройках каждого порта:

```
ecorouter(config)#port te1
ecorouter(config-port)#mtu 9728
```

Возможные значения — от 1347 до 9728 байт.

Установка значения L3 MTU для пакетов доступна в настройках каждого интерфейса:

```
ecorouter(config)#interface te0
ecorouter(config-if)#ip mtu 1500
```

Максимальное значение MTU L3 интерфейса соответствует максимальному значению MTU L2 за вычетом размера заголовка L2 уровня.

Значение MTU для многих сетевых протоколов не превышает 1522, однако в EcoRouter существует возможность задать значение MTU в пределах от 1347 до 9728. Таким образом становится возможным использование кадров типа Jumbo frame и других нестандартных размеров.

4.6 Интерфейс loopback

Интерфейс loopback (Loopback Interface) — это виртуальный L3 loopback-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.<NAME>`.

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:

Создание интерфейса loopback. Где **NAME** — произвольный номер.

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

Назначение IP-адреса с префиксом.

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

Или назначение IP-адреса с маской подсети.

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

Команда включения интерфейса.

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

Команда выключения интерфейса.

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

4.7 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.

4.8 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.

Таблица 15 — Базовая настройка Bridge Domain Interface

Команда	Описание
<code>interface <NAME></code>	Создание интерфейса бридж домена. Где NAME — произвольное имя
<code>ip address <IP><MASK></code>	Назначение IP-адреса с маской подсети
<code>connect to bridge <NAME></code>	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
```

4.9 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 порте может существовать много сервисных интерфейсов, которые будут обрабатывать разные метки VLAN'ов по-разному.

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

Название субинтерфейса задаётся администратором. В каждой строке service instance может содержаться только один признак трафика.

Пример:

Сервисный интерфейс создаётся в режиме конфигурации порта.

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

Создание сервисных интерфейсов.

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

Указание номера, обрабатываемого VLAN.

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

Указание выполняемой операции.

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

Указание в какой интерфейс нужно отправить обработанные кадры.

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

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
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 te1/QQ1 is up
```

Просмотр состояния и конфигурации определённого 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 определённого интерфейса:

```
ecorouter#show interface e3
Interface e3[10] is up, line protocol is up
Snmp index: 7
Ethernet address: 1234.ab00.00ff (configured)
Base MAC: 1c87.7640.fa02 (not in use)
MTU: 1500
NAT: no
ICMP redirection is on
```



```
Label switching is disabled
<UP,BROADCAST,RUNNING,MULTICAST>
Connect port te0 service instance te0/e1 symmetric
inet 100.200.200.253/31
total input packets 156, bytes 14976
total output packets 156, bytes 14976
```

Краткий вывод статусов всех интерфейсов:

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

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/SFP+/QSFP+/QSFP28 -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 так же, как и для других команд **show**.

Для SFP-модулей выводится следующая информация:

- Module Type — Тип передатчика. Примеры:
 - 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;
 - 40GBASE-LR - модуль стандарта 40GBASE-LR - 40 Гбит/с, максимальная длина сегмента 10 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.
- **Module distance** — Максимальная поддерживаемая длина для кабеля в метрах/километрах. Значения выводятся для определённой кабельной линии в зависимости ее типа: медный (Copper), оптический одномодовый (SMF), оптический многомодовый в соответствии стандартам ISO (OM1, OM2, OM3).
- **Tx/RX avg optical power** — Уровень оптической мощности в дБм. Актуальные значения выводятся для каналов передачи и приема. При поддержке нескольких отдельных оптических каналов (QSFP) уровень будет показан для отдельно для каждого.

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

Пример вывода информации для порта без SFP+ модуля:

```
ecorouter#show transceiver
Port: te0
Module doesn't identify itself as SFF-compatible
```

Пример вывода информации для порта с QSFP+ модулем:

```
ecorouter#show transceiver
Port: qe0/0
Module Type: 40G_Base-LR4
Module Vendor Name: YN
Module Part Number: 40G010QPN
Module Serial Number: 202012210090
Module Revision: 1A
Module Manufacturing Date: 210122
Module supports DDM: yes
Module temperature: 28.00 C
Module voltage: 3.27 V
Module distance SMF: 10 km
Module distance OM3: 0 m
Module distance OM2: 0 m
Module distance OM1: 0 m
Module distance copper or active cable: 0 m
```

```
Tx avg optical power (Channel 1): 1.2019 mW / 0.80 dBm
Rx avg optical power (Channel 1): 0.0944 mW / -10.25 dBm
Tx avg optical power (Channel 2): 1.2317 mW / 0.91 dBm
Rx avg optical power (Channel 2): 0.0944 mW / -10.25 dBm
Tx avg optical power (Channel 3): 1.3010 mW / 1.14 dBm
Rx avg optical power (Channel 3): 0.0753 mW / -11.23 dBm
Tx avg optical power (Channel 4): 1.3301 mW / 1.24 dBm
Rx avg optical power (Channel 4): 0.0634 mW / -11.98 dBm
...
```

5 Сервисные интерфейсы

При входе на порт кадр с меткой VLAN будет помещён в сервисный интерфейс, выделенный для обработки данной метки VLAN. После сервисный интерфейс может заменить, добавить или снять метку VLAN 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 порте по признаку инкапсулированного в него тега dot1q или по его отсутствию. На одном порте 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 16 — Параметры команды **encapsulation**

Вид инкапсуляции	Описание
encapsulation untagged	Нетегированные кадры
encapsulation default	Указание на то, что данный сервисный интерфейс будет обрабатывать все теги, которые не обрабатываются другими сервисными интерфейсами на данном порте.



Вид инкапсуляции	Описание
	Применяется в L3 бриджах и в соединениях без участия L3 маршрутизации.
<code>encapsulation dot1q any</code>	Инкапсуляция IEEE 802.1q с любым тегом в кадре.
<code>encapsulation dot1q <TAG></code>	Инкапсуляция IEEE 802.1q с конкретным тегом в кадре.
<code>encapsulation dot1q <TAG> second-dot1q <TAG></code>	Инкапсуляция IEEE 802.1q с 2-мя тегами, содержащимися в кадре. Значения тегов указываются по порядку начиная с внешнего.
<code>encapsulation dot1q <TAG1>-<TAG2></code>	Инкапсуляция IEEE 802.1q с диапазоном тегов.
<code>encapsulation dot1q {<TAG> <TAG1>-<TAG2>} exact</code>	Аргумент exact указывает на то, что данный сервисный интерфейс будет принимать кадры только с одним указанным тегом или всеми тегами из диапазона, но только с одним тегом в кадре. Данный аргумент идентичен поведению по умолчанию.
<code>encapsulation dot1q {<TAG> <TAG1>-<TAG2>} non-exact</code>	Аргумент non-exact указывает на то, что данный сервисный интерфейс будет принимать кадры только с указанным тегом или всеми тегами из диапазона, но допускает наличие двух тегов в кадре.

5.2 Операции над метками

После того, как кадр с определённой меткой был направлен в определённый 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 уровне L3 (исключение интерфейс BDI) 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 17 — Варианты команд операций над метками

Вид операции над меткой	Описание
<code>rewrite pop <VALUE></code>	Операция снятия метки. VALUE равен 1 или 2.
<code>rewrite push <VALUE> <VALUE></code>	Добавление метки. VALUE значение метки. Верхняя метка — первая.
<code>rewrite translate 1-to-1 <VALUE></code>	Замена одной метки на другую. Где VALUE значение новой метки.
<code>rewrite translate 1-to-2 <VALUE> <VALUE></code>	Замена одной метки на две других.
<code>rewrite translate 2-to-2 <VALUE> <VALUE></code>	Замена двух меток на две других.
<code>rewrite translate 2-to-1 <VALUE></code>	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 присоединённому 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 от порта к интерфейсу метка 3 будет сниматься, при движении от интерфейса к порту — 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.

5.3 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.

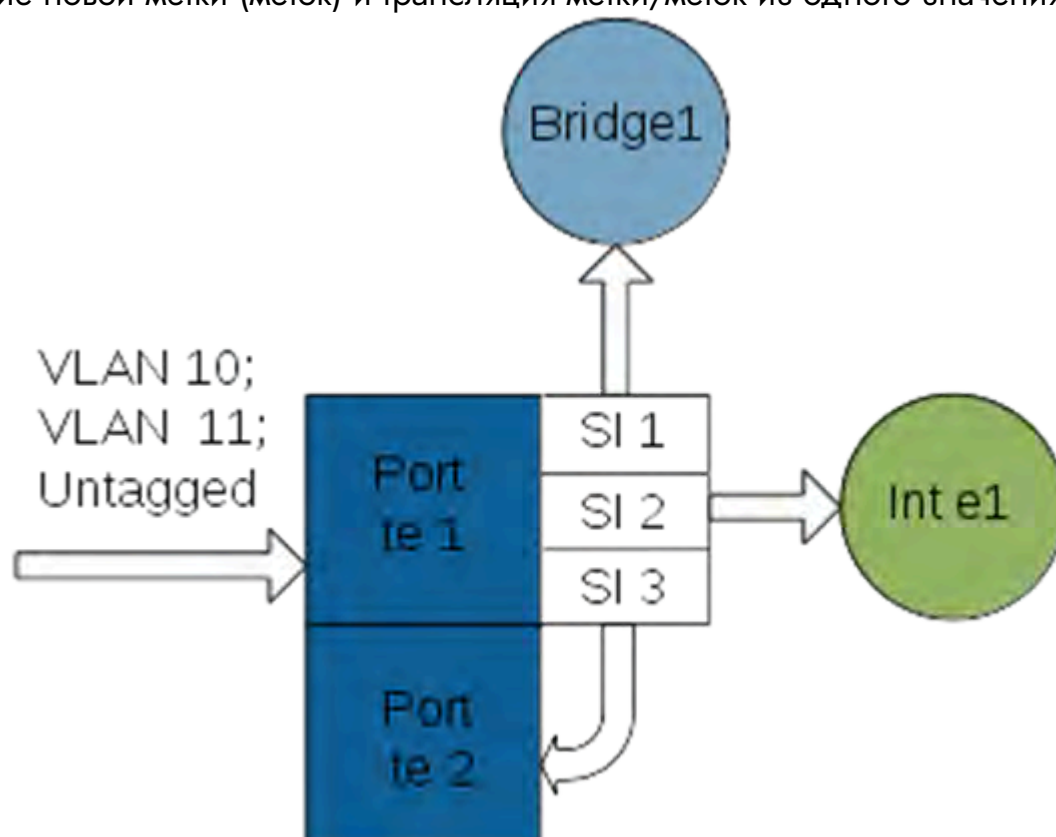


Рисунок 5

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

5.3.1 Трансляция меток

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

На порте, куда приходит VLAN 10, создаём сервисный интерфейс для операции с этими метками.

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

Из всего объёма 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.

5.3.2 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
```

Здесь мы 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.

5.3.3 Добавление меток

Весь нетегированный трафик обрабатываем 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**.

5.3.4 Снятие меток

В service-instance 2 будем обрабатывать VLAN 11 на порте te1. Создаём 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.

5.3.5 Настройка service instance для маршрутизации 2 VLAN'ов

Имеется следующая схема сети.

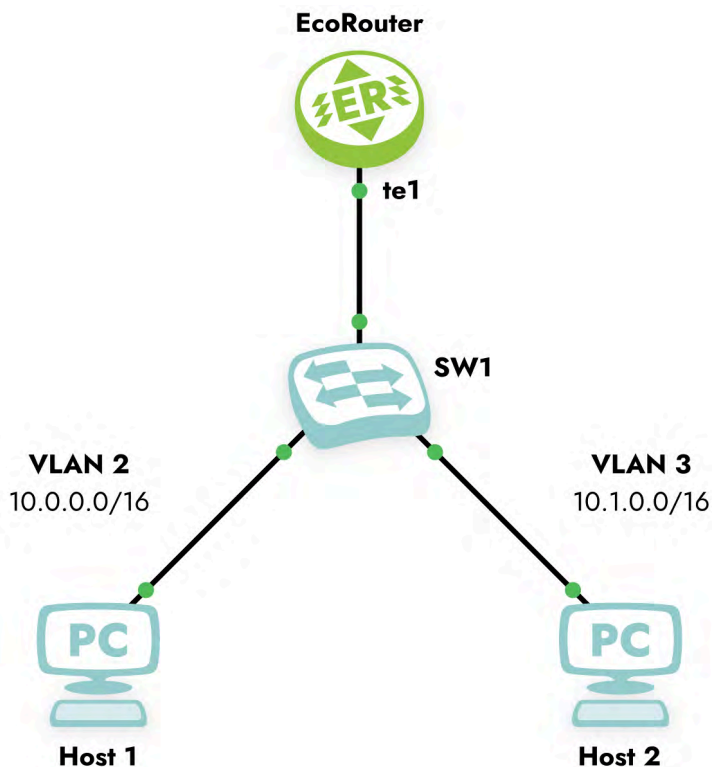


Рисунок 6

Шаг 1. Создаём 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. Создаём service-instance на порте 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 порте 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.

5.3.6 Настройка сервисного интерфейса для функционирования EcoRouter в роли L2 устройства

Имеется следующая схема сети.

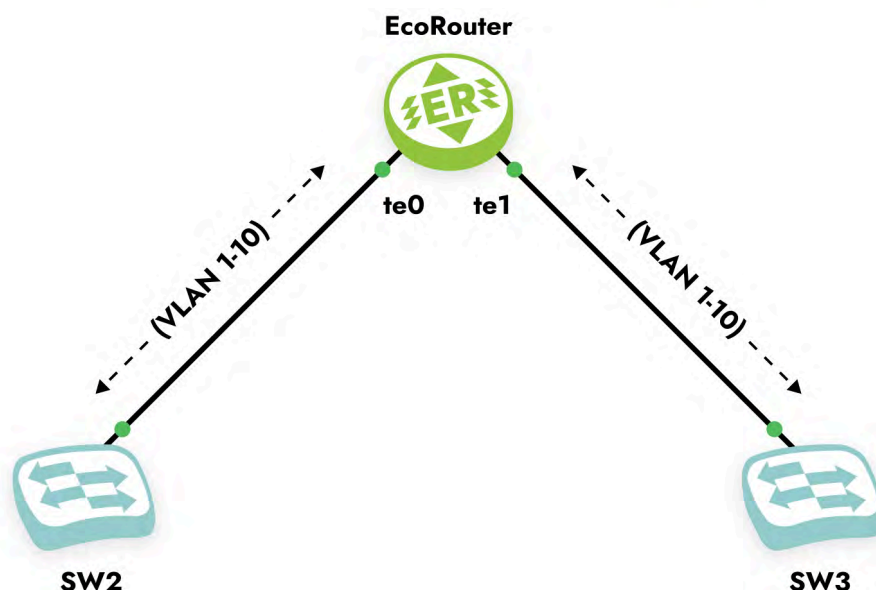


Рисунок 7

Шаг 1. Создаём service-instance на порте 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 порте 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 ещё 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 чёткого понимания происходящих операций над пакетами на интерфейсах и портах маршрутизатора. В CLI есть несколько команд группы **show** для просмотра сконфигурированных фильтров.

5.4 Просмотр настроек сервисных интерфейсов

5.4.1 Просмотр всех сервисных интерфейсов на всех портах

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

Ingress — описание порядка обработки кадра при движении через порт во входящем направлении. Как описано в сервисном интерфейсе администратором.

Egress — описание порядка обработки кадра при движении через порт в исходящем 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#
```

5.4.2 Просмотр сервисных интерфейсов на отдельном порте

Для просмотра настроек сервисных интерфейсов, имеющих на конкретном порте, используется команда `show port <NAME>` или ее сокращенная форма: `sh port <NAME>`.

```
ecorouter#sh port te1
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
ecorouter#
```

5.4.3 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 te1 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
```



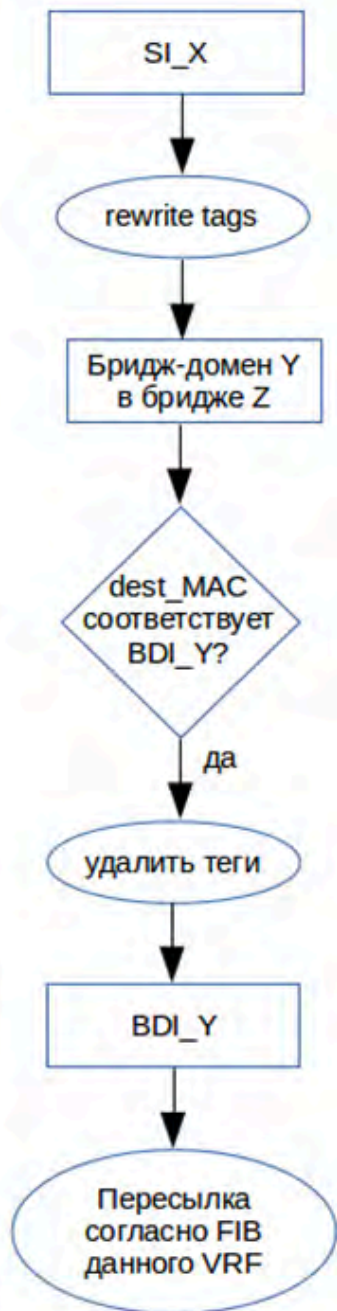
6 Бриджинг с поддержкой L3

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.

L2 домен → L3 домен



L3 домен → L2 домен



Рисунок 8

6.1 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)#`.

Таблица 18 — Команды конфигурирования сервисного интерфейса

Команда	Описание
<code>encapsulation {default dot1q untagged}</code>	Задание инкапсуляции (тегирования) для внешнего трафика
<code>rewrite {pop push translate}</code>	Преобразование инкапсуляции при отправке в бридж
<code>connect bridge <NAME></code>	Подключение к созданному ранее бриджу

Тегирование (инкапсуляция) может быть произвольным (см. раздел «Операции над метками в сервисных интерфейсах»), причём, как сказано выше, VLAN ID сервисного интерфейса на одном порте может совпадать с VLAN ID сервисного интерфейса на другом порте, и это будут разные VLAN, до тех пор, пока эти SI находятся в разных бридж-доменах. Бридж-домен на бридже образуют подключенные к нему сервисные интерфейсы с одинаковым значением инкапсуляции на бридже, 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
```

6.2 Создание BDI

Интерфейс BDI создается как обычный L3 интерфейс с двумя дополнительными командами в контексте конфигурирования интерфейса, описанными в таблице ниже.

Таблица 19 — Команды создания BDI-интерфейса

Команда	Описание
<code>rewrite push</code>	Преобразование инкапсуляции при отправке в бридж
<code>connect bridge <NAME></code>	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**.

6.3 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 bdi1 symmetric
```

Для просмотра информации об интерфейсах BDI используется стандартная для всех интерфейсов команда `show interface <BDI_NAME>`.

```
ecorouter#show interface bdi1
Interface bdi1 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 приведённом примере показаны следующие параметры и их значения:

- **L3 BDI address:** 192.168.1.1/24 — IP-адрес L3 интерфейса в данном бридже;
- **BD Aging time** — время устаревания для каждого mac-адреса в секундах;
- **Outer Vlan** — значение внешнего vlan, с которым был подключён пользователь;
- **Inner Vlan** — значение внутреннего vlan, с которым был подключён 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.

7 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.

7.1 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.

7.2 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 идёт копирование на TFTP сервер с IP-адресом 192.168.10.10, можно задать путь копирования одним из способов, описанных в таблице ниже.

Таблица 20 — Способы задания пути к TFTP серверу

Вариант записи пути	Расположение файла	Имя файла
tftp://192.168.10.10/	корневая директория сервера	по умолчанию
tftp://192.168.10.10/folder/	определённая 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	определённая директория	указанное имя файла, расширение по умолчанию



Вариант записи пути	Расположение файла	Имя файла
tftp://192.168.10.10/folder/name.res	определённая директория	указанное имя файла, указанное расширение

Приведённый пример демонстрирует гибкость задания пути при копировании архива конфигурации.

7.3 Архив конфигурации

При экспорте конфигурации по умолчанию создается архив с названием следующего вида: **startupbackupимяхоста_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.

7.4 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>
```

7.5 Экспорт конфигурации

В случае экспорта конфигурации происходит копирование из startup-config на FTP или TFTP сервер. При этом копируется последняя сохранённая 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>
```

Примеры команд экспорта конфигурации по FTP:

- Экспорт на указанный FTP сервер, параметры по умолчанию: `copy startup-config ftp ftp://user:password@192.168.10.10/` .
- Экспорт на указанный FTP сервер, имя архива задано: `copy startup-config ftp ftp://user:password@192.168.10.10/my_name_of_archive` .
- Экспорт на указанный FTP сервер, имя и расширение архива задано: `copy startup-config ftp ftp://user:password@192.168.10.10/my_name_of_archive.res` .
- Экспорт на указанный FTP сервер через виртуальный маршрутизатор по умолчанию `copy startup-config ftp .ftp://user:password@192.168.10.10/ vr default` .
- Экспорт на указанный FTP сервер через заданный виртуальный маршрутизатор `copy startup-config ftp .ftp://user:password@192.168.10.10/ vr VR1` .

Примеры команд экспорта конфигурации по TFTP:

- Экспорт на указанный TFTP сервер, параметры по умолчанию: `copy startup-config tftp tftp://192.168.10.10/` .
- Экспорт на указанный TFTP сервер, имя архива задано: `copy startup-config tftp tftp://192.168.10.10/my_name_of_archive` .
- Экспорт на указанный TFTP сервер, имя и расширение архива задано: `copy startup-config tftp tftp://192.168.10.10/my_name_of_archive.res` .
- Экспорт на указанный TFTP сервер через виртуальный маршрутизатор по умолчанию: `copy startup-config tftp tftp://192.168.10.10/ vr default` .
- Экспорт на указанный TFTP сервер через заданный виртуальный маршрутизатор: `copy startup-config tftp tftp://192.168.10.10/ vr VR1` .



7.6 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 21 — команды импорта конфигурации

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,

Команда	Описание
	параметры по умолчанию
<code>copy tftp startup-config tftp://192.168.10.10/my_name_backup vr default</code>	Импорт с указанного TFTP сервера через виртуальный маршрутизатор по умолчанию
<code>copy tftp startup-config tftp://192.168.10.10/startup_backup_EcoRouterOS_20160623175405.tar.gz vr VR1</code>	Импорт с указанного TFTP сервера через заданный виртуальный маршрутизатор
<code>copy tftp startup-config tftp://192.168.10.10/my_name_backup mgmt</code>	Импорт с указанного TFTP сервера через management-интерфейс

7.7 Пример резервного копирования конфигурации

После загрузки с помощью команды `enable` перейдите в административный режим.

Для создания резервной копии конфигурации введите команду:
`copy startup-config storage description Rezervnaya_kopiya` .

Обратите внимание, что записан будет именно `startup-config` , т.е. конфигурация с которой загрузился EcoRouter или последняя сохранённая командой `write memory` конфигурация.

Для просмотра списка резервных копий введите команду `show backups` :

```
ecorouter#show backups
```

Entry number	Filename format YYYY-MM-DD_HH-MM-SS	Description text
1.	startup_backup_LNS_2025-12-23_14-44-49.tar.gz	Rezervnaya_kopiya

Скопируйте резервную копию по протоколу SCP на нужный вам адрес:

```
copy scp backup startup_backup_LNS_2025-12-23_14-44-49.tar.gz
admin@10.0.0.1:/home/admin/
```

Удалите резервную копию командой `delete backup storage startup_backup_LNS_2025-12-23_14-44-49.tar.gz` .

Для удаления всех резервных копий используйте команду `delete backup storage`

*.

8 Операции с прошивкой

Предусмотрено 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 произведён 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 — её стабильность.

8.1 Скачивание образа прошивки

Для обновления image-прошивки предусмотрена возможность скачивания её с SCP, FTP или TFTP-сервера. Команды для скачивания описаны в таблице ниже.

Примеры команд для скачивания образа прошивки:

- С SCP-сервера из папки `/images` будет скачан образ прошивки с именем `filename` для обновления с текущей версии прошивки, SCP-сервер доступен через менеджмент-порт (mgmt): `copy scp image user@xxx.xxx.xxx.xxx:/images/filename 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/mgmt` .

- 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 ftp image ftp://user:password@xxx.xxx.xxx.xxx/filename vr default` .
- С TFTP-сервера будет скачан подходящий образ прошивки для обновления с текущей версии прошивки. EcoRouter сам 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/ vr vrname` .
- 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): `copy tftp image tftp://xxx.xxx.xxx.xxx/filename mgmt` .

В общем виде команда для скачивания образа прошивки маршрутизатора выглядит следующим образом:

```
copy <scp | 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 скаченных образах и их состоянии используется команда административного режима `show images storage` (для просмотра образов, размещённых на внутреннем накопителе устройства) или `show images usb` (для просмотра образов, размещённых на подключённых 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 прошёл проверку целостности,
`verification is failed` — образ не прошёл проверку целостности.

Соответственно, образы могут подходить для установки (suitable for installation) или не подходить (not suitable for installation) по разным причинам. В приведённом примере первый и второй образы не прошли проверку на зависимость версий, а третий несовместим с платформой ER-1004.

Предусмотрена возможность копирования данных по протоколу SCP.

Команды для скачивания образа прошивки по протоколу SCP:

- Копирование с сервера образа Docker-контейнера: `copy scp container <URL>` .
- Копирование с сервера образа прошивки: `copy scp image <URL>`
- Копирование с сервера образа виртуальной машины: `copy scp virtual-disk <URL>` .

URL для данной команды должен быть задан в формате: `<логин>@<адрес сервера>:<путь к файлу на сервере>` .

Например:

```
copy scp image admin@10.0.0.1:/home/admin/eco.image
```

8.2 Установка 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`.

8.2.1 Приоритет выбора прошивки для загрузки

При загрузке соблюдается следующий порядок выбора прошивки по убыванию приоритетов:

- Незаводская прошивка со статусом temporary.
- Незаводская прошивка со статусом active.
- Незаводская прошивка со статусом stable.
- Factory-прошивка.

8.3 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 22 — Обозначение в выводе команды `show running-config diff`

Значение	Описание
----------	----------



Значение	Описание
<pre>--- _line1, line2_ ---- **** line1, line2 ****</pre>	Диапазон номеров строк, где произошли изменения (--- для running конфигурации, *** для startup конфигурации)
+ text	Команда присутствует в running конфигурации, отсутствует в startup конфигурации
- text	Команда присутствует в startup конфигурации, отсутствует в running конфигурации
! text	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#
```

8.4 Удаление образа прошивки

Для того чтобы удалить файл image прошивки, который больше не будет использоваться, существует команда:

```
delete image storage <IMAGE_NAME>
```

, где **IMAGE_NAME** — один из образов, указанных в выводе команды `show images storage`.

Для удаления установленной прошивки существует команда:

```
delete image firmware <IMAGE-A/B>
```

, где **IMAGE-A/B** — установленная прошивка `a-image` или `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.

8.5 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> >
```

Где: **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.



8.6 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
771e77b5d1ffbf9db37b958d2ae2faab libpcr.so.1.2.7: OK
a6aa50ed7b77fc1fd06d8626d8b7d78c libpcr.la: OK
b9fd49b80acaf6173a22b7d5bb6b4f1c libpcrposix.so.0.0.4: OK
60f530c64889d00ad21dd15534e11dea libpcrposix.la: OK
b9f29f6dedee7bfddcc52d9cd3386e51e er-ripd-ns@-start: OK
Checksum validation PASSED
ecorouter#
```

8.7 Сброс до заводской версии ПО

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.

8.8 "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**.

9 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 включён 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.

Функционал протокола настраивается в конфигурационном режиме при помощи команд, представленных в таблице ниже.

Таблица 23 — Команды настройки протокола ARP

Команда	Описание
<code>arp <IP ADDRESS> <MAC ADDRESS></code>	Создание статической записи для конкретного IP-адреса
<code>arp vrf <VRF NAME> <IP ADDRESS> <MAC ADDRESS></code>	Создание статической записи для конкретного IP-адреса в заданном VRF
<code>arp expiration-period <0-300></code>	Настройка времени хранения динамической записи в ARP-таблице в минутах. Значение по умолчанию — 5 минут
<code>arp incomplete-time <5-300></code>	Настройка времени хранения incomplete записи в ARP-таблице в секундах. Значение по умолчанию — 60 секунд
<code>arp request-interval <0-100></code>	Задание интервала времени отправки ARP-запросов в секундах в случае отсутствия ARP-ответов. Значение по умолчанию — 1 секунда
<code>arp request-number <0-100></code>	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 24 — Параметры команды просмотра таблицы ARP-записей

Команда	Описание
<code>show arp</code>	Вывод полной ARP-таблицы
<code>show arp interface <INTERFACE NAME></code>	Вывод ARP-таблицы для записей, полученных с определенного интерфейса
<code>show arp ip <IP ADDRESS></code>	Вывод ARP-записи для определенного IP-адреса

Команда	Описание
<code>show arp mac <MAC ADDRESS></code>	Вывод ARP-записей для определенного MAC-адреса
<code>show arp vrf <VRF NAME></code>	Вывод полной ARP-таблицы в VRF
<code>show arp vrf <VRF NAME> interface <INTERFACE NAME></code>	Вывод ARP-таблицы для записей, полученных с определенного интерфейса в VRF
<code>show arp vrf <VRF NAME> ip <IP ADDRESS></code>	Вывод ARP-записи в VRF для определенного IP-адреса
<code>show arp vrf <VRF NAME> mac <MAC ADDRESS></code>	Вывод ARP-записей в VRF 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**.

10 LLDP

Link Layer Discovery Protocol (LLDP) — протокол канального уровня, который позволяет сетевым устройствам анонсировать в сеть информацию о себе и своих возможностях, а также собирать эту информацию о соседних устройствах.

Каждое устройство, на котором включён активный режим LLDP (передача и прослушивание LLDP пакетов), отправляет информацию о себе соседям независимо от того, отправляет ли сосед информацию о себе. LLDP хранит информацию о соседях, но не перенаправляет её дальше. Информация об устройстве, которая передаётся с помощью LLDP:

- Имя устройства (System Name),
- Описание устройства (System Description),
- Идентификатор шасси (Chassis ID) — MAC адрес на порте,
- Идентификатор порта (Port ID) — Имя интерфейса,
- Время хранения информации о соседе (Time-to-Live).

Для включения функционала протокола LLDP введите команду `lldp enable` в режиме конфигурации устройства.

```
ecorouter(config)#lldp enable
```

Ввод этой команды приведёт к включению режима прослушивания LLDP пакетов (доступна обработка как нетегированных LLDP пакетов, так и пакетов с VLAN тегами в заголовках Ethernet) на всех интерфейсах, однако маршрутизатор не будет передавать соседям информацию о себе.

Для включения передачи LLDP сообщений с определённого порта, воспользуйтесь командой `lldp mode active` в режиме конфигурирования порта (передача LLDP пакетов осуществляется без инкапсуляции, без дополнительных VLAN тегов). Команда доступна только после включения LDP командой `lldp enable`.

```
ecorouter(config-port)#lldp mode active
```

Для отключения активного режима и обратного перехода в режим прослушивания LLDP сообщений введите команду `no lldp mode active`

Таблица 25 — Дополнительные команды конфигурирования LLDP

Команда	Режим	Описание
<code>lldp system-name <NAME></code>	(config)#	Имя системы, по умолчанию используется имя устройства (параметр hostname). Команда не будет



Команда	Режим	Описание
		отображаться в конфигурации если hostname и введённый параметр NAME совпадают.
lldp system-description <LINE>	(config)#	Описание системы, по умолчанию используется имя операционной системы — EcoRouterOS.
lldp tx-interval <5-3600>	(config-port)#	Интервал отправки LLDP сообщений в сторону соседей в секундах. По умолчанию — 30 секунд. При изменении параметра динамически меняется и Time-to-Live (TTL) — время в течение которого наш сосед будет хранить информацию о нас. Формула для расчёта $TTL = tx_interval * 4$, TTL = 120 секундам по умолчанию.

Для просмотра информации о LLDP соседях и счётчиках на интерфейсах воспользуйтесь командами:

```
ecorouter#show lldp neighbors port ge0
Local port: ge0
Remote neighbors:
  System Name : eco test
  System Description : DGS-1210-28MP
  Port Description : D-Link DGS-1210-28MP
  TTL : 120
  System Capabilities : L3 Routing
  Mandatory TLVs:
    Chassis ID Type : Chassis MAC Address: f0b4.d254.d360
    Port ID Type : Interface Name: ge0
  802.1 Specific TLV:
    Port VLAN id      : 0
    PP VLAN id        : 0
  Protocols Advertised:
    VID Usage Digest  : 0
    Management VLAN   : 0
    LAG Status         : Disabled
    LAG Port ID        : 0
  802.3 Specific TLV:
    AutoNegotiation    :
    Capability          : 0
    Operational MAU Type: 0
    Max Frame Size      : 0

ecorouter#show counters lldp port ge0
Agent Mode: Nearest bridge
```



```
Enable Tx/Rx: No/Yes
MED Enabled: No
Device Type: Not defined
LLDP Agent traffic statistics:
Total frames transmitted: 0
Total entries aged: 0
Total frames received: 2652
Total frames received in error: 0
Total frames discarded: 0
Total discarded TLVs: 0
Total unrecognised TLVs: 0
...
```

11 Dynamic Host Configuration Protocol

Протокол DHCP — это стандартизированный клиент-серверный протокол, позволяющий серверу динамически распределять IP-адреса и сведения о конфигурации (параметры) клиентам. Параметры выдаются в виде опций формате TLV и их подопций.

Компоненты DHCP:

- DHCP-клиент — запрашивает настройки TCP/IP;
- DHCP-сервер — сервер, выдающий параметры 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. В роли ретранслятора могут выступать маршрутизаторы.

Для своей работы DHCP использует протокол UDP:

- Сообщения от клиента к серверу — порт сервера 67 UDP,
- Сообщения от сервера клиенту — порт клиента UDP 68.

По умолчанию сообщения DHCP ограничены текущей подсетью. Для своей работы DHCP использует широковещательные (broadcast) сообщения, а маршрутизаторы не пропускают широковещательный трафик за пределы широковещательного домена.

DHCP определён в RFC 1541 (заменён на RFC 2131) и RFC 2132 (опции DHCP),

Широковещательный домен (broadcast domain) — область сети, в которой все узлы могут общаться между собой с помощью широковещательных сообщений, без участия маршрутизатора. Обычно широковещательному домену соответствует физическая или логическая подсеть.



Рисунок 9 — Процесс DHCP маршрутизации

Параметры, передаваемые клиентом, содержатся в пакетах DHCP Discovery и Request:

- CHADDR — Client HW address клиента (MAC) в теле пакета (совпадает с CRS MAC в заголовке кадра при отправке от клиента).
- Option 61 — Client Identified (Client ID), уникальный идентификатор клиента в домене.
- Option 12 — Host name, имя хоста клиента.
- Option 55 — Parameter Request Field, список опций, запрашиваемых клиентом.

При ретрансляции DHCP Relay/Proху параметры CHADDR и Option 61 передаются без изменений.

Параметры, передаваемые клиенту содержатся в пакетах DHCP OFFER и ACK:

- IP адрес/маска подсети (IP-адрес в десятичном формате с точками, опция 1).
- Разница во времени в подсети клиента по сравнению с Универсальным скоординированным временем (UTC) (опция 2).
- IP-адрес (а) шлюза (ов) по умолчанию (Один или несколько IP-адресов в десятичном формате с точками, опция 3).
- Срок аренды адреса (IP Address Lease Time, опция 51).
- Имя домена (DNS Domain Name, опция 15).
- Домены поиска (DNS Search Domain, опция 119).
- Адреса DNS серверов (опция 6).
- Адреса NTP серверов времени (опция 42).
- Адреса серверов TFTP (только один IP: опции 66, несколько IP: опция 150 (cisco proprietary)).
- Названия файла загрузки с TFTP сервера (опция 67).
- Маршрут DHCP (опция 121).
- Адреса SIP сервера (опция 120, RFC 3361).

Параметры, передаваемые ретранслятором:

- Опция 82 — опция протокола **DHCP**, используемая для того чтобы проинформировать **DHCP**-сервер о том, от какого **DHCP**-ретранслятора (Agent Remote ID) и через какой его порт (Agent Circuit ID) был получен запрос. Применяется при решении задачи привязки IP-адреса к порту коммутатора и для защиты от атак с использованием протокола **DHCP** (**DHCP** snooping).

Опция 82 состоит из двух подопций:

- **Agent Circuit ID** — содержит информацию о том, с какого порта пришёл запрос на DHCP-ретранслятор.
- **Agent Remote ID** — идентификатор самого DHCP-ретранслятора (который задаётся при настройке, можно например использовать MAC-адрес коммутатора или его описание, любое удобное значение).

Если на коммутаторе включено перенаправление DHCP запросов (DHCP Relay) и включена поддержка опции 82 протокола DHCP, коммутатор перехватывает DHCP запрос от пользователя, добавляет данные Option82 (Agent Circuit ID и Agent Remote ID) к DHCP пакету и перенаправляет запрос на DHCP сервер.

11.1 Работа DHCP

11.1.1 Получение настроек

DHCP работает по схеме клиент-сервер. Для автоматической конфигурации компьютер-клиент на этапе конфигурации сетевого устройства обращается к DHCP-серверу и получает от него нужные параметры.

Процесс получения настроек происходит в несколько этапов и описывается схемой DORA (Discover-Offer-Request Acknowledge).



Рисунок 10 — Процесс получения настроек от DHCP сервера

Discover (Обнаружение)

Клиент DHCP подключается к сети и приступает к инициализации (состояние INIT). Первым делом он ищет в сети подходящий DHCP-сервер, для чего отправляет запрос **DHCPDISCOVER** на широковещательный адрес 255.255.255.255. В качестве своего адреса клиент указывает 0.0.0.0, поскольку своего адреса у него еще нет. Также в запросе клиент указывает свой MAC-адрес. Запрос доставляется всем компьютерам, находящимся в данном сегменте сети, но отвечают на него только DHCP-сервера.

Offer (Предложение)

DHCP-сервер, получивший запрос DHCPDISCOVER, анализирует его содержимое, выбирает подходящую конфигурацию сети и отправляют ее в сообщении **DHCPOFFER**. Обычно DHCPOFFER отправляется на MAC-адрес клиента, указанный в DHCPDISCOVER, хотя иногда может использоваться широковещательный ответ. Если в сети находятся несколько DHCP-серверов, то клиент получает несколько ответов DHCPOFFER и выбирает из них один, как правило полученный первым.

Request (Запрос)

Получив ответ сервера, клиент отвечает сообщением **DHCPREQUEST**, в котором "официально" запрашивает у сервера предоставленные настройки. В сообщении DHCPREQUEST содержится та же информация, что и в DHCPDISCOVER, а также IP-адрес выбранного DHCP-сервера. DHCPREQUEST отправляется на широковещательный адрес и те DHCP-сервера, чей адрес отсутствует в сообщении, понимают что их предложение отвергнуто.

Acknowledge (Подтверждение)

DHCP-сервер, адрес которого указан в DHCPREQUEST, получает сообщение и понимает, что его выбрали. Он фиксирует привязку для клиента и отвечает сообщением **DHCPACK**, подтверждая выданные клиенту настройки. DHCPACK отправляется на MAC-адрес клиента, указанный в DHCPREQUEST. Клиент получает сообщение DHCPACK, проверяет настройки и применяет конфигурацию (состояние BOUND), которая была получена в сообщении DHCPOFFER. 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.

Клиент может проверить полученный от DHCP-сервера адрес, например с помощью широковещательного ARP-запроса. Если обнаружится, что предложенный адрес уже используется в сети, то клиент отправляет серверу сообщение **DHCPDECLINE** и начинает процедуру инициализации заново. Сообщение DHCPDECLINE передается в широковещательном режиме, поскольку клиент отвергает предложенный ему IP-адрес. DHCP-сервер, получив сообщение DHCPDECLINE, должен пометить IP-адрес как недоступный, а также уведомить администратора о возможных проблемах в конфигурации.

11.1.2 Дополнительные варианты

У клиента уже есть назначенный ранее адрес.

Если у клиента имеется выданный ранее сетевой адрес и он хочет его использовать, то можно пропустить некоторые этапы. В этом случае клиент передает широковещательное сообщение DHCPREQUEST, указывая в сообщении имеющийся у него адрес. DHCP-сервер, получивший запрос, проверяет корректность сети и адреса и

в случае успешной проверки посылает клиенту подтверждение DHCPACK. Клиент получает подтверждение и применяет настройки.

Если DHCP-сервер обнаруживает, что клиент находится в неподходящей сети, он отвечает отказом DHCPNACK. Если сеть корректна, то проверяется наличие записи для этого клиента и доступность запрошенного адреса. Если адрес по какой либо причине не подходит (например занят), то сервер отвечает отказом DHCPNACK. Получив отказ, клиент больше не может пользоваться сохраненным сетевым адресом и должен запросить новый адрес, начав полную процедуру инициализации.

Если же на сервере нет записи для этого клиента, то он считает, что адрес был выдан другим DHCP-сервером и просто оставляет запрос без ответа. Такое поведение позволяет находиться в одной сети несколькими независимым DHCP-серверам.

У клиента есть адрес, полученный другим способом.

Если у клиента уже есть адрес, назначенный любым другим способом (например вручную), то он может запросить у DHCP-сервера только конкретные параметры конфигурации (например адреса DNS-серверов) с помощью сообщения **DHCPINFORM**. Сообщение передается на адрес сервера (если он известен) либо широковещанием на адрес 255.255.255.255. DHCP-сервер, получивший сообщение DHCPINFORM, отвечает сообщением DHCPACK с требуемыми параметрами конфигурации, но без проверки аренды и выделения сетевого адреса. Сообщение передается клиенту напрямую. Клиент принимает ответ и применяет полученные настройки.

Если клиент не получает от сервера сообщения DHCPACK в течение срока ожидания, то он должен выдать пользователю сообщение о проблеме и приступить к работе в сети, используя параметры, рекомендованные в RFC 1122.

Процедура получения адреса может отличаться в зависимости от настроек DHCP-сервера.

Например сервер может выдать клиенту адрес, отличающийся от запрошенного, предложить адрес из другой подсети и даже вообще отказать в предоставлении адреса. Также, DHCP-сервер вовсе не должен отвечать на каждый поступивший к нему запрос. Это позволяет контролировать доступ к сети, например можно выдавать адреса только клиентам, прошедшим определенную проверку.

11.1.3 Обновление адреса

IP-адрес выдается клиенту на определенное время, которое называется временем аренды (lease time). Время аренды зависит от настроек сервера и может варьироваться от нескольких минут до недель и даже месяцев. По прошествии половины срока клиент пробует обновить аренду. Если сразу обновить аренду не удастся, то клиент будет

пытаться сделать это снова вплоть до окончания срока. В том случае, если все попытки окажутся неудачными, по окончании срока клиент будет искать другой DHCP-сервер.

В процессе обновления клиент проходит два состояния — обновление адреса (RENEWING) и обновление конфигурации (REBINDING). Первое состояние наступает на примерно половине срока аренды адреса (T1), второе — по истечении 87.5% (или 7/8) полного срока аренды (T2). Для предотвращения синхронизации разных клиентов при расчете значений T1 и T2 к ним добавляется случайное отклонение.

Обновление адреса (RENEWING).

Это состояние означает, что клиент может начать процесс обновления аренды. Для обновления клиент посылает запрос **DHCPREQUEST**, но не широковещательный, а адресованный своему DHCP-серверу. Сервер получает запрос, после чего возможно два варианта:

- Сервер соглашается продлить аренду. Для подтверждения продления аренды он посылает клиенту сообщение **DHCPACK** с указанием нового срока аренды и тех параметров, которые могли измениться с момента создания или последнего продления аренды;
- Сервер отказывается продлевать аренду. В этом случае он шлет клиенту сообщение об отказе **DHCPNACK**.

В зависимости от полученного ответа клиент:

- В случае положительного ответа DHCPACK отмечает новый срок истечения аренды и все измененные параметры, полученные от сервера, сбрасывает таймеры T1 и T2 и переходит в нормальное (BOUND) состояние.
- Получив отрицательный ответ DHCPNACK немедленно переходит в состояние инициализации (INIT) и начинает процедуру получения аренды заново.

Обновление конфигурации (REBINDING).

Если клиент сразу не получает ответ от сервера на запрос обновления аренды, то он ожидает ответ в течение времени $(T2 - t)/2$ сек (но не меньше 60 сек), где t — время отправки последнего сообщения DHCPREQUEST, затем отправляет сообщение повторно. Пока сервер не ответит, клиент остается в состоянии RENEWING и регулярно шлет запрос DHCPREQUEST на сервер. В течение этого времени он сохраняет свой текущий адрес и продолжает нормально работать.

Если ответ от сервера не поступил к моменту T2, клиент переходит в состояние REBINDING и передает уже широковещательное сообщение DHCPREQUEST со своим текущим адресом. В этом случае срок повтора запросов DHCPREQUEST рассчитывается аналогично предыдущему случаю, только вместо T2 используется полное время окончания срока аренды.



В том случае, если срок аренды завершается до получения клиентом ответа от сервера, клиент должен прекратить все сетевые операции и перейти в состояние инициализации (INIT). Если DHCP-сервер все-таки ответит после завершения аренды, то клиент может возобновить работу с прежним адресом.

11.1.4 Освобождение адреса

Клиент может явно отказаться от аренды сетевого адреса, передав серверу сообщение **DHCPRELEASE**. При получении этого сообщения сервер помечает адрес как свободный, но сохраняет запись с параметрами клиента в базе на тот случай, если клиент захочет использовать адрес повторно. Стоит уточнить, что клиент не освобождает аренду при обычном выключении, все настройки сохраняются локально. Клиент передает DHCPRELEASE только при явной необходимости отказаться от аренды, например при перемещении в другую подсеть. Также освободить аренду можно вручную (например, с помощью команды `ipconfig /release`).

11.1.5 Режим ретрансляции

EcoRouter поддерживает 2 режима ретрансляции:

- DHCP-relay;
- DHCP-proxy.

Если опция 82 включена на EcoRouter, то в режиме DHCP-relay ее параметры добавляются в запрос (REQUEST) клиента.

В таблице ниже приведены особенности этих режимов.

Таблица 26 — Особенности режимов DHCP-relay и DHCP-relay-proxy

Действие или событие	Действие EcoRouter в режиме DHCP-relay	Действие EcoRouter в режиме DHCP-proxy
==> DHCPDISCOVER 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

Действие или событие	Действие EcoRouter в режиме DHCP-relay	Действие EcoRouter в режиме DHCP-proxy
<p><== DHCP OFFER</p> <p>DHCP-серверы отправили сообщения OFFER</p>	EcoRouter перенаправляет сообщения OFFER от всех ответивших DHCP-серверов клиенту	EcoRouter подменяет в сообщении OFFER от первого ответившего клиенту DHCP-сервера его адрес своим адресом, добавляет в таблицу информацию о выданном IP-адресе и времени начала аренды, а остальные сообщения OFFER игнорирует
<p>==> DHCPREQUEST</p> <p>Клиент послал сообщение REQUEST</p>	EcoRouter перенаправляет широковещательное сообщение REQUEST	EcoRouter подменяет адрес клиента на собственный и перенаправляет сообщение REQUEST выбранному DHCP-серверу
<p><<== DHCPACK</p> <p>DHCP-сервер послал сообщения ACK на MAC-адрес компьютера, указанного в сообщении REQUEST</p>	EcoRouter перенаправляет сообщение ACK клиенту	EcoRouter перенаправляет сообщение ACK клиенту
<p>Наступил момент для запроса обновления аренды адреса (RENEWING) (определяется настройками DHCP-сервера)</p>	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 времени получения последнего пакета подтверждения от сервера
<p>Наступил момент для запроса обновления конфигурации (REBINDING) (определяется настройками DHCP-сервера)</p>	EcoRouter перенаправляет широковещательное сообщение REQUEST с текущим сетевым адресом клиента	EcoRouter самостоятельно направляет широковещательное сообщение REQUEST с текущим собственным сетевым адресом

11.1.5.1 Настройка опции 82

Для начала настройки необходимо выбрать соответствующий dhcp-profile:

```
ecorouter>enable
ecorouter#config
ecorouter(config)# dhcp-profile DHCP_PROFILE_INDEX
ecorouter(config-dhcp)#
```

В этом режиме можно настроить опцию 82:

```
ecorouter(config-dhcp)#information-option ?
  circuit-id  Circuit ID identifies the circuit
  remote-id   Remote ID identifies the remote host
  rewrite     Re-write option (82) toward servers
```

После ввода `circuit-id` или `remote-id` можно посмотреть подсказку:

```
ecorouter(config-dhcp)#information-option circuit-id ?
  LINE  Circuit ID line
  hex    Circuit ID in HEX format
ecorouter(config-dhcp)#information-option circuit-id hex ?
  HEX_DATA  Circuit ID in HEX format
```

После `hex` можно вводить значение, которое будет интерпретировано как шестнадцатеричное число и передано без изменений.

Число должно начинаться с `0x`, иначе не будет валидировано, и будет выведено сообщение об ошибке.

Максимально доступная длина HEX числа для ввода — 512 байт (256 двухбайтных ASCII символов).

Пример ввода HEX значения:

```
ecorouter(config-dhcp)#information-option circuit-id hex 0x010203ff9900
```

Можно использовать следующие шаблоны:

- `%{vr}`
 - Virtual Router ID
 - Кодировается в HEX как Int
- `%{cmac}`
 - Client MAC Address
 - Передается в CISCO-формате, в виде одного шестнадцатеричного числа без разделителей

- `%{port}`
 - Символьное имя порта.
 - Кодировается в HEX как ASCII
- `%{hname}`
 - Hostname роутера.
 - Кодировается в HEX как ASCII
- `%{svlan}`
 - Service (Outer) VLAN ID.
 - Кодировается в HEX как Int
- `%{cvlan}`
 - Customer (Inner) VLAN ID.
 - Кодировается в HEX как Int

Пример использования шаблона:

```
ecorouter(config-dhcp)#information-option circuit-id hex 0x010203%  
{cvlan}ff9900
```

11.1.5.2 Настройка DHCP-ретранслятора

DHCP-ретранслятор (DHCP Relay Agent) — это сетевой компонент, который пересылает DHCP-запросы между клиентами и DHCP-серверами, находящимися в разных подсетях.

Для указания IP-адреса интерфейса DHCP-ретранслятора, на котором было получено DHCP-сообщение, ретранслятор отправляет в сообщениях в адрес DHCP-сервера поле `giaddr` (Gateway IP Address / Relay Agent IP).

Функции поля `giaddr` :

- Сообщает DHCP-серверу о подсети, в которой находится DHCP-клиент, запрашивающий аренду IP-адреса.
- Сообщает DHCP-серверу IP-адрес, который будет использоваться для взаимодействия с ретранслятором.

Если в сети используется `secondary` адресация на L3 интерфейсах и множество различных IP подсетей находится в одном физическом сегменте, то выбор на стороне DHCP сервера корректного IP пула заметно усложняется.

В EcoRouterOS есть возможность автоматического подбора IP адресов в опцию `giaddr` DHCP-ретранслятора из списка IP адресов, которые представлены на L3 интерфейсе за которым находятся DHCP клиенты. Это функционал называется `smart relay`.

Функция включается из режима конфигурирования профиля DHCP в режиме `проху`:

```
R1(config)#dhcp-profile 1
R1(config-dhcp)#mode проху
R1(config-dhcp)#relay-agent smart <1-65535>
```

`<1-65535>` — количество попыток отправки запросов DISCOVER в адрес DHCP-сервера. Запрос считается неудачным, если в течение одной секунды от DHCP-сервера не приходит ответа.

При отсутствии ответа после указанного числа попыток, агент ретрансляции (маршрутизатор) возьмёт следующий IP из списка `secondary` на L3 интерфейсе и отправит DHCP запросы с ним в поле `giaddr`. И так по порядку, пока DHCP сервер не ответит.

Если у одного из абонентов закончилось время аренды IP-адреса и его место в пуле адресов DHCP освободилось, благодаря тому, что DHCP `проху` хранит информацию о DHCP записи, которую выдал сервер, следующий DHCP запрос будет отправлен на сервер с тем значением поля `giaddr`, что был у этого абонента.

Если высвободились адреса нескольких абонентов — DHCP `проху` пытается отправить запросы DHCP серверу с соответствующими IP-адресами в порядке прописанном на интерфейсе.

11.2 Настройка DHCP-функционала на EcoRouter

11.2.1 Список команд

Таблица 27 — Список команда для работы с DHCP

Команда	Описание
<code>ecorouter(config)#dhcp-profile VALUE</code>	Создание DHCP-профиля, где VALUE — любое численное значение.
<code>ecorouter(config-dhcp)#description LINE</code>	Описание созданного профиля, где LINE — любое значение. Необязательная команда.
<code>ecorouter(config-dhcp)#mode проху</code>	Включение режима работы <code>проху</code> для ретрансляции запросов к серверу. Задание режима работы обязательно.
<code>ecorouter(config-dhcp)#mode relay</code>	Включение режима работы <code>relay</code> для ретрансляции запросов к серверу. Задание режима работы



Команда	Описание
	обязательно.
<code>ecorouter(config-dhcp)#server IP-address</code>	Указание IP-адреса DHCP-сервера. Обязательная команда.
<code>ecorouter(config-dhcp)#server IP-address lease VALUE</code>	Указание адреса сервера с возможным временем использования адреса от него в секундах. Значение по умолчанию 3600. Работает только для режима проху.
<code>ecorouter(config-dhcp)#information-option circuit-id LINE</code>	Опция передачи дополнительной информации серверу. Необязательная команда.
<code>ecorouter(config-dhcp)#information-option install</code>	Принудительная установка информационной опции. Необязательная команда.
<code>ecorouter(config-dhcp)#information-option remote-id LINE</code>	Опция передачи информации с mac-адресом клиента, который отправил запрос. Необязательная команда.
<code>ecorouter(config-dhcp)#information-option rewrite</code>	Перезапись информационной опции. Если circuit-id и remote-id не будут заданы на маршрутизаторе, то опция будет просто удалена из пакета. Необязательная команда.
<code>ecorouter(config-if)#dhcp-profile VALUE</code>	Команда привязки созданного профиля к интерфейсу, где VALUE номер созданного профиля.

11.2.2 Базовая настройка DHCP-ретранслятора

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 28 — Параметры опции 82

Параметр	Описание
PORT	номер порта, откуда пришёл запрос
CMAC	mac-адрес клиентского оборудования
SVLAN	номер сервисного VLAN'а
CVLAN	номер VLAN'а 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
```


11.3 Настройка DHCP-сервера

Для настройки DHCP-сервера необходимо в режиме конфигурации ввести команду `dhcp-server <NUMBER>`, где **NUMBER** — номер сервера в системе маршрутизатора. При этом изменится приглашение командной строки.

```
ecorouter(config)#dhcp-server 8
```

```
ecorouter(config-dhcp-server)#
```

Настройки могут раздаваться DHCP-сервером в двух режимах: статическом и динамическом. Для динамической конфигурации устройств в сети на DHCP-сервере используется концепция пулов, в которых содержатся настройки для множества конечных устройств. При использовании данной конфигурации, клиент получает первый свободный IP-адрес из пула. Если используется статическая запись, то клиент получит IP-настройки только в случае совпадения определенных характеристик, которые позволяют однозначно его идентифицировать. Если в настройках DHCP-сервера указать RADIUS-группу, то информация по настройке ipv4 адреса у абонента будет ожидаться с RADIUS-сервера.

11.3.1 Настройка динамического режима

Для создания пула используется команда конфигурационного режима `ip pool <NAME>`, где **NAME** — имя пула. Команда переводит CLI в контекстный режим настройки пула, где командой `range <A.B.C.D-E.F.G.H>` через дефис задаются диапазоны адресов. Единичный адрес задаётся командой `range <A.B.C.D>`, но в конфигурации он будет записан в виде диапазона. Всего можно задать до 15 диапазонов в одном пуле.

По умолчанию каждый диапазон включён в работу командой `activate`. При необходимости диапазон можно исключить из работы командой `deactivate` в контексте диапазона.

В контексте настройки пула можно задать описание пула командой `description <LINE>`, где **LINE** — текстовая строка (допустимы пробелы).

Как только устройства начнут запрашивать конфигурацию у сервера, то им будут выделены указанные IP-адреса.

Теперь следует указать как и какие именно пулы будут использоваться DHCP-сервером. У каждого пула есть собственный базовый набор свойств, это его имя, маска подсети, приоритет и DHCP-опция 82.

Правила выдачи IP-настроек для конечных устройств.



- Если клиент находится в сети, непосредственно подключенной к DHCP-серверу (в одном широковещательном домене), то поле `giaddr` в пакете DHCP Discover будет пустым. При таких условиях DHCP-сервер из множества пулов находит самый приоритетный из соответствующих IP-подсети, сконфигурированной на интерфейсе (куда пришел DHCP discover). Если на интерфейсе присутствует `secondary` IP-адрес, то проверка пула на соответствие по `secondary` адресу будет проводиться только в том случае, если основной пул уже был исчерпан. Обратите внимание, что если сконфигурирована DHCP-опция 82 на L2/L3 устройствах, то на приеме она должна совпасть с настройками опции на сервере.
- Если клиент находится в удаленной сети (в другом широковещательном сегменте), то поле `giaddr` в пакете DHCP Discover будет содержать адрес DHCP-ретранслятора. При таких условиях DHCP-сервер из множества сконфигурированных пулов находит самый приоритетный, соответствующий IP-подсети DHCP-ретранслятора (но не адреса источника DHCP-сообщения!). Обратите внимание, что если сконфигурирована DHCP-опция 82 на L2/L3 устройствах, то на приеме она должна совпасть с настройками опции на сервере.
- Статические правила имеют приоритет над динамическими (пулами). Поэтому при совпадении параметров, которые позволяют однозначно идентифицировать клиента (MAC-адреса источника (в заголовке BOOTP а не Ethernet), опции Client ID, или опции 82 в пакете DHCP discover), IP-настройки будут выданы без проверки подсетей на интерфейсах. Обратите внимание, что если сконфигурирована DHCP-опция 82 на L2/L3 устройствах, то на приеме она должна совпасть с настройками опции на сервере.

Исходя из этих правил, в конфигурацию DHCP-сервера вводятся следующие параметры пулов.

- **Имя** — это имя ранее созданных IP-пулов, их может быть несколько в конфигурации DHCP-сервера.
- **Маска** — этот параметр совместно с IP-адресами из пула, будет указывать нам — из какого пула следует выдать настройки для конечного устройства и какую маску подсети им передать в этих настройках в качестве DHCP-опции.
- **Приоритет** — Это свойство определяет порядок обработки всех сконфигурированных пулов в сервере при приёме DHCP discover пакета от конечных устройств. Приоритет пула как при работе с ACL или route-map



задаётся с помощью определённого номера последовательности (пула). Чем ниже номер тем выше приоритет. Напомним, что у статических правил приоритет всегда выше чем у пулов.

Все эти свойства являются ключевыми параметрами для выбора правильного пула для выдачи динамических настроек.

Для создания пула используется команда в режиме конфигурации DHCP сервера:

`pool <NAME><Priority SEQ>`, где **NAME** — имя пула, **Priority SEQ** — номер пула, определяющий его приоритет. Чем ниже номер тем выше приоритет.

При вводе вышеуказанной команды произойдет переход в режим конфигурации пула.

```
ecorouter(config-dhcp-server-pool)#
```

Свойства и опции для динамической настройки клиентов с помощью пулов конфигурируются в этом режиме. Доступные для настройки параметры приведены в таблице ниже.

Таблица 29 — Команды режима конфигурации пула

Параметр	Описание
<code>mask <X.X.X.X></code>	Маска подсети в 4-х октетном формате. Можно ввести длину маски в сокращенном десятичном формате. Например, 16 для маски 255.255.0.0
<code>lease <TIME></code>	Время аренды адреса в секундах
<code>information-option <circuit-id><remote-id> <STRING></code>	Опция 82 в формате строки. Где circuit-id ассоциируется с клиентом, а remote-id с L2/L3 сетевыми устройствами на пути до DHCP сервера.
<code>gateway <X.X.X.X></code>	Шлюз по умолчанию

Для удаления или изменения настроек можно воспользоваться стандартными вариантами команды **no**.

Пример:

От DHCP relay приходит пакет DHCP Discover на L3 интерфейс EcoRouter с IP-адресом 10.0.0.1/24, который в свою очередь ретранслировал этот DHCP Discover из сети 172.16.0.0/16 с L3 интерфейса 172.16.0.1/16. При приеме DHCP-сервер обнаружит в поле giaddr адрес 172.16.0.1 — на него сервер и будет ориентироваться при поиске нужного пула для выдачи всех IP-настроек. Допустим на сервере в этот момент присутствует три пула с разными именами "A", "B" и "C", где:

- пул A с номером 10 и адресами из сети 192.168.0.0 с маской 16, без опции 82,

- пул В с номером 20 и адресами из сети 172.0.0.0 с маской 8, без опции 82,
- пул С с номером 30 и адресами из сети 172.16.0.0 с маской 16, без опции 82.

Два пула “В” и “С” соответствуют адресу 172.16.0.1, но т. к. приоритет у пула В больше (статических правил нет а у пула “С” номер 30) и опции 82 не сконфигурировано на сети, то будет использоваться пул В.

11.3.2 Настройка статического режима

Как уже упоминалось ранее — выдача IP-адреса из пула осуществляется в произвольном порядке, какой IP-адрес в пуле свободен, тот и будет передан клиенту. При настройке DHCP-сервера есть возможность создать статическую привязку IP-адреса и других опций к конечному устройству, это позволит выдавать клиенту желаемые и запланированные вами настройки на постоянной основе.

Для настройки статической привязки следует ввести команду контекстного режима **static ip <A.B.C.D>**, где <A.B.C.D> — IP-адрес, выдаваемый клиенту.

После ввода этой команды произойдет переход в режим конфигурации статической записи:

```
(config-dhcp-server-static)#
```

Для того, чтобы определенному пользователю выдавались нужные настройки, в пакете DHCP discover необходимо выбрать поля для однозначной идентификации клиента. Сделать это можно по:

- опции 82,
- полям Source MAC (не в заголовке Ethernet) в заголовке DHCP
- или опции Client-ID.

Обратите внимание, что можно использовать Client-ID и Source MAC одновременно. Таким образом к базовому набору свойств для статической записи, помимо IP-адреса добавляются: опция 82, опция Client ID и Source MAC.

Все остальные опции настраиваются аналогично пулам. Доступные для настройки параметры приведены в таблице ниже.

Таблица 30 — Команды настройки статического режима

Параметр	Описание
<code>chaddr <XXXX.XXXX.XXXX></code>	Настройка поля CHADDR. XXXX.XXXX.XXXX — MAC-адрес клиента в HEX формате.
<code>client-id mac <XXXX.XXXX.XXXX></code>	Настройка поля Source MAC. XXXX.XXXX.XXXX — MAC-адрес клиента в HEX формате.

Параметр	Описание
<code>mask <X.X.X.X></code>	Маска подсети в 4-х октетном формате. Можно ввести длину маски в сокращённом десятичном формате. Например, 16 для маски 255.255.0.0
<code>lease <TIME></code>	Время аренды адреса в секундах
<code>information-option <circuit-id><remote-id> <STRING></code>	Опция 82 в формате строки. Где circuit-id ассоциируется с клиентом, а remote-id с L2/L3 сетевыми устройствами на пути до DHCP сервера
<code>gateway <X.X.X.X></code>	Шлюз по умолчанию

Для удаления или изменения настроек можно воспользоваться стандартными вариантами команды **no**.

11.3.3 Настройка RADIUS-группы

У пользователей есть возможность получить IP настройки от удалённого RADIUS сервера. Функционал более известен под названием DHCP-RADIUS-Proxy.

При получении DHCP Discovery пакета от абонента (пользователя), будет сформирован RADIUS request с информацией по сессии. Ожидается получить RADIUS Reply с атрибутами для ipv4 настроек абонента. **Framed-IP-Address** — где указан конкретный адрес или **Framed-Pool** — с именем пула, из которого необходимо выделить адрес. Если получен Access-Accept, то будет продолжен процесс DORA с полученными параметрами. В случае с Access-Reject — процесс DORA будет остановлен.

При использовании атрибута **Framed-Pool** в маршрутизаторе должен быть сконфигурирован IP пул с идентичным именем.

При использовании на RADIUS сервере атрибута **Framed-Pool** совместно с **Framed-IP-Address** для работы функционала DHCP-RADIUS-Proxy будет использован атрибут **Framed-IP-Address**.

Помимо передачи IP адреса абонента с RADIUS сервера, есть возможность передачи информации о маске подсети и дефолтного шлюза, для этого используйте стандартные атрибуты RADIUS сервера **Framed-IP-Netmask** и **Framed-Route**.

Например:

```
Framed-Route = 0.0.0.0 10.0.0.1 1
Framed-IP-Netmask = 255.255.0.0
```

Обратите внимание, что для передачи информации о шлюзе по умолчанию, наличие записи 0.0.0.0 в строке атрибута **Framed-Route** является обязательным!

Для применения RADIUS-группы следует ввести команду контекстного режима `external radius <NAME>`, где **NAME** — имя сконфигурированной RADIUS-

группы.

Все остальные опции для передачи абонентам (DNS, TFTP, NTP ...) настраиваются аналогично пулам.

Приведём пример настройки DHCP сервера и RADIUS группы на BRAS интерфейсе маршрутизатора при использовании атрибута **Framed-IP-Address**

```
!  
radius-group test  
mode active-standby  
radius-server 20.0.0.2 secret pass1234 priority 10  
!  
subscriber-aaa test  
authentication radius test  
!  
dhcp-server 1  
external radius test  
ntp 8.8.8.8  
dns 8.8.8.8  
!  
interface bmi.1  
connect port te0 service-instance test  
dhcp-server 1  
subscriber-map test  
session-trigger ip  
ip address 10.0.0.1/16  
!  
interface radius  
connect port te1 service-instance test  
ip address 20.0.0.1/16  
!
```

11.3.4 Глобальная настройка

Часто можно встретить ситуацию, когда конфигурация опций в пулах одинаковая или пользователь забыл указать какую-либо специфическую для пула опцию, для таких случаев предусмотрена возможность сконфигурировать глобальные опции для всего DHCP-сервера. Сделать это можно не в режимах пула или статической записи, а в режиме конфигурации самого сервера с помощью тех же команд.

Для настройки DHCP-сервера необходимо в режиме конфигурации ввести команду `dhcp-server <NUMBER>`, где **NUMBER** — номер сервера в системе маршрутизатора. При этом изменится приглашение командной строки.

```
ecorouter(config)#dhcp-server 8
```




```
ecorouter(config-dhcp-server)#
```

Доступные для настройки параметры приведены в таблице ниже.

Таблица 31 — Команды режима настройки DHCP-сервера

Параметр	Описание
<code>lease <TIME></code>	Время аренды адреса в секундах
<code>gateway <X.X.X.X></code>	Шлюз по умолчанию

Это не относится к базовым свойствам пулов или статических записей! Имена пулов, маски, приоритеты, опции 82 — уникальны в рамках своих пулов. IP-адрес, `client MAC`, опции 82 и `client ID` — уникальны в рамках своих статических записей.

11.3.5 Привязка к интерфейсу

После настройки сервера необходимо указать, на каком интерфейсе маршрутизатор будет принимать пакеты DHCP Discover и отвечать на них предложением с IP-настройками. Привязка происходит стандартным способом — в режиме конфигурации соответствующего интерфейса указывается нужный сервер. При помощи команды `dhcp-server <NUMBER>`, где **NUMBER** номер заранее сконфигурированного сервера.

11.3.6 Пример конфигурации

Теперь соберем все в единую конфигурацию для нашего вышеуказанного примера, добавим статическую запись и глобальный параметр `lease`.

```
ip pool A
  range 192.168.0.10
  range 192.168.0.2-192.168.0.8
  range 192.168.0.12-192.168.0.255
ip pool B
  range 172.16.0.0-172.16.255.255
ip pool C
  range 172.16.0.1-172.16.0.255
  range 172.16.1.100-172.16.1.200
dhcp-server 100
  lease 300
  static ip 192.168.0.200
  chaddr 0123.4567.89ab
  lease 3600
  gateway 192.168.0.1
  pool A 10
```



```
chaddr cdef.0123.4567
gateway 192.168.0.1
pool B 20
mask 8
gateway 172.16.0.1
pool C 30
mask 16
gateway 172.16.0.1
interface test
dhcp-server 100
```

11.3.7 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-проху, 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 test
IP Address MAC Address DHCP Server ACK Time Lease Time
-----
192.168.1.3 c403.130f.0000 20.0.0.1 296 86400
```

Команда `show dhcp-server clients <NAME>`, где **NAME** — имя интерфейса, к которому привязан DHCP-профиль. Данной командой выводится на экран таблица, содержащая список всех DHCP-клиентов для данного сервера. В таблице содержатся записи с выданными IP-адресами, мас-адресами клиентов, время подтверждения, время, на которое адрес был выдан.

```
ecorouter#show dhcp-server clients bmi.1
Total DHCP clients count: 16
IP Address MAC Address ACK Time Lease Time
-----
10.210.10.31 0c87.2c42.9d59 27 300
10.210.10.62 0017.c8af.6216 15 300
10.210.10.41 00ec.ef08.1b30 180 300
10.210.10.46 00e8.2cf5.5450 169 300
10.210.10.79 205a.3a48.971f 17 300
10.210.10.15 02ce.7be1.c72e 73 300
10.210.10.32 f110.002f.1237 235 300
10.210.10.7 011d.5cb3.5b2b 180 300
10.210.10.99 008c.fd68.2001 172 300
10.210.10.47 0318.d6a1.7eb1 140 300
10.210.10.10 0400.23e1.c666 117 300
10.210.10.113 20e2.bace.f5eb 176 300
10.210.10.12 28d5.4779.0f3e 180 300
10.210.10.81 2c56.dc76.6c9b 271 300
10.210.10.115 2243.26ab.e15a 44 300
10.210.10.118 2c59.e5d7.c280 172 300
```

12 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.

12.1 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`.



12.2 Дополнительные функции

В реализации EcoRouterOS VRRP также поддерживает ряд функций, описанных ниже.

12.2.1 Функция `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`.

12.2.2 Функция `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.

12.2.3 Функция `circuit-failover`

Для отслеживания состояния какого-либо сетевого соединения маршрутизатора, при выходе из строя которого потребуются 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.



12.2.4 Функция 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.

12.2.5 Функция 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 сентисекундах (1 cs = 0.01 s).

12.2.6 Функция 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}`.

12.3 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.

12.4 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, подключённых 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.

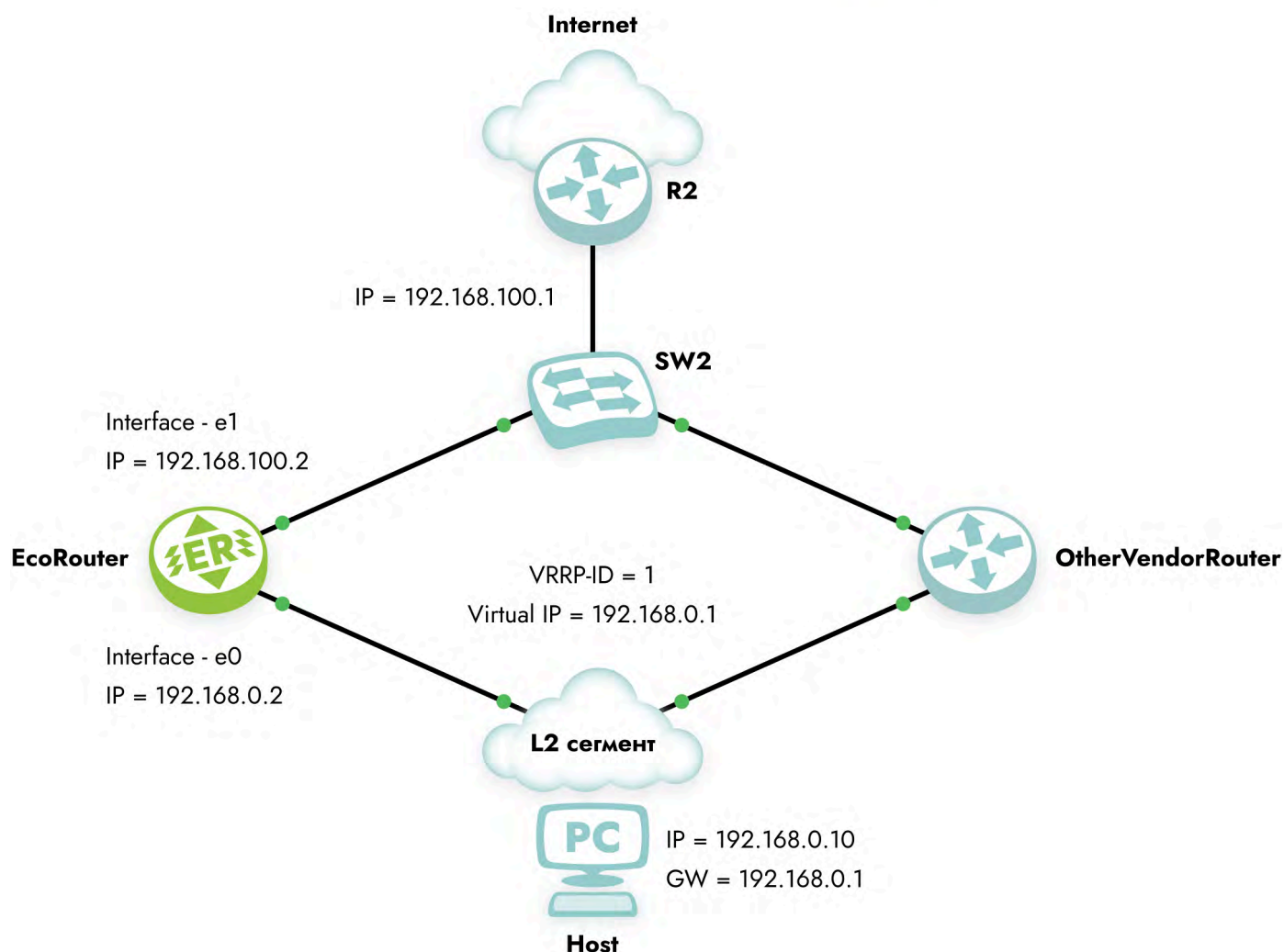


Рисунок 11

В приведённой 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 начнёт вещание VRRP-сообщений, вынудив соседа сменить роль и перестать отвечать на ARP-запросы, отправляемые владельцу IP-адреса 192.168.0.1.

12.5 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.

13 IP SLA

Функционал IP SLA (Соглашение об уровне обслуживания IP) позволяет анализировать параметры качества обслуживания различных IP сервисов и приложений, помогает обнаруживать неисправности и предотвращать аварийные ситуации в сети. Механизм его работы основан на технологии активного мониторинга трафика. Идеология "активного мониторинга" или "активного съёмника трафика" подразумевает, что на первом шаге будет происходить отправка специальных тестовых сообщений на объект исследования (investigation object), а уже на втором шаге будет произведён какой-либо анализ. Один из самых простых примеров подобного мониторинга является проверка доступности ресурса с помощью утилиты ping (отправки ICMP-запроса и ожидания ICMP-ответа).

Для настройки IP SLA в режиме конфигурации маршрутизатора введите команду:

```
ip sla-profile <NAME>, где NAME имя профиля.
```

IP SLA профиль создаёт необходимое окружение для тестов, активного мониторинга и существует с момента его создания и до его удаления (lifetime: forever) с помощью команды `no ip sla-profile <NAME>`.

Минимально необходимый набор параметров для запуска мониторинга включает в себя настройку объекта исследования, характеристик мониторинга и глобальное включение профиля. Объект и тип исследования задаётся в режиме конфигурации профиля командой:

```
icmp A.B.C.D [source E.F.G.H] [size <42-1500>] [vrf NAME] [df-bit] [dscp <0-63>]  
[num-packets <1-100>], где A.B.C.D это IP адрес объекта исследования, все остальные  
аргументы опциональные. E.F.G.H это IP адрес маршрутизатора с которого будут  
отправлены ICMP запросы, size это размер данных в теле ICMP (payload), NAME — это  
имя vrf через который будут отправляться ICMP запросы, df-bit это флаг для установки DF  
(Don't Fragment) бита в IP заголовке, dscp позволяет установить отличное от нуля  
значение в поле Type of Service в IP заголовке для приоритизации трафика, а num-  
packets задаёт число посылаемых пакетов за одну транзакцию, при этом логика работы  
команды monitor, для одного или нескольких пакетов, различна. По умолчанию всё  
работает для одного пакета. Пример для нескольких пакетов будет рассмотрен ниже.
```

Характеристиками для мониторинга, как вместе, так и по отдельности, могут служить:

- количество потерянных пакетов (Packet Loss).
- время приёма/передачи от начальной точки к месту назначения, и обратно к изначальной точке (Round Trip Timer).

Их настройка осуществляется командой:

`monitor [packet-loss <1-255>] [rtt-max <100-10000>]`, где **packet-loss** соответствует количеству потерянных пакетов (по умолчанию равен 1) после которого профиль перейдёт в состояние DOWN, параметр **rtt-max** соответствует максимально возможному RTT и задаётся в миллисекундах (по умолчанию равен 1000 мс).

В профиле есть возможность указать параметр `packet-frequency <1-300>`, который задаёт интервал отправки тестовых сообщений в секундах (по умолчанию равен 5 секундам)

Параметр **rtt-threshold** присутствует в SLA профиле по умолчанию и равен 1000 мс. В системе должен присутствовать параметр, который задаёт максимальное время ожидания выполнения транзакций (например ожидание ICMP reply сообщения в ответ на ICMP request). Значение этого параметра может быть изменено. Если параметр **rtt-max** в команде **monitor** необходим для принятия решения о успешности или неуспешности транзакции по уровню задержки, то параметр **rtt-threshold** служит в качестве индикатора потерянного (loss) пакета.

После задания вышеуказанных настроек и включения мониторинга IP SLA профиль сможет отслеживать два статуса:

- Статус последней транзакции (Latest transaction status) :
 - Success/Failure
- Операционный статус профиля (Operational profile status):
 - SHUTDOWN/DOWN/UP

Статус последней транзакции указывает на успех или неудачу последней выполненной операции. Например при отсутствии ответа на последний отправленный маршрутизатором ICMP запрос статус будет Failure. Операционный статус показывает глобальное состояние всего IP SLA профиля и он зависит от количества успешных или неуспешных выполненных транзакций/операций. Профиль можно административно включить с помощью команд **enable** или **no shutdown** (по умолчанию профиль выключен).

Изменение операционного статуса профиля можно отследить в журнале событий или посредством **SNMP trap** сообщения от агента-маршрутизатора на SNMP менеджере.

Одним из основных преимуществ IP SLA функционала, является возможность операционного статуса профиля влиять на политики маршрутизации трафика. У других вендоров сетевого оборудования этот функционал называют трекированием (tracking object). В качестве иллюстрации распространённого случая с динамическим переключением трафика на резервный канал рассмотрим следующую топологию:

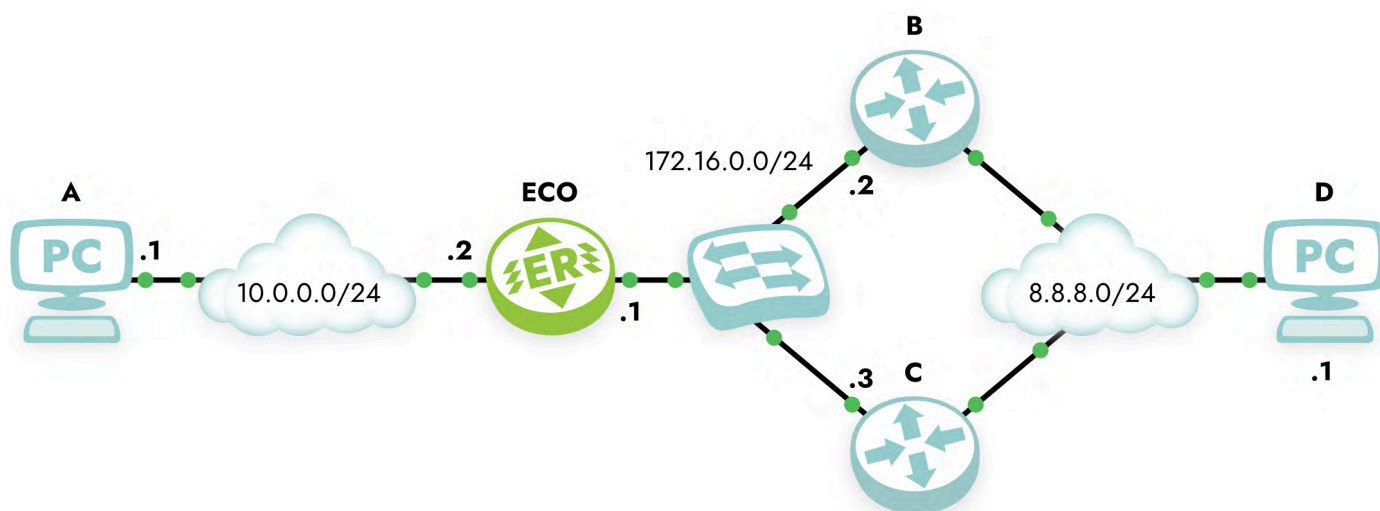


Рисунок 12

Здесь в качестве основного канала приёма/передачи данных между хостами **A** и **D** используется канал, заданный с помощью статического маршрута через маршрутизатор **B**. В случае обрыва физического линка между коммутатором и маршрутизатором **B** запросы ICMP через основной канал от маршрутизатора **ECO** в сторону конечного хоста **D** будут неуспешными. Запущенный IP SLA мониторинг способен отследить такую ситуацию и исключить статический маршрут через основной канал из таблицы маршрутизации **ECO**. Динамически переключившись на резервный статический маршрут через маршрутизатор **C**, хост **D** снова будет виден для хоста **A**.

Для того чтобы привязать IP SLA профиль к статическому маршруту в качестве последнего аргумента классической команды `ip route` укажите `sla` и имя необходимого профиля. Пример:

`ip route 1.1.1.1/32 10.0.0.1 sla TEST`, где **TEST** имя заранее сконфигурированного IP SLA профиля

Приведём пример конфигурации IP SLA на маршрутизаторе ECO для вышеуказанного примера:

```
!
ip sla-profile TEST
  icmp 172.16.0.2 sdf-bit
  monitor packet-loss 3 rtt-max 350
  packet-frequency 10

  enable
!
ip route 8.8.8.0/24 172.16.0.2 sla TEST
ip route 8.8.8.0/24 172.16.0.3 100
!
```

В качестве объекта исследования в примере выступает узел с IP адресом 8.8.8.1. Тестовая среда предполагает отправку ICMP сообщения раз в 10 секунд с выставленным DF (Don't Fragment) битом. Операционный статус IP SLA профиля перейдёт в состояние DOWN в случае отсутствия факта получения трёх последовательных ICMP ответов или 3 пакетов с RTT > 350мс. Задан статический маршрут через основной канал с привязкой к IP SLA профилю TEST. Указан резервный статический маршрут с административной дистанцией 100.

Важно заметить, что на политики маршрутизации операционный статус профиля в SHUTDOWN не влияет. Маршруты с привязанными IP SLA профилями в статусе SHUTDOWN, без исключений, добавляются в таблицу маршрутизации.

У администраторов есть возможность внести задержку на переключение состояний IP SLA профиля. Как и на любой другой процесс в операционной системе маршрутизатора на частые переключения в IP SLA расходуются ресурсы центрального процессора. Чтобы нивелировать влияние множественных переключений и сгладить реакцию системы на периодическую недоступность объектов мониторинга настройте задержку для переходов состояния UP→DOWN и DOWN→UP в секундах командой:

```
delay down <1-300> up <1-300> .
```

Таким образом, если за время задержки IP SLA профиль поменял своё операционное состояние на первоначальное, переключения не произойдёт.

Для просмотра краткой информации по всем сконфигурированным профилям воспользуйтесь командой: `show ip sla-profile summary` .

```
ecorouter#sh ip sla-profile summary
```

Profile Name	Operation Type	Destination Object	Operational Status
test1	ICMP	10.1.0.10	UP
test2	ICMP	172.16.2.20	UP
test3	ICMP	192.168.3.30	UP

А для детального вывода информации по определенному IP SLA профилю воспользуйтесь командой `show ip sla-profile <NAME>` , где **NAME** это имя профиля.

```
ecorouter#sh ip sla-profile test1
```

Type: ICMP
Operational profile status: UP
IP VRF: default
Object of investigation: 10.1.0.10
Monitor characteristics:
IP packet loss: 3
Maximum RTT: 500 msec
Latest transaction status: Success




```
Latest transaction RTT: 33 ms
Failure transactions count: 0
Success transactions count: 783
Success transactions percentage: 100.00%
Profile lifetime: forever
```

Операционный статус SLA профиля может влиять и на другие настройки в системе, помимо наличия или отсутствия маршрута в таблице маршрутизации, он может изменить значение приоритета в протоколе VRRP, а также изменить статус интерфейса и перевести его в состояние UP/DOWN. Приведём пример соответствующих настроек.

Для интерфейса:

```
router(config)#interface test
router(config-if)#monitor sla IP_SLA_PROFILE_NAME
```

Изменение операционного статуса профиля переводит статус интерфейса в соответствующее состояние.

Для VRRP:

```
ecorouter(config)#router vrrp 1 iface
ecorouter(config-router)#monitor sla IP_SLA_PROFILE_NAME decrement <1-255>
```

Перевод операционного статуса профиля в DOWN уменьшает приоритет VRRP маршрутизатора на значение, указанное после ключевого слова decrement.

Несколько SLA профилей можно объединять в логические группы, тем самым создавая более сложную логику для перестроения маршрутов. В ситуации когда помимо работоспособности основного канала, требуется отслеживать характеристики резервного канала, и переключаться на резервный маршрут только при отсутствии какой-либо деградации сервисов на нем, объединение SLA профилей в логические группы может стать удобным инструментом.

Создание логических групп:

```
router(config)#ip sla-profile A
router(config-sla)#sla ?
    match-all Boolean and
    match-any Boolean or
```

Ключевым словам **match-all** и **match-any** соответствуют логические операции «И» и «ИЛИ». Нижеследующая конфигурация будет подразумевать, что SLA профиль **C** будет в операционном статусе UP, только если и SLA профиль A и SLA профиль B в операционном статусе UP:

```
ip sla-profile A
  icmp 10.0.0.1
  monitor packet-loss 3
  packet-frequency 5
  enable
```



```
!  
ip sla-profile B  
  icmp 10.0.0.2  
  monitor packet-loss 3  
  packet-frequency 5  
  enable  
!  
ip sla-profile C  
  sla match-all A B  
  enable
```

Для использования логики «НЕ» в конфигурации профиля добавьте ключевое слово `inverse` при включении SLA профиля.

```
ip sla-profile B  
  icmp 10.0.0.2  
  monitor packet-loss 3  
  packet-frequency 5  
  enable inverse
```

Приведем пример создание SLA профилей с объединением логики «ИЛИ — НЕ»

```
ip sla-profile A  
  icmp 10.0.0.1  
  monitor packet-loss 3  
  packet-frequency 5  
  enable  
!  
ip sla-profile B  
  icmp 10.0.0.2  
  monitor packet-loss 3  
  packet-frequency 5  
  enable  
!  
ip sla-profile C  
  icmp 10.0.0.3  
  monitor _rtt-max 500_  
  packet-frequency 5  
  enable inverse  
!  
ip sla-profile D  
  sla match-all A B  
  enable  
!  
ip sla-profile E
```



```
sla match-any D C
enable
```

Подобные SLA профили можно абсолютно также связывать с маршрутами, VRRP группами, L3 интерфейсами.

С помощью Булевой логики в SLA профилях на EcoRouter можно реализовать следующий сценарии в сети:

«Не переключаться на резервный маршрут, если и на резервном направлении фиксируется ухудшение качества канала».

Во всех вышеуказанных примерах при организации мониторинга использовался одиночный ICMP пакет. В EcoRouterOS есть возможность отправить целую совокупность пакетов за одну транзакцию и на основании потерь и задержек пакетов в этой совокупности сделать вывод о качестве каналов. Как показывает практика, этот вариант мониторинга более наглядно и достоверно определяет проблемы в каналах. Для его настройки воспользуйтесь параметром **num-packets** в команде `icmp`. При таком варианте работы есть особенности в определении операционного статуса SLA профиля, обсудим их ниже:

Параметр **rtt-threshold** задаёт пороговое значение таймера, который определяет время ожидания пакета, по истечению которого одиночный пакет или пакет из совокупности считается утерянным (loss). Поэтому **rtt-threshold** должен быть больше **rtt-max**, который задаёт порог для средне арифметического значения задержки для всех пакетов в совокупности (pool), превысив который, транзакция считается неуспешной Failure и операционный статус SLA профиля моментально переходит в состояние DOWN.

Решение об операционном статусе SLA профиля принимается на основании одной транзакции. Одна транзакцией это одна отправка целой пачки ICMP request пакетов.

Пример вывода статуса транзакции из команды группы `show`:

```
Maximum RTT: 350 msec (for any pool of transactions)
Latest pool of transactions status: Failure
Latest pool of transactions RTT: 570 ms
```

Статусы транзакций и профиля зависят от количества потерянных пакетов (**packet-loss**), которые теперь подсчитываются не по статусу последней транзакции, а по утерянным пакетам внутри одной совокупности (pool), таким образом получается, что **packet-loss** не должен превышать количество пакетов в совокупности (**num-packets**).

Параметр **packet-frequency** теперь задаёт интервал отправки пачки тестовых сообщений (по умолчанию равен также 5 секундам).

14 Агрегирование каналов

Агрегирование каналов — объединение нескольких каналов в один логический канал для увеличения пропускной способности и резервирования. Чтобы добавить порты в объединённый 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 16 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 порте (см. раздел Сервисные интерфейсы).

14.1 Вычисление хэш-функции

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

Таблица 32 — Поля, используемые для вычисления хэш-функции по умолчанию

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

Где:

- 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 — изменяемое значение, уникальное в пределах маршрутизатора. Может принимать значения от 0 до 255.
- Protocol Type — протокол транспортного уровня.



- Port.no — номер порта, принявшего 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).

14.2 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 включён PDU сообщения. LACP может работать в пассивном и активном режимах. Устройство, на котором LACP настроен в пассивном режиме, не отсылает PDU (Protocol Data Unit) самостоятельно при настроенном агрегированном канале, а ждёт получения PDU от соседнего устройства и только в случае получения отсылает свои. В активном режиме LACP постоянно отправляет 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 прослушивания переводится в состояние передачи данных в случае одновременного выполнения следующих условий:

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

14.2.1 Настройка параметров

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

Таблица 33 — Команды режима конфигурирования агрегированного порта

Команда	Описание
---------	----------

Команда	Описание
<code>lACP enable</code>	Включает функционал LACP на агрегированном порте. По умолчанию функционал выключен
<code>lACP key <NUM></code>	Значение по умолчанию равно порядковому номеру порта в агрегированном канале. Изменяется в пределах от 0 до 65535
<code>lACP mode (active passive)</code>	Режим работы LACP
<code>lACP period (fast slow)</code>	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.
<code>lACP system-id <ID></code>	Идентификатор системы в формате XXXX:XXXX:XXXX
<code>lACP system-priority <NUM></code>	Задаёт 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** задаёт 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.

14.2.2 Команды просмотра

Для просмотра статистики по LACP и состояния агрегированных портов используются описываемые ниже команды группы `show`.

Для просмотра состояния и текущей конфигурации портов можно воспользоваться командой `show port brief`.

```
ecorouter#show port brief
Name           Physical  Admin   LACP
-----
ge1             DOWN     UP      *
ge2             DOWN     UP      *
te1/1          UP       UP      *
ae.0            *        UP      UP
  te0/0         UP       UP      UP
  te0/1         UP       UP      UP
```

В выводе выше включён LACP и порт ae.0 работает, все порты в группе также работают и обозначаются как **"UP"**. Когда LACP выключен, но порт ae.0 работает, в столбце "Lacp" отображаются символы звёздочка **"*"**. В ситуации, когда происходит сбой LACP, все порты в группе и ae.0 обозначаются как **"DOWN"**.

Для просмотра счётчиков 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
te1            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
te1/0     SA   bnd1  32767  0x10  8    0x3D
te1/1     SA   bnd1  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
te1/0     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
te1/1     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.

Примеры краткого и детализированного вывода команды:

```

ecorouter#show 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
te1/0     FA  32768  908d.7845.9bc0  1      28      0x3F
te1/1     FA  32768  908d.7845.9bc0  9      27      0x3F

ecorouter#show 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
te1/0     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
te1/1     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, как и для любых других команд `show`.

14.3 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 включён по умолчанию.

14.4 Настройка Link aggregation

14.4.1 Именованние 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.

14.4.2 Команды настройки агрегированного порта

Таблица 34 — Команды настройки агрегированного порта

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

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

14.4.3 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 порте:

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

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

14.4.4 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 порте (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.

14.4.5 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 определённого 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
...
```

15 Настройки зеркалирования SPAN/RSPAN

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.

15.1 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 35 — Параметры объекты конфигурации типа `mirror-session`

Параметр	Описание
<code>mirror-session</code> <code><название></code>	Название правила, по которому осуществляется зеркалирование трафика. Название может быть задано только цифрами
<code>description</code>	Описание правила. Необязательный параметр
<code>destination port</code> <code><название></code>	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)
<code>source <тип></code> <code><название></code> <code><параметры></code>	<p>Источник, трафик которого дублируется. В качестве источника может быть указан:</p> <ul style="list-style-type: none">- port,- interface,- service-instance. <p>У правила может быть несколько источников трафика, в этом случае они указываются с новой строки. Для удаления одного из источников в конфигурации <code>mirror-session</code> используется команда <code>no source <тип> <название></code>.</p>

Параметр	Описание
	Возможность настройки правил зеркалирования одновременно с конфигурированием сервисного интерфейса EcoRouter описана ниже.
Параметры source	
<направление>	<p>Определяет, какой именно трафик необходимо дублировать:</p> <ul style="list-style-type: none"> - tx — исходящий, - rx — входящий, - both — оба направления. <p>Для service-instance возможно зеркалирование только входящего трафика (rx)</p>
<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
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.

15.2 Пример настройки зеркалирования

Рассмотрим пример настройки зеркалирования для маршрутизатора и двух клиентских устройств, сконфигурированных, как представлено на схеме ниже.



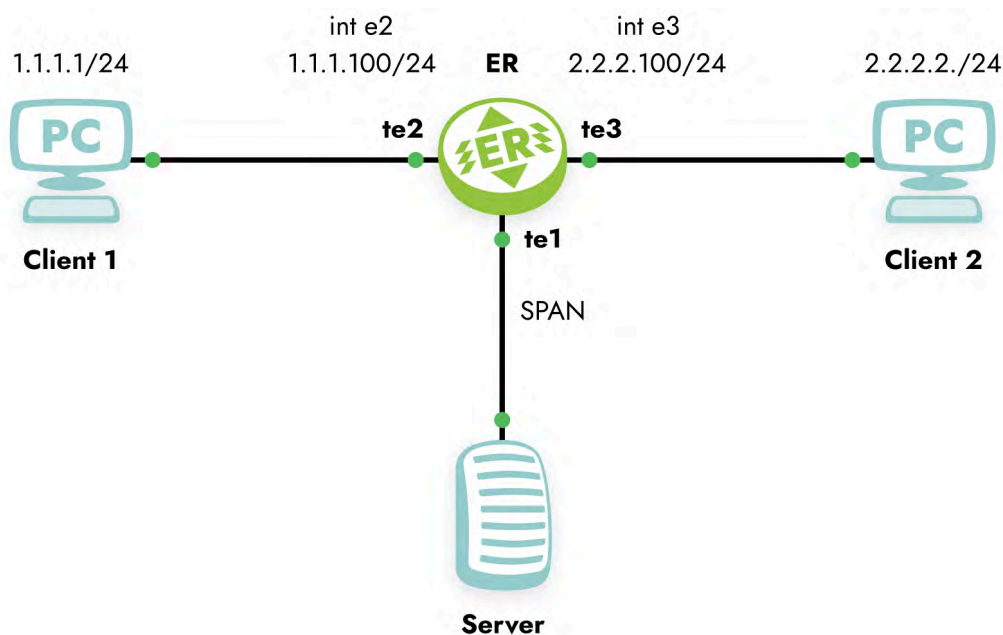


Рисунок 13

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 te1  
!  
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".

15.2.1 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 te1
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 te1
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 счётчиков для **port te2** и **port te1**. Схема зеркалирования, реализуемая правилом **mirror-session 0** представлена ниже.

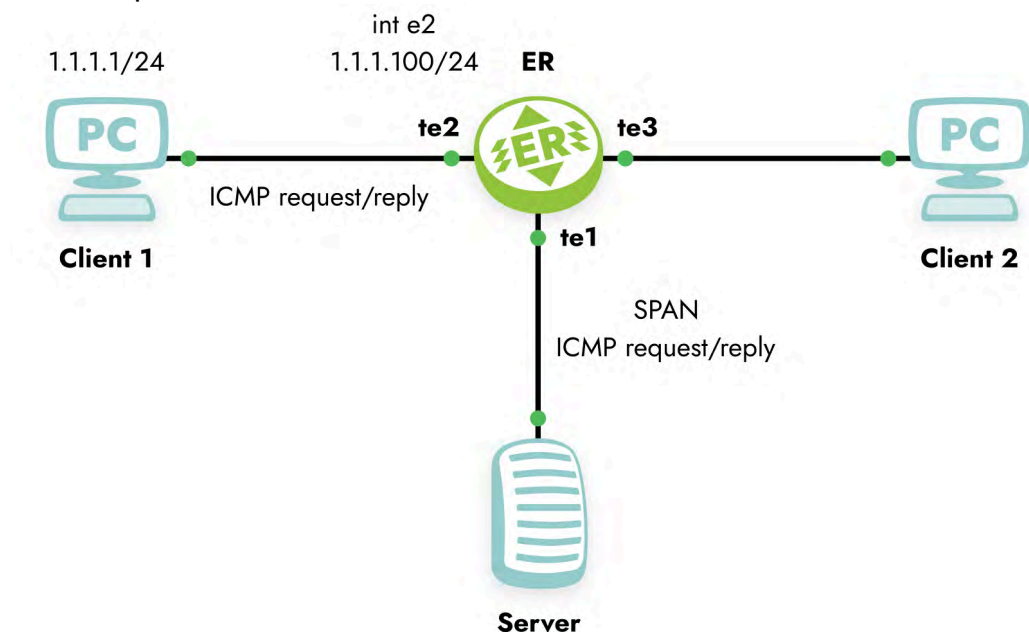


Рисунок 14

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 te1
  Total transmitted packets: 20
```

15.2.2 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 te1
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 te1
  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 счётчиков для **port te3** и **port te1**. Схема зеркалирования, реализуемая правилом **mirror-session 1** представлена ниже.

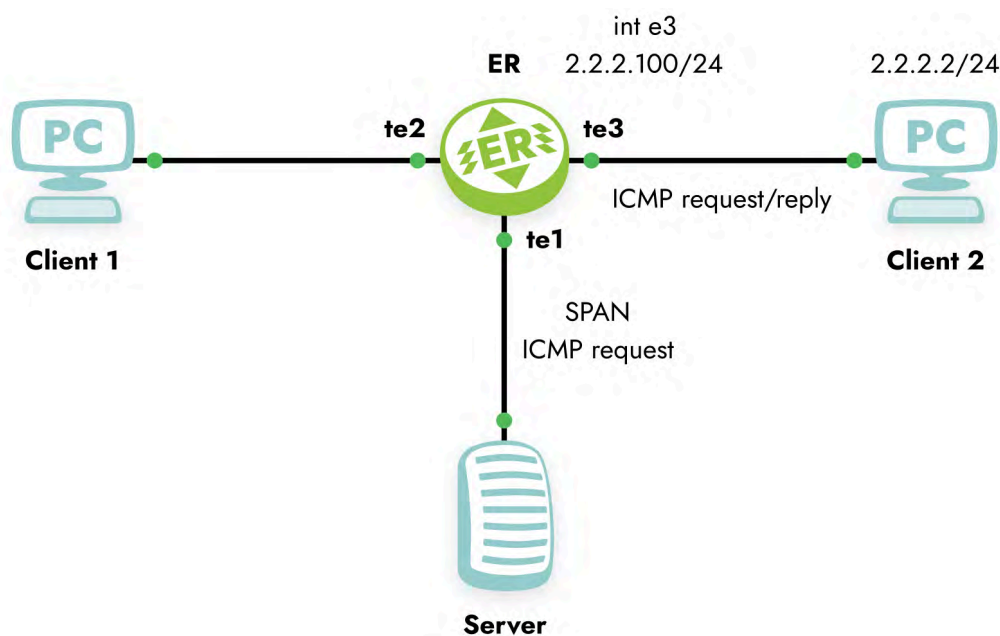


Рисунок 15

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 te1
  Total transmitted packets: 10
```

15.2.3 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 te1
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 te1
  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 счётчиков для **port te3** и **port te1**. Схема зеркалирования, реализуемая правилом **mirror-session 2** представлена ниже.

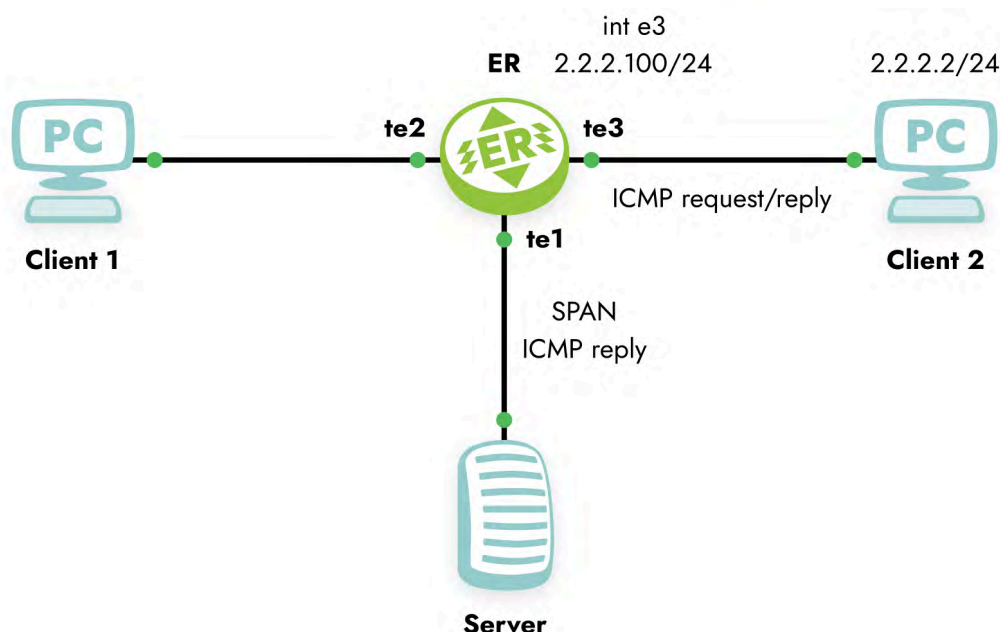


Рисунок 16

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
```

15.3 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
```


15.4 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 te1
```

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 te1
  port te2 both
  rx packets 0, bytes 0
  tx packets 17, bytes 1022
Mirror session 1 is up
  Destination: port te1
  service instance te3/3 rx
  rx packets 7, bytes 570
Mirror session 2 is up
  Destination: port te1
  interface e3 tx
  tx packets 0, bytes 0
```

Для сброса значений счётчиков правил зеркалирования используется команда `clear counters mirror-session [<название>]`. В случае, если не указано название

правила, счётчики будут обнулены для всех правил. Данная команда действует в конфигурационном режиме консоли.

16 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, подключённом к интерфейсу управления управляемого устройства. Собирает данные с управляемого устройства и передаёт 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.

16.1 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 включён 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 определённого протокола к SNMP в EcoRouter используется команда конфигурационного режима `snmp restart <bgp | isis | ldp | mrib | ospf | pim | rib | vrrp>`.

```
ecorouter(config)#snmp restart bgp
```

16.2 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 нём 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 36 — Параметры команды `snmp-server community`

Параметр	Описание
<code><COMMUNITY-NAME></code>	Community-string. Максимальная длина 32 символа
<code>view <VIEW-NAME></code>	Указать имя представления, определяющего поддерево MIB, доступное данному сообществу. Представление должно быть предварительно создано командой <code>snmp-server view</code>
<code><GROUP-NAME></code>	Имя группы
<code>ro</code>	Доступ только на чтение — значение выставляется по умолчанию
<code>rw</code>	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>`.

16.3 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 37 — Параметры команды `snmp-server view`

Параметр	Описание
----------	----------

Параметр	Описание
<code><VIEW-NAME></code>	Имя представления. Максимальная длина 32 символа
<code><OID-TREE></code>	Идентификатор поддерева MIB, которое должно быть включено в представление или исключено из него. Указывается в виде строки из цифр, разделённых точками, например, .1.3.6.2.4
<code>included</code>	Включить поддерево в SNMP представление
<code>excluded</code>	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>`.

16.4 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 38 — Параметры команды `snmp-server host`

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 задаётся, 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 >
```

16.5 Внешняя установка параметров

В EcoRouterOS доступна внешняя установка двух параметров:

- `sysContact (.1.3.6.1.2.1.1.4.0)` — наименование и контактные данные владельца устройства.
- `sysLocation (.1.3.6.1.2.1.1.6.0)` — физическое расположение устройства.

В качестве значений параметры принимают строки длиной до 255 символов.

Параметры определены в документе RFC 1158.

Чтобы иметь возможность устанавливать параметры извне, необходимо настроить SNMP-сервер и создать сообщество с правами на запись:

```
# Запуск сервиса
snmp-server enable snmp mgmt
# Создание сообщества с правами на чтение
snmp-server community com_read ro
# Создание сообщества с правами на запись
snmp-server community com_write2 rw
# Установка параметров
snmp-server contact home1
snmp-server location location1
```

Внешняя установка параметров производится с помощью NMS или с помощью сторонней утилиты, например snmpset.

Пример команд для утилиты snmpset на запись параметров:

```
snmpset -v 2c -c com_write2 192.168.255.1 1.3.6.1.2.1.1.4.0 = "home2"
snmpset -v 2c -c com_write2 192.168.255.1 1.3.6.1.2.1.1.6.0 = "location2"
```

Пример команд для утилиты snmpget на чтение параметров:

```
snmpget -v 2c -c com_read 192.168.255.1 1.3.6.1.2.1.1.4.0
snmpget -v 2c -c com_read 192.168.255.1 1.3.6.1.2.1.1.6.0
```

16.6 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.

16.6.1 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 39 — Параметры команды `snmp-server user`

Параметр	Описание
USERNAME	Имя пользователя
GROUPNAME	Имя группы
encrypted	Указание этого параметра означает, что далее введен уже зашифрованный пароль (пароли), и к нему (к ним) хэширование применять уже не нужно
auth (md5 sha)	Выбор алгоритма хэширования аутентификационного пароля. Если будет задан параметр <code>priv (des aes)</code> , 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 . Выбор возможен, только если задействован параметр <code>auth</code>
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>]]` .

16.6.2 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 40 — Параметры команды `snmp-server group`

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 определённый 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
```

Присутствует возможность включить строгий режим работы SNMP агента — при котором обрабатываются сообщения только третьей версии протокола SNMP.

```
ecorouter(config)#snmp-server v3-strict
```

Удаление группы производится при помощи команды **no snmp-server group <GROUPNAME> ((v1 | v2c | v3 (auth | noauth | priv)) (read VIEW-NAME |) (write VIEW-NAME |) |)**.

16.6.3 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

17 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 определённый 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 удалённость от первоисточника времени.

Протокол работает на основе протокола UDP, используя 123 порт.

Синхронизация с заданным NTP сервером происходит каждые 15 минут.

Таблица 41 — Команды настройки NTP

Команда	Описание
<code>ntp authentication-key <1-65535> md5 string</code>	Задание ключа для аутентификации сервера. Первое значение является порядковым номером ключа. Сам ключ задаётся в открытом виде, после чего хранится в зашифрованном
<code>ntp server <ip-адрес сервера> ... <ip-адрес сервера> <key></code>	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
<code>ntp server <ip-адрес сервера> ... <ip-адрес сервера> mgmt</code>	Указание на работу протокола только через management порт
<code>ntp server <ip-адрес сервера> ... <ip-адрес сервера> <имя виртуального маршрутизатора> <key></code>	Задание ip-адреса NTP сервера доступного через виртуальный маршрутизатор и номер ключа
<code>ntp timezone <часовой пояс UTC></code>	Задание временного пояса. Возможные значения UTC, UTC+1...UTC-12.
<code>ntp date <гггг.мм.дд> <чч:мм></code>	Задание даты и времени

17.1 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.

17.2 Команды просмотра NTP

Таблица 42 — Команды просмотра NTP

Команда	Описание
<code>show ntp status</code>	Отображает адреса ntp-серверов для синхронизации
<code>show ntp date</code>	Отображает текущую дату и время
<code>show ntp timezone</code>	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.

VM	Network Device	Network	VLAN	IP	Gateway
rtr-cod	net1	178.207.179.0/29	-	178.207.179.4	178.207.179.1
	net2	172.16.1.0/30		172.16.1.1	
fw-cod	net0	172.16.1.0/30		172.16.1.2	172.16.1.1
	net1	192.168.10.0/24	100	192.168.10.254	
		192.168.20.0/24	200	192.168.20.254	
		192.168.30.0/24	300	192.168.30.254	
		192.168.40.0/24	400	192.168.40.254	
		192.168.50.0/24	500	192.168.50.254	
sw1-cod	SVI	192.168.30.0/24	300	192.168.30.1	192.168.30.254
sw2-cod	SVI	192.168.30.0/24	300	192.168.30.2	192.168.30.254
cli-cod	net0	192.168.40.0/24	400	192.168.40.40	192.168.40.254
srv1-cod	net0	192.168.10.0/24	100	192.168.10.1	192.168.10.254
	net1	192.168.20.0/24	200	192.168.20.1	
srv2-cod	net0	192.168.10.0/24	100	192.168.10.2	192.168.10.254
	net1	192.168.20.0/24	200	192.168.20.2	
sip-cod	net0	192.168.50.0/24	500	192.168.50.50	192.168.50.254
admin-cod	net0	192.168.30.0/24	300	192.168.30.30	192.168.30.254
rtr-a	net1	178.207.179.24/29	-	178.207.179.28	178.207.179.25
	net2	172.20.10.0/24	100	172.20.10.254	
		172.20.20.0/24	200	172.20.20.254	
		172.20.30.0/24	300	172.20.30.254	
sw1-a	SVI	172.20.30.0/24	300	172.20.30.1	172.20.30.254
sw2-a	SVI	172.20.30.0/24	300	172.20.30.2	172.20.30.254
dc-a	net0	172.20.10.0/24	100	172.20.10.10	172.20.10.254
cli1-a	net0	172.20.20.0/24	200	172.20.20.1	172.20.20.254
cli2-a	net0	172.20.20.0/24	200	172.20.20.2	172.20.20.254

Последнее изменение: понедельник, 10 ноября 2025, 09:44

Предыдущий
элемент курса
[Виртуальный стенд](#)

Перейти на...

Следующий элемент
курса
[Настройка виртуальной
машины ISP](#)

Тема оформления сайта разработана

conecti.me

1. Настройка имён и IP-адресации на устройствах rtr-cod и rtr-a

[REG26](#)

- 1. Настройка имён и IP-адресации на устройствах rtr-cod и rtr-a

Вариант реализации:

rtr-cod (ecorouter):

Назначение имени на устройство:

- Для назначения имени устройства согласно требованиям задания используем следующие команды:
 1. переходим в режим администрирования (**enable**);
 2. переходим в режим конфигурации (**configure terminal**);
 3. задаём имя устройству (**hostname <NAME>**);
 4. задаём доменное имя (**ip domain-name <DOMAIN_NAME>**);
 5. сохраняем конфигурацию (**write memory**).

```
ecorouter>enable
ecorouter#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#hostname rtr-cod
rtr-cod(config)#ip domain-name cod.ssa2026.region
rtr-cod(config)#write memory
Building configuration...

rtr-cod(config)#
```

- Проверить имя устройства можно командой **show hostname** из режима администрирования (**enable**):

```
rtr-cod#show hostname
rtr-cod
rtr-cod#
rtr-cod#
```

- Проверить доменное имя устройства можно командой **show running-config | include domain-name** из режима администрирования (**enable**):

```
rtr-cod#show running-config | include domain-name
ip domain-name cod.ssa2026.region
rtr-cod#
rtr-cod#
```

Назначение IP-адресов на устройство:

- Основные понятия касающиеся EcoRouter:
 1. **Порт (port)** – это устройство в составе EcoRouter, которое работает на физическом уровне (L1);
 2. **Интерфейс (interface)** – это логический интерфейс для адресации, работает на сетевом уровне (L3);
 3. **Service instance (Сабинтерфейс, SI, Сервисный интерфейс)** является логическим сабинтерфейсом, работающим на канальном уровне (L2) и связывает L1, L2 и L3 уровни;

- 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.
- Таким образом, для того чтобы назначить IPv4-адрес на EcoRouter, необходимо придерживаться следующего алгоритма в общем виде:
 1. Создать интерфейс с произвольным именем и назначить на него IPv4-адрес;
 2. В режиме конфигурирования порта создать service-instance с произвольным именем:
 - указать (инкапсулировать) что будет обрабатываться тегированный или не тегированный трафик;
 - указать в какой интерфейс (ранее созданный) нужно отправить обработанные кадры.
- Посмотреть физические порты можно командой **show port brief** из режима администрирования (**enable**):
 - порт **te0** направлен в сторону BM **isp**;
 - порт **te1** направлен в сторону BM **fw-cod**.

```
rtr-cod#show port brief
Name          Physical  Admin  LACP  Description
-----
te0            UP        UP      *
te1            UP        UP      *
```

- Создадим интерфейс с именем **isp** и назначим на него IP-адрес **178.207.179.4/29**, также зададим для данного интерфейса описание (description - опциональный, необязательный параметр):

```
rtr-cod(config)#interface isp
rtr-cod(config-if)#ip address 178.207.179.4/29
rtr-cod(config-if)#description "Connecting to an ISP provider"
rtr-cod(config-if)#exit
rtr-cod(config)#
```

- Создадим интерфейс с именем **fw-cod** и назначим на него IP-адрес **172.16.1.1/30**, также зададим для данного интерфейса описание (description - опциональный, необязательный параметр):

```
rtr-cod(config)#interface fw-cod
rtr-cod(config-if)#ip address 172.16.1.1/30
rtr-cod(config-if)#description "Connecting to fw-cod"
rtr-cod(config-if)#exit
rtr-cod(config)#
```

- Проверить назначенные IP-адреса на интерфейсы можно командой **show ip interface brief** из режима администрирования (**enable**):
 - созданные интрфейсы пока не добавлены в какие-либо **Service instance**, а значит не привязаны и к порту, отсюда и статус **down**

```
rtr-cod#show ip interface brief
Interface      IP-Address      Status      VRF
-----
isp             178.207.179.4/29  down
fw-cod          172.16.1.1/30    down
```

- В режиме конфигурирования порта **te0** необходимо создать **service-instance** с произвольным именем, например **te0/isp**:
 - также необходимо указать (инкапсулировать) что будет обрабатываться не тегированный трафик (**untagged**);
 - и указать в какой интерфейс (ранее созданный с именем **isp**) нужно отправлять обработанные кадры.

```
rtr-cod(config)#port te0
rtr-cod(config-port)#service-instance te0/isp
rtr-cod(config-service-instance)#encapsulation untagged
rtr-cod(config-service-instance)#connect ip interface isp
rtr-cod(config-service-instance)#exit
rtr-cod(config-port)#exit
rtr-cod(config)#
```

- В режиме конфигурирования порта **te1** необходимо создать **service-instance** с произвольным именем, например **te1/fw-cod**:
 - также необходимо указать (инкапсулировать) что будет обрабатываться не тегированный трафик (**untagged**);
 - и указать в какой интерфейс (ранее созданный с именем **fw-cod**) нужно отправлять обработанные кадры.

```
rtr-cod(config-port)#service-instance te1/fw-cod
rtr-cod(config-service-instance)#encapsulation untagged
rtr-cod(config-service-instance)#connect ip interface fw-cod
rtr-cod(config-service-instance)#exit
rtr-cod(config-port)#exit
rtr-cod(config)# write memory
Building configuration...

rtr-cod(config)#
```

- Проверить назначенные IP-адреса на интерфейсы можно командой **show ip interface brief** из режима администрирования (**enable**):

```
rtr-cod#show ip interface brief
Interface          IP-Address          Status              VRF
-----
isp                178.207.179.4/29    up                  def
fw-cod             172.16.1.1/30       up                  def
rtr-cod#
```

- Проверить созданные Service instance можно командой **show service-instance brief** из режима администрирования (**enable**):

```
rtr-cod#show service-instance brief
Total instances: 2
> - Active:
N - Not connected:
D - Down:
PORT          SERVICE INSTANCE  ENDPOINT  ENDPOINT
NAME          NAME              TYPE      NAME
-----
> te0         te0/isp           iface     isp
> te1         te1/fw-cod        iface     fw-cod
rtr-cod#
```

- IP-адрес шлюза по умолчанию на данном устройстве на текущий момент не задаётся (будет рассмотрено далее, в соответствующем разделе), т.к. по условиям задания:
 - Маршрутизатор ЦОД должен получать маршрут по умолчанию по BGP
 - Ручное создание маршрута по умолчанию ЗАПРЕЩЕНО!
- Но связность с Интернет провайдером **ISP** проверить стоит:

```

rtr-cod#ping 178.207.179.1
PING 178.207.179.1 (178.207.179.1) 56(84) bytes of data.
64 bytes from 178.207.179.1: icmp_seq=1 ttl=64 time=39.9 ms
64 bytes from 178.207.179.1: icmp_seq=2 ttl=64 time=193 ms
64 bytes from 178.207.179.1: icmp_seq=3 ttl=64 time=23.8 ms

--- 178.207.179.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 23.758/85.655/193.319/76.414 ms
rtr-cod#

```

rtr-a (ecorouter):

Назначение имени на устройство:

- Реализация аналогично **rtr-cod**, за исключением соответствующего доменного имени:
 - имя устройства должно быть:

```

rtr-a#show hostname
rtr-a
rtr-a#
rtr-a#

```

- доменное имя на устройстве должно быть:

```

rtr-a#show running-config | include domain-name
ip domain-name office.ssa2026.region
rtr-a#
rtr-a#

```

Назначение IP-адресов на устройство:

- Реализация аналогично **rtr-cod**, за исключением того, что на базе физического порта **te1** должны быть созданы интерфейсы и Service instance с целью обработки тегированного трафика для предоставления возможности маршрутизации между VLAN (рассмотрено далее)
 - должен быть создан интерфейс для подключения к Интернет провайдеру ISP:

```

rtr-a#show ip interface brief
Interface      IP-Address      Status      VRF
-----
isp            178.207.179.28/29  down        defa
rtr-a#

```

- должен быть создан Service instance на порту **te0**
- на созданный Service instance реализована обработка не тегированного трафика
- в созданный Service instance добавлен интерфейс для подключения к Интернет провайдеру

```

rtr-a#show ip interface brief
Interface      IP-Address      Status      VRF
-----
isp            178.207.179.28/29  up          defa
rtr-a#show service-instance brief
Total instances: 1
> - Active:
N - Not connected:
D - Down:
PORT          SERVICE INSTANCE  ENDPOINT  ENDPOINT
NAME          NAME              TYPE      NAME
-----
> te0         te0/isp           iface     isp
rtr-a#

```


- В отличие от **rtr-cod**, на **rtr-a** по заданию нет никаких требований по настройке BGP, а значит для доступа в сеть Интернет, маршрут по умолчанию можно задать вручную:
 - переходим в режим конфигурации (**configure terminal**)
 - с помощью команды **ip route <IP_NETWORK/PREFIX> <NEXT_HOP_IP_ADDRESS>** задаём маршрут по умолчанию (шлюз)

```
rtr-a(config)#ip route 0.0.0.0/0 178.207.179.25
rtr-a(config)#
```

- Проверить назначенный маршрут по умолчанию можно командой **show ip route static** из режима администрирования (**enable**):

```
rtr-a#show ip route static
IP Route Table for VRF "default"
Gateway of last resort is 178.207.179.25 to network 0.0.0.0

S* 0.0.0.0/0 [1/0] via 178.207.179.25, isp
rtr-a#
```

- Так же стоит проверить доступ в сеть Интернет:

```
rtr-a#ping 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=51 time=104 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=51 time=99.5 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=51 time=38.6 ms
64 bytes from 77.88.8.8: icmp_seq=4 ttl=51 time=37.6 ms

--- 77.88.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3058ms
rtt min/avg/max/mdev = 37.625/69.814/103.518/31.719 ms
rtr-a#
```

- Реализуем создание под-интерфейсов для дальнейшей маршрутизации между VLAN-ами:
 - Создаём интерфейсы с произвольными именами для каждого VLAN-а и назначаем на них IP-адреса:

```
rtr-a(config)#interface v1100
rtr-a(config-if)#ip address 172.20.10.254/24
rtr-a(config-if)#description "VLAN - SRV"
rtr-a(config-if)#exit
rtr-a(config)#
rtr-a(config)#interface v1200
rtr-a(config-if)#ip address 172.20.20.254/24
rtr-a(config-if)#description "VLAN - CLI"
rtr-a(config-if)#exit
rtr-a(config)#
rtr-a(config)#interface v1300
rtr-a(config-if)#ip address 172.20.30.254/24
rtr-a(config-if)#description "VLAN - MGMT"
rtr-a(config-if)#exit
rtr-a(config)#
```

- Проверить назначенные IP-адреса на интерфейсы можно командой **show ip interface brief** из режима администрирования (**enable**):
 - созданные интерфейсы пока не добавлены в какие-либо **Service instance**, а значит не привязаны и к порту, отсюда и статус **down**

```
rtr-a#show ip interface brief
```

Interface	IP-Address	Status	VRF
isp	178.207.179.28/29	up	def
vl100	172.20.10.254/24	down	def
vl200	172.20.20.254/24	down	def
vl300	172.20.30.254/24	down	def

```
rtr-a#
```

- на базе физического интерфейса **te1** для каждого VLAN-а создаём **service-instance** с инкапсуляцией соответствующих тегов (**VID**) и подключением необходимых интерфейсов:

Пояснение к листингу команд указанному ниже

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.

Пояснение команд:

- Указание номера, обрабатываемого VLAN выполняется на service-instance с помощью команды **encapsulation dot1q <VID> exact**
 - опция **exact** показывает, что под это правило попадут кадры только с меткой равной **<VID>**
 - слово **exact** писать не обязательно, так как это поведение по умолчанию и в выводе **show run** это слово не отображается
- Указание выполняемой операции выполняется на service-instance с помощью команды **rewrite pop <Nº>**
 - ключ **1** показывает, что снимаем только одну, верхнюю метку, на L3 кадр должен поступать без признаков VLAN

```
rtr-a(config)#port te1
rtr-a(config-port)#service-instance te1/vl100
rtr-a(config-service-instance)#encapsulation dot1q 100
rtr-a(config-service-instance)#rewrite pop 1
rtr-a(config-service-instance)#connect in
rtr-a(config-service-instance)#connect ip interface vl100
rtr-a(config-service-instance)#exit
rtr-a(config-port)#
rtr-a(config-port)#service-instance te1/vl200
rtr-a(config-service-instance)#encapsulation dot1q 200
rtr-a(config-service-instance)#rewrite pop 1
rtr-a(config-service-instance)#connect ip interface vl200
rtr-a(config-service-instance)#exit
rtr-a(config-port)#
rtr-a(config-port)#service-instance te1/vl300
rtr-a(config-service-instance)#encapsulation dot1q 300
rtr-a(config-service-instance)#rewrite pop 1
rtr-a(config-service-instance)#connect ip interface vl300
rtr-a(config-service-instance)#exit
rtr-a(config-port)#exit
rtr-a(config)#write memory
Building configuration...

rtr-a(config)#
```

- Проверить назначенные IP-адреса на интерфейсы можно командой **show ip interface brief** из режима администрирования (**enable**):

```
rtr-a#show ip interface brief
Interface      IP-Address      Status      VRF
-----
isp            178.207.179.28/29 up          def
vl100          172.20.10.254/24 up          def
vl200          172.20.20.254/24 up          def
vl300          172.20.30.254/24 up          def
rtr-a#
```

- Проверить созданные Service instance можно командой **show service-instance brief** из режима администрирования (**enable**):

```
rtr-a#show service-instance brief
Total instances: 4
> - Active:
N - Not connected:
D - Down:
PORT      SERVICE INSTANCE  ENDPOINT  ENDPOINT
NAME      NAME              TYPE      NAME
-----
> te0      te0/isp           iface     isp
> te1      te1/vl100         iface     vl100
> te1      te1/vl200         iface     vl200
> te1      te1/vl300         iface     vl300
rtr-a#
```

Последнее изменение: понедельник, 10 ноября 2025, 17:15

Предыдущий
элемент курса

[Настройка виртуальной
машины ISP](#)

Следующий элемент
курса

[2. Подход к настройке
fw-cod](#)

Перейти на...

Обратная связь

Подпишитесь

Вы используете гостевой
доступ (Вход)

Сводка хранения данных

Тема оформления сайта разработана

conecti.me

2. Подход к настройке fw-cod

[REG26](#)

> 2. Подход к настройке fw-cod

Вариант реализации:

fw-cod (ideco):

Создание учетной записи администратора

- Создайте учетную запись администратора после уведомления **Создание аккаунта администратора:**

```
Ideco NGFW 21.5.99
-----
Внимание! Аккаунт администратора отсутствует.
Требуется предварительно его создать.
Создание аккаунта администратора.

Введите новый логин и нажмите Enter.
#
```

- Требования к логину и паролю:
 - Логин:
 - Не должен начинаться с цифры.
 - Не должен содержать специальных и пробельных символов, кроме дефиса -.
 - Длина логина** - от 1 до 31 символа включительно.
 - Пароль:
 - Длина пароля** - от 10 до 42 символов.
 - Содержит строчные и заглавные латинские буквы.**
 - Содержит цифры.**
 - Содержит специальные символы** (! # \$ % & ' * + и другие).
- Если пароль не соответствует требованиям политики безопасности, то появится надпись с информацией, что пароль ненадежен. Потребуется ввести новый пароль с учетом требований к нему.
- Создадим пользователя **admin** с паролем **P@ssw0rd1234**:

```
Введите новый логин и нажмите Enter.
# admin
Введите новый пароль и нажмите Enter.
Введите 'b' и нажмите Enter для возврата.
#
Повторите пароль и нажмите Enter.
Введите 'b' и нажмите Enter для возврата.
▶ каунт администратора создан успешно.
```

Настройка интерфейса управления

- Для корректной идентификации сетевой карты используйте MAC-адрес сетевой карты
- Для настройки Ideco NGFW Novum через веб-интерфейс настройте Control Plane интерфейс в локальном меню

- Control Plane - интерфейс администрирования, используется для настройки NGFW Novum через браузер и должен иметь свой выход в Интернет
- Выполните вход из под созданного пользователя **admin** с паролем **P@ssw0rd1234**

Нажмите любую клавишу для перехода к локальному меню.
Вход в локальное меню.

Введите логин и нажмите Enter.

admin

Введите пароль и нажмите Enter.

Введите 'b' и нажмите Enter для возврата.

#

Управление сервером

- Перейдите в раздел **Ethernet-интерфейсы (3)** → **Создать интерфейс (3)**:

```
1. Консоль
2. Физические порты
3. Ethernet-интерфейсы
4. Отключить все статические маршруты и добавить новый
5. Выбор типа кластерной сети
6. Включить доступ к веб-интерфейсу из внешней сети
7. Включить доступ к серверу по SSH из Интернет
8. Включить доступ к серверу по SSH из локальных сетей
9. Включить режим 'Разрешить Интернет всем'
10. Сбросить блокировки по IP
11. Отключить пользовательский межсетевой экран
12. Создать новый бэкап
13. Восстановить из бэкапа
14. Мгновенно восстановить из бэкапа
15. Включить доступ Удаленного Помощника
16. Контакты технической поддержки
17. Восстановиться на предыдущую версию
18. Перезагрузка сервера
19. Отключить сервер
. Выход
```

Введите номер пункта и нажмите Enter.

3

```
1. Показать список интерфейсов
2. Включить/Отключить интерфейс
3. Создать интерфейс
4. Изменить интерфейс
5. Удалить интерфейс
```

Введите номер пункта и нажмите Enter.

Введите 'с' и нажмите Enter для отмены.

3

Выберите порт:

N	Порт	MAC-адрес	Назначение	Производитель
1	eth0_01	bc:24:11:95:c8:01	Сервис	Red Hat, In
2	eth1_da	bc:24:11:55:66:da	Сервис	Red Hat, In
3	eth2_4f	bc:24:11:02:c4:4f	Сервис	Red Hat, In

Введите номер пункта и нажмите Enter.

Введите 'с' и нажмите Enter для отмены.

#

Пояснения в контексте текущего задания:

- Поскольку Idco рекомендуется настраивать через веб-интерфейс, надо настроить IP-адрес для этого доступа
- Поскольку в cod.ssa2026.region между fw-cod и sw1-cod необходимо организовать агрегированное соединение 802.3ad, (не актуально, от 802.3ad в задании отказались)

- о а также за маршрутизацию между VLAN-ами по топологии будет отвечать fw-cod,
- о то так называемые под-интерфейсы необходимо реализовывать поверх агрегированного канала 802.3ad,
- о в консоле Ideco - это реализовать не получится, править конфигурационные файлы в ideco плохо, можно лишиться технической поддержки.
- Поэтому как один из способов доступа к веб-интерфейсу это:
 - о использование интерфейса в сторону rtr-cod,
 - о после реализации туннеля и маршрутизации между rtr-cod и rtr-a,
 - о а также коммутации между sw1-a и sw2-a,
 - о у нас появится возможность конфигурировать fw-cod через веб-интерфейс, например с cli1-a или cli2-a,
 - о вследствие чего и полная связность между устройствами cod.ssa2026.region и office.ssa2026.region

Продолжение настройки:

- Выберите порт для доступа к веб-интерфейсу:
 - о сравнив MAC-адреса на уровне виртуальной машины

Virtual Machine 102 (fw-cod) on node 'stand0' **ideco-ngfw-novum-21**

Summary Add Remove Edit Disk Action Revert

Console Memory 16.00 GiB [balloon=0]

Hardware Processors 4 (1 sockets, 4 cores) [host]

Cloud-Init BIOS OVMF (UEFI)

Options Display Default

Task History Machine Default (i440fx)

Monitor SCSI Controller VirtIO SCSI single

Backup CD/DVD Drive (sata2) none,media=cdrom

Replication Hard Disk (scsi0) local-lvm:vm-102-disk-1,ioread=1,size=150G

Snapshots Network Device (net0) virtio=BC:24:11:95:C8:01,bridge=vibr3

Firewall Network Device (net1) virtio=BC:24:11:55:66:DA,bridge=vibr10

Permissions Network Device (net2) virtio=BC:24:11:02:C4:4F,bridge=vibr11

EFI Disk local-lvm:vm-102-disk-0,efitype=4m,pre-enrolled

Введите номер пункта и нажмите Enter.
Введите 'с' и нажмите Enter для отмены.
3

N	I	Port	MAC-адрес	Назначение	Производитель
1	I	eth0_01	bc:24:11:95:c8:01	Сервис	Red Hat, Inc.
2	I	eth1_da	bc:24:11:55:66:da	Сервис	Red Hat, Inc.
3	I	eth2_4f	bc:24:11:02:c4:4f	Сервис	Red Hat, Inc.

Введите номер пункта и нажмите Enter.
Введите 'с' и нажмите Enter для отмены.
1

- Выберите порт для доступа к веб-интерфейсу:
 - о в текущем случае выбирается порт в сторону виртуальной машины **rtr-cod**

Выберите порт:

N	I	Port	MAC-адрес	Назначение	Производитель
1	I	eth0_01	bc:24:11:95:c8:01	Сервис	Red Hat, Inc.
2	I	eth1_da	bc:24:11:55:66:da	Сервис	Red Hat, Inc.
3	I	eth2_4f	bc:24:11:02:c4:4f	Сервис	Red Hat, Inc.

Введите номер пункта и нажмите Enter.
Введите 'с' и нажмите Enter для отмены.
1

- Введите имя интерфейса:

```
Введите имя интерфейса (или оставьте пустым) и нажмите Enter.
Введите 'b' и нажмите Enter для возврата.
Введите 'c' и нажмите Enter для отмены.
# rtr-cod
```

- Выберите роль **LAN**:

```
Выберите роль для интерфейса:
1. CP
2. LAN
3. WAN

Введите номер пункта и нажмите Enter.
Введите 'c' и нажмите Enter для отмены.
# 2
```

- Выберите **Корневой контекст**:

```
Выберите VCE:
1. Системный контекст

Введите номер пункта и нажмите Enter.
Введите 'c' и нажмите Enter для отмены.
# 1
```

- Настройте локальную сеть **Вручную**:

```
Настроить адресацию на интерфейсе
1. Автоматически через DHCP
2. Вручную
3. Без адресации

Введите номер пункта и нажмите Enter.
Введите 'c' и нажмите Enter для отмены.
# 2
```

- Введите локальный IP-адрес и маску подсети в формате **ip/маска** и нажмите **Enter**:

```
Введите IP/префикс и нажмите Enter.
Введите 'b' и нажмите Enter для возврата.
Введите 'c' и нажмите Enter для отмены.
# 172.16.1.2/30
Интерфейс успешно настроен.
```

- Проверить можно выбрав соответствующий пункт меню:

```
1. Показать список интерфейсов
2. Включить/Отключить интерфейс
3. Создать интерфейс
4. Изменить интерфейс
5. Удалить интерфейс

Введите номер пункта и нажмите Enter.
Введите 'c' и нажмите Enter для отмены.
# 1
```

N	I	Состояние	Название	Роль	VCE	Port
1		Вкл.	rtr-cod	LAN	Системный контекст	eth

```
Введите номер пункта и нажмите Enter.
Введите 'c' и нажмите Enter для отмены.
#
```

- Также при вводе с клавиатуры "**с**" и нажатие **Enter** или же сочетание клавиш **Ctrl + D**:

- о можно увидеть адрес и порт для доступа к веб-интерфейсу для дальнейшей настройки

```
Добро пожаловать в панель мониторинга сервера Ideco NGFW 21.5.99!

Название сервера:                Без названия fddcbdbb-81a5-420
Состояние локальных интерфейсов:    Настроены
Доступ Удаленного Помощника:        Отключено
Режим `Разрешить Интернет всем`:      Отключено
Доступ в веб-интерфейс:              Доступен
Доступ к веб-интерфейсу из внешней сети:  Отключено

Адреса веб-интерфейса:
  https://172.16.1.2:8443

В случае возникновения ошибок на сервере, пожалуйста,
обратитесь в техподдержку:

Email: help@ideco.ru
Портал тех. поддержки: help.ideco.ru
Время работы тех. поддержки: ideco.ru/support

Нажмите любую клавишу для перехода к локальному меню.
Press Ctrl+R if you don't see the symbols above.
```

Последнее изменение: вторник, 2 декабря 2025, 16:17

Предыдущий
элемент курса

[1. Настройка имён и IP-адресации на устройствах rtr-cod и rtr-a](#)

Перейти на...

Следующий элемент
курса

[3. Настройка маршрутизации BGP на rtr-cod](#)

Обратная связь

Подпишитесь

Вы используете гостевой
доступ (Вход)

Сводка хранения данных

Тема оформления сайта разработана

conecti.me

3. Настройка маршрутизации BGP на rtr-cod

[REG26](#)

> 3. Настройка маршрутизации BGP на rtr-cod

Вариант реализации:

rtr-cod (ecorouter):

Базовая настройка BGP:

- Запустите протокол BGP, указав нужную автономную систему, командой:
router bgp <№>:

```
rtr-cod(config)#router bgp 64500
rtr-cod(config-router)#
```

- Указать уникальный идентификатор маршрутизатора в протоколе BGP, командой: **bgp router-id <IP>**:

```
rtr-cod(config-router)#bgp router-id 178.207.179.4
rtr-cod(config-router)#
```

- Сконфигурируйте BGP соседство с Интернет провайдером **ISP**, указав адрес соседа и номер локальной AS, используя команду: **neighbor <NEIGHBOR_IP> remote-as <\$>**:

```
rtr-cod(config-router)#neighbor 178.207.179.1 remote-as 31133
rtr-cod(config-router)#exit
rtr-cod(config)#write memory
Building configuration...

rtr-cod(config)#
```

- Проверить состояние всех соединений BGP можно командой **show ip bgp summary** из режима администрирования (**enable**):

```
rtr-cod#show ip bgp summary
BGP router identifier 178.207.179.4, local AS number 64500
BGP table version is 2
1 BGP AS-PATH entries
0 BGP community entries

Neighbor      V    AS    MsgRcv  MsgSen  TblVer  InQ    Ou
-----
178.207.179.1  4    31133   7        5        2        0        0

Total number of neighbors 1
Total number of Established sessions 1
rtr-cod#
```

- Также по условиям задания **rtr-cod** должен получать маршрут по умолчанию по BGP
- Проверить маршрут по умолчанию можно командой **show ip route** из режима администрирования (**enable**):

```

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

IP Route Table for VRF "default"
Gateway of last resort is 178.207.179.1 to network 0.0.0.0

B*    0.0.0.0/0 [20/0] via 178.207.179.1, isp, 00:00:16
C     172.16.1.0/30 is directly connected, fw-cod
C     178.207.179.0/29 is directly connected, isp
rtr-cod#

```

- Проверить доступ в сеть Интернет:

```

rtr-cod#ping 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=51 time=187 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=51 time=89.9 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=51 time=91.9 ms
64 bytes from 77.88.8.8: icmp_seq=4 ttl=51 time=95.2 ms

--- 77.88.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 89.859/115.988/186.932/41.004 ms
rtr-cod#

```

Последнее изменение: понедельник, 10 ноября 2025, 17:12

Предыдущий
элемент курса

[2. Подход к настройке
fw-cod](#)

Перейти на...

Следующий элемент
курса

[4. Настройка туннелей
между офисом «а» и
«cod»](#)

Обратная связь

Вы используете гостевой
доступ (Вход)

Подпишитесь

Сводка хранения данных

Тема оформления сайта разработана

conecti.me

4. Настройка туннелей между офисом «а» и «cod»

[REG26](#)

- > 4. Настройка туннелей между офисом «а» и «cod»

Вариант реализации:

rtr-cod (ecorouter):

Настройка туннеля GRE между устройствами:

- Создайте интерфейс туннеля с именем **tunnel.0**:

```
rtr-cod(config)#interface tunnel.0
rtr-cod(config-if-tunnel)#
```

- Назначьте ip адрес в соответствие с требованиями задания:

```
rtr-cod(config-if-tunnel)#ip address 10.10.10.1/30
rtr-cod(config-if-tunnel)#
```

- Задайте режим работы туннеля GRE и адресов начала (источника - **rtr-cod**) и конца туннеля (назначения - **rtr-a**):

```
rtr-cod(config-if-tunnel)#ip tunnel 178.207.179.4 178.207.179.28
mode gre
rtr-cod(config-if-tunnel)# exit
rtr-cod(config)#write memory
Building configuration...

rtr-cod(config)#
```

- Для просмотра состояния туннеля используется команда **show interface tunnel.0** из режима администрирования (**enable**):

```
rtr-cod#show interface tunnel.0
Interface tunnel.0 is up
  Snmp index: 8
  Ethernet address: (port not configured)
  MTU: 1476
  Tunnel source: 178.207.179.4
  Tunnel destination: 178.207.179.28
  Tunnel mode: GRE
  Tunnel keepalive: disabled
  NAT: no
  ARP Proxy: disable
  ICMP redirects on, unreachable on
  IP URPF is disabled
  Label switching is disabled
  <UP, BROADCAST, RUNNING, NOARP, MULTICAST>
  inet 10.10.10.1/30 broadcast 10.10.10.3/30
  total input packets 0, bytes 0
  total output packets 0, bytes 0
rtr-cod#
```

rtr-a (ecorouter):

Настройка туннеля GRE между устройствами:

- Реализация аналогично **rtr-cod**, за исключением соответствующего IP-адресов **источника** и **назначения**:
 - состояния туннеля должно быть:

```
rtr-a#show interface tunnel.0
Interface tunnel.0 is up
  Snmp index: 9
  Ethernet address: (port not configured)
  MTU: 1476
  Tunnel source: 178.207.179.28
  Tunnel destination: 178.207.179.4
  Tunnel mode: GRE
  Tunnel keepalive: disabled
  NAT: no
  ARP Proxy: disable
  ICMP redirects on, unreachable on
  IP URPF is disabled
  Label switching is disabled
  <UP, BROADCAST, RUNNING, NOARP, MULTICAST>
  inet 10.10.10.2/30 broadcast 10.10.10.3/30
  total input packets 0, bytes 0
  total output packets 0, bytes 0
rtr-a#
```

- Должна быть связность по туннелю:

```
rtr-a#show ip interface brief
Interface      IP-Address      Status      VRF
-----
isp            178.207.179.28/29  up          def
vl100          172.20.10.254/24  up          def
vl200          172.20.20.254/24  up          def
vl300          172.20.30.254/24  up          def
tunnel.0       10.10.10.2/30     up          def
rtr-a#ping 10.10.10.1
PING 10.10.10.1 (10.10.10.1) 56(84) bytes of data.
64 bytes from 10.10.10.1: icmp_seq=1 ttl=64 time=106 ms
64 bytes from 10.10.10.1: icmp_seq=2 ttl=64 time=96.8 ms
64 bytes from 10.10.10.1: icmp_seq=3 ttl=64 time=111 ms
64 bytes from 10.10.10.1: icmp_seq=4 ttl=64 time=93.2 ms

--- 10.10.10.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 93.181/101.848/111.388/7.220 ms
rtr-a#
```

Последнее изменение: понедельник, 10 ноября 2025, 17:13

Предыдущий
элемент курса

[3. Настройка
маршрутизации BGP на
rtr-cod](#)

Перейти на...

Следующий элемент
курса

[5. Настройка доступа в
Интернет](#)

Обратная связь

Вы используете гостевой
доступ (Вход)

Подпишитесь

Сводка хранения данных

Тема оформления сайта разработана

connecti.me

5. Настройка доступа в Интернет

[REG26](#)

> 5. Настройка доступа в Интернет

Вариант реализации:

rtr-cod (ecorouter):

Настройка dynamic PAT:

- Определяемся с внутренними (**inside**) и внешними (**outside**) интерфейсами с точки зрения технологии **NAT**:
 - интерфейс **isp** - внешний (**outside**);
 - интерфейс **fw-cod** - внутренний (**inside**)

```
rtr-cod#show ip interface brief
Interface          IP-Address          Status      VRF
-----
tunnel.0           10.10.10.1/30       up          defa
isp                 178.207.179.4/29    up          defa
fw-cod              172.16.1.1/30       up          defa
rtr-cod#show interface description
Interface          Status      Protocol    Description
-----
isp                up          up           "Connecting t
fw-cod             up          up           "Connecting t
tunnel.0           up          up
rtr-cod#
```

- Назначаем интерфейс **isp** как **ip nat outside**:

```
rtr-cod(config)#interface isp
rtr-cod(config-if)#ip nat outside
rtr-cod(config-if)#exit
rtr-cod(config)#
```

- Назначаем интерфейс **fw-cod** как **ip nat inside**:

```
rtr-cod(config)#interface fw-cod
rtr-cod(config-if)#ip nat inside
rtr-cod(config-if)#exit
rtr-cod(config)#
```

- Создаём **nat pool**, чтобы указать диапазоны IP-адресов, который в дальнейшем будут попадать под правила трансляции:
 - необходимо не только указать диапазон IP-адресов из сети между **rtr-cod** и **fw-cod**
 - но и диапазоны IP-адресов из сетей:
 - 192.168.10.0/24 - vlan100
 - 192.168.30.0/24 - vlan300
 - 192.168.40.0/24 - vlan400
 - 192.168.50.0/24 - vlan500
 - т.к. в дальнейшем будет реализована маршрутизация между **rtr-cod** и **fw-cod**
 - диапазон 192.168.20.0/24 - vlan200 указывать не надо, т.к. по условиям задания
 - "Для «cod» трафик VLAN - DATA не должен маршрутизироваться"

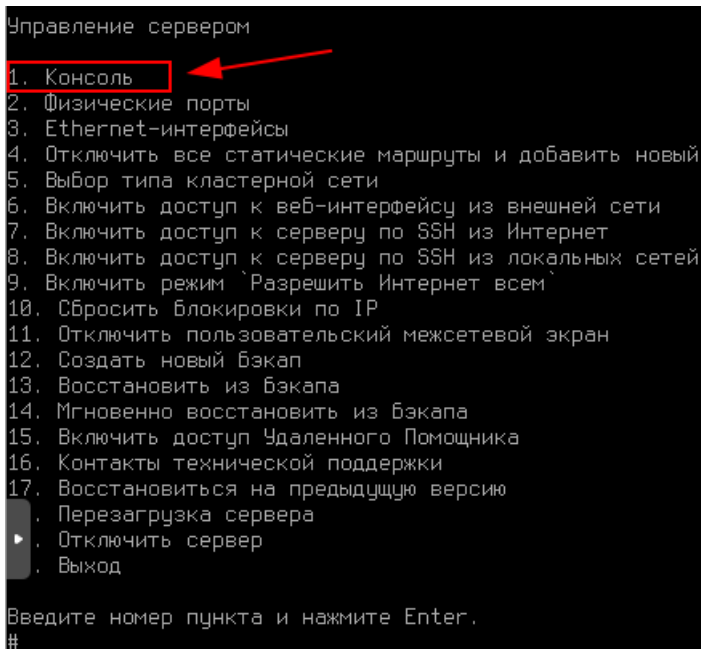
```
rtr-cod(config)#ip nat pool fw-cod 172.16.1.1-172.16.1.2
rtr-cod(config)#ip nat pool vlan100 192.168.10.1-192.168.10.254
rtr-cod(config)#ip nat pool vlan300 192.168.30.1-192.168.30.254
rtr-cod(config)#ip nat pool vlan400 192.168.40.1-192.168.40.254
rtr-cod(config)#ip nat pool vlan500 192.168.50.1-192.168.50.254
rtr-cod(config)#
```

- Создать правило трансляции адресов для каждого созданного **nat pool**, через интерфейс, который с точки зрения NAT **outside**:

```
rtr-cod(config)#ip nat source dynamic inside pool fw-cod overload
interface isp
rtr-cod(config)#ip nat source dynamic inside pool vlan100 overload
interface isp
rtr-cod(config)#ip nat source dynamic inside pool vlan300 overload
interface isp
rtr-cod(config)#ip nat source dynamic inside pool vlan400 overload
interface isp
rtr-cod(config)#ip nat source dynamic inside pool vlan500 overload
interface isp
rtr-cod(config)#write memory
Building configuration...

rtr-cod(config)#
```

- Проверить доступ в сеть Интернет с виртуальной машины **fw-cod** используя **консоль**:



```
Управление сервером
1. Консоль
2. Физические порты
3. Ethernet-интерфейсы
4. Отключить все статические маршруты и добавить новый
5. Выбор типа кластерной сети
6. Включить доступ к веб-интерфейсу из внешней сети
7. Включить доступ к серверу по SSH из Интернет
8. Включить доступ к серверу по SSH из локальных сетей
9. Включить режим `Разрешить Интернет всем`
10. Сбросить блокировки по IP
11. Отключить пользовательский межсетевой экран
12. Создать новый бэкап
13. Восстановить из бэкапа
14. Мгновенно восстановить из бэкапа
15. Включить доступ Удаленного Помощника
16. Контакты технической поддержки
17. Восстановиться на предыдущую версию
. Перезагрузка сервера
. Отключить сервер
. Выход
Введите номер пункта и нажмите Enter.
#
```

- временно назначаем адрес шлюза по умолчанию, используя команду **ip route add**:

```
ip route add 0.0.0.0/0 via 172.16.1.1
```

- Проверяем доступ в сеть Интернет:


```
[admin@bez-nazvaniya-fddcbdbb-81a5-420d-bece-e00da33673ff ~]# ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data:
64 bytes from 77.88.8.8: icmp_seq=1 ttl=50 time=88.8 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=50 time=88.5 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=50 time=86.9 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 86.943/88.072/88.770/0.805 ms
[admin@bez-nazvaniya-fddcbdbb-81a5-420d-bece-e00da33673ff ~]#
```

- На **rtr-cod** проверяем таблицу трансляции адресов командой **show ip nat translations** из режима администрирования (**enable**):

```
rtr-cod#show ip nat translations
Static translations:

Source                               Translated                           VRF

Destination                         Translated                           VRF

Empty list.
Total: 0

PAT translations:
```

Source	Translated	Destination
Time: 9s, Protocol: UDP, VRF: default		
IN: 172.16.1.2:21260	178.207.179.4:1024	216.239.32.1
OUT: 216.239.32.105:53	172.16.1.2:21260	178.207.179.
Time: 9s, Protocol: UDP, VRF: default		
IN: 172.16.1.2:15566	178.207.179.4:1025	84.201.185.2
OUT: 84.201.185.208:53	172.16.1.2:15566	178.207.179.
Time: 1s, Protocol: ICMP, VRF: default		
IN: 172.16.1.2	178.207.179.4	77.88.8.8
OUT: 77.88.8.8	172.16.1.2	178.207.179.

- Так же на данном этапе стоит добавить статические маршруты в локальные сети COD-a:
 - динамическую маршрутизацию, например OSPF между **rtr-cod** и **fw-cod** по условиям задания реализовать нельзя, т.к.
 - на **rtr-cod** в дальнейшем для обеспечения динамической маршрутизации между офисом «а» и «cod»
 - "Все интерфейсы, кроме туннельных, должны быть переведены в пассивный режим"
 - так же не стоит добавлять маршрут в сеть 192.168.20.0/24 (vlan200), т.к.
 - "Для «cod» трафик VLAN - DATA не должен маршрутизироваться"

```
rtr-cod(config)#ip route 192.168.10.0/24 172.16.1.2
rtr-cod(config)#ip route 192.168.30.0/24 172.16.1.2
rtr-cod(config)#ip route 192.168.40.0/24 172.16.1.2
rtr-cod(config)#ip route 192.168.50.0/24 172.16.1.2
rtr-cod(config)#
```

- Проверить таблицу маршрутизации можно командой **show ip route** из режима администрирования (**enable**):

```

rtr-cod#show ip route
Codes: 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
       * - candidate default

IP Route Table for VRF "default"
Gateway of last resort is 178.207.179.1 to network 0.0.0.0

B*    0.0.0.0/0 [20/0] via 178.207.179.1, isp, 00:03:17
C     10.10.10.0/30 is directly connected, tunnel.0
C     172.16.1.0/30 is directly connected, fw-cod
C     178.207.179.0/29 is directly connected, isp
S     192.168.10.0/24 [1/0] via 172.16.1.2, fw-cod
S     192.168.30.0/24 [1/0] via 172.16.1.2, fw-cod
S     192.168.40.0/24 [1/0] via 172.16.1.2, fw-cod
S     192.168.50.0/24 [1/0] via 172.16.1.2, fw-cod
rtr-cod#

```

rtr-a (ecorouter):

Настройка dynamic PAT:

- Реализация аналогично **rtr-cod**, за исключением:
 - интерфейсы с точки зрения NAT должны быть:

```

rtr-a#show interface isp | grep NAT
NAT: outside
rtr-a#show interface vl100 | grep NAT
NAT: inside
rtr-a#show interface vl200 | grep NAT
NAT: inside
rtr-a#show interface vl300 | grep NAT
NAT: inside
rtr-a#

```

- **nat pool** должны быть:

```

rtr-a#show run | grep pool
ip nat pool vlan100 172.20.10.1-172.20.10.254
ip nat pool vlan200 172.20.20.1-172.20.20.254
ip nat pool vlan300 172.20.30.1-172.20.30.254
rtr-a#

```

- созданные правила трансляции:

```

rtr-a#show run | grep overload
ip nat source dynamic inside-to-outside pool vlan100 overload in
ip nat source dynamic inside-to-outside pool vlan200 overload in
ip nat source dynamic inside-to-outside pool vlan300 overload in
rtr-a#

```

- Доступ в сеть Интернет можно проверить с виртуальной машины **sw1-a** используя команды временного назначения IP-адресов и шлюза:
 - назначив средствами **iproute2** временно на интерфейс, смотрящий в сторону **rtr-a (ens19)**,
 - создав тегированный подинтерфейс с IP-адресом из подсети для **vlan300**

```

ip link add link ens19 name ens19.300 type vlan id 300
ip link set dev ens19.300 up
ip addr add 172.20.30.1/24 dev ens19.300
ip route add 0.0.0.0/0 via 172.20.30.254

```

- Доступ в сеть Интернет с **sw1-a**:

```
[root@localhost ~]# ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=50 time=38.6 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=50 time=38.6 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=50 time=39.7 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 38.598/38.981/39.732/0.530 ms
[root@localhost ~]#
```

- На **rtr-a** проверяем таблицу трансляции адресов командой **show ip nat translations** из режима администрирования (**enable**):

```
PAT translations:

      Source                Translated                Destination
Time: 29s, Protocol: ICMP, VRF: default
IN: 172.20.30.1             178.207.179.28           77.88.8.8
OUT: 77.88.8.8              172.20.30.1             178.207.179.28

Time: 35s, Protocol: ICMP, VRF: default
IN: 172.20.30.1             178.207.179.28           77.88.8.8
OUT: 77.88.8.8              172.20.30.1             178.207.179.28
```

Последнее изменение: четверг, 13 ноября 2025, 13:32

Предыдущий
элемент курса

[4. Настройка туннелей
между офисом «а» и
«cod»](#)

Перейти на...

Следующий элемент
курса

[6. Настройка
динамической
маршрутизации между
офисом «а» и «cod»](#)

Обратная связь

Подпишитесь

Вы используете гостевой
доступ (Вход)

Сводка хранения данных

Тема оформления сайта разработана

connecti.me

6. Настройка динамической маршрутизации между офисом «а» и «cod»

[REG26](#)

- 6. Настройка динамической маршрутизации между офисом «а» и «cod»

Вариант реализации:

rtr-cod (ecorouter):

Настройка динамической маршрутизации OSPF:

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

```
rtr-cod(config)#router ospf 1
rtr-cod(config-router)#
```

- Сконфигурируйте OSPF идентификатор маршрутизатора (необязательный этап)
 - Используйте команду **ospf router-id <значение>**, в качестве значения укажем туннельный IP-адрес маршрутизатора:

```
rtr-cod(config-router)#router-id 10.10.10.1
rtr-cod(config-router)#
```

- Переводим все интерфейсы в пассивный режим:

```
rtr-cod(config-router)#passive-interface default
rtr-cod(config-router)#
```

- Исключаем интерфейс **tunnel.0** из пассивного режима для установления соседства и дальнейшего обмена маршрутной информацией:

```
rtr-cod(config-router)#no passive-interface tunnel.0
rtr-cod(config-router)#
```

- Объявляем сети, которые будут задействованы в процессе маршрутизации:
 - импортировав (**redistribute**) ранее указанные статические маршруты (до локальных сетей COD-а) в процесс OSPF

```
rtr-cod(config-router)#redistribute static
rtr-cod(config-router)#network 10.10.10.0/30 area 0
rtr-cod(config-router)#network 172.16.1.0/30 area 0
rtr-cod(config-router)#exit
rtr-cod(config)#
```

- Обеспечиваем защиту протокола маршрутизации посредством парольной защиты:

```

rtr-cod(config)#interface tunnel.0
rtr-cod(config-if-tunnel)#ip ospf authentication message-digest
rtr-cod(config-if-tunnel)#ip ospf message-digest-key 1 md5 P@ssw0rd
rtr-cod(config-if-tunnel)#exit
rtr-cod(config)#write memory
Building configuration...

rtr-cod(config)#

```

- Просмотр данных о состоянии и сконфигурированных настройках на интерфейсах, участвующих в OSPF процессе можно командой **show ip ospf interface** из режима администрирования (**enable**):

```

rtr-cod#show ip ospf interface
tunnel.0 is up, line protocol is up
Internet Address 10.10.10.1/30, Area 0.0.0.0, MTU 1476
Process ID 1, VRF (default), Router ID 10.10.10.1, Network Typ
Transmit Delay is 1 sec, State DR, Priority 1, TE Metric 1
Designated Router (ID) 10.10.10.1, Interface Address 10.10.10.
No backup designated router on this network
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retran
Hello due in 00:00:10
Neighbor Count is 0, Adjacent neighbor count is 0
Crypt Sequence Number is 2764
Hello received 0 sent 17, DD received 0 sent 0
LS-Req received 0 sent 0, LS-Upd received 0 sent 0
LS-Ack received 0 sent 0, Discarded 0
Message-digest authentication, using key-id 1
rtr-cod#

```

rtr-a (ecorouter):

Настройка динамической маршрутизации OSPF:

- Реализация аналогично **rtr-cod**, за исключением:
 - должны быть указаны следующие сети:

```

rtr-a#show running-config ospf
!
!
router ospf 1
ospf router-id 10.10.10.2
passive-interface default
no passive-interface tunnel.0
network 10.10.10.0/30 area 0.0.0.0
network 172.20.10.0/24 area 0.0.0.0
network 172.20.20.0/24 area 0.0.0.0
network 172.20.30.0/24 area 0.0.0.0
!
interface tunnel.0
ip ospf authentication message-digest
ip ospf message-digest-key 1 md5 0xeb5442f9b26607db
!
rtr-a#

```

- Просмотр сведений о соседских отношениях между OSPF маршрутизаторами можно командой **show ip ospf neighbor** из режима администрирования (**enable**):

```

rtr-a#show ip ospf neighbor

Total number of full neighbors: 1
OSPF process 1 VRF(default):
Neighbor ID    Pri   State           Dead Time   Address
10.10.10.1     1     Full/Backup     00:00:34   10.10.10.1
rtr-a#

```

- Проверить таблицу маршрутизации можно командой **show ip route** из режима администрирования (**enable**):

```

rtr-a#show ip route
Codes: 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
       * - candidate default

IP Route Table for VRF "default"
Gateway of last resort is 178.207.179.25 to network 0.0.0.0

S*      0.0.0.0/0 [1/0] via 178.207.179.25, isp
C       10.10.10.0/30 is directly connected, tunnel.0
O       172.16.1.0/30 [110/2] via 10.10.10.1, tunnel.0, 00:00:13
C       172.20.10.0/24 is directly connected, vl100
C       172.20.20.0/24 is directly connected, vl200
C       172.20.30.0/24 is directly connected, vl300
C       178.207.179.24/29 is directly connected, isp
O E2    192.168.10.0/24 [110/20] via 10.10.10.1, tunnel.0, 00:00:00
O E2    192.168.30.0/24 [110/20] via 10.10.10.1, tunnel.0, 00:00:00
O E2    192.168.40.0/24 [110/20] via 10.10.10.1, tunnel.0, 00:00:00
O E2    192.168.50.0/24 [110/20] via 10.10.10.1, tunnel.0, 00:00:00
rtr-a#

```

- Аналогично на rtr-cod:

```

rtr-cod#show ip ospf neighbor
Total number of full neighbors: 1
OSPF process 1 VRF(default):
Neighbor ID Pri State Dead Time Address
10.10.10.2 1 Full/DR 00:00:37 10.10.10.2
rtr-cod#show ip route
Codes: 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
       * - candidate default

IP Route Table for VRF "default"
Gateway of last resort is 178.207.179.1 to network 0.0.0.0

B*      0.0.0.0/0 [20/0] via 178.207.179.1, isp, 00:04:57
C       10.10.10.0/30 is directly connected, tunnel.0
C       172.16.1.0/30 is directly connected, fw-cod
O       172.20.10.0/24 [110/2] via 10.10.10.2, tunnel.0, 00:00:43
O       172.20.20.0/24 [110/2] via 10.10.10.2, tunnel.0, 00:03:22
O       172.20.30.0/24 [110/2] via 10.10.10.2, tunnel.0, 00:03:22
C       178.207.179.0/29 is directly connected, isp
S       192.168.10.0/24 [1/0] via 172.16.1.2, fw-cod
S       192.168.30.0/24 [1/0] via 172.16.1.2, fw-cod
S       192.168.40.0/24 [1/0] via 172.16.1.2, fw-cod
S       192.168.50.0/24 [1/0] via 172.16.1.2, fw-cod
rtr-cod#

```

- Также можно проверить связность между sw1-a и fw-cod:
 - если ранее указанный временный маршрут по умолчанию на fw-cod всё ещё присутствует,
 - а на sw1-a ранее указанный временный IP-адрес присутствует

```
[root@localhost ~]# ping -c3 172.16.1.2
PING 172.16.1.2 (172.16.1.2) 56(84) bytes of data.
64 bytes from 172.16.1.2: icmp_seq=1 ttl=62 time=130 ms
64 bytes from 172.16.1.2: icmp_seq=2 ttl=62 time=130 ms
64 bytes from 172.16.1.2: icmp_seq=3 ttl=62 time=130 ms

--- 172.16.1.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 129.508/129.920/130.158/0.292 ms
[root@localhost ~]# tracepath -n 172.16.1.2
 0: 172.16.1.2 [LOCALHOST]  pmtu 1500
 1: 172.20.30.254            72.228ms
 1: 172.20.30.254            21.991ms
 2: 172.20.30.254            64.922ms pmtu 1300
 2: 10.10.10.1              97.804ms
 3: 172.16.1.2              99.717ms reached

Resume: pmtu 1476 hops 3 back 3
[root@localhost ~]#
```

Последнее изменение: вторник, 11 ноября 2025, 17:21

Предыдущий
элемент курса

[5. Настройка доступа в
Интернет](#)

Перейти на...

Следующий элемент
курса

[7. Настройка
коммутации между sw1-
а и sw2-а](#)

Обратная связь

Вы используете гостевой
доступ (Вход)

Подпишитесь

Сводка хранения данных

Тема оформления сайта разработана

conecti.me

7. Настройка коммутации между sw1-а и sw2-а

[REG26](#)

> 7. Настройка коммутации между sw1-а и sw2-а

Вариант реализации:

sw1-a (alt-server):

Назначение имени на устройство:

- Для назначения имени устройства согласно топологии используем следующую команду:

```
hostnamectl set-hostname sw1-a.office.ssa2026.region; exec bash
```

- Так же рекомендуется указать имя в файле `/etc/sysconfig/network`:

```
vim /etc/sysconfig/network
```

- указать имя в параметре **HOSTNAME**:

```
# When set to no, this may cause most daemons' initscripts skip starting
NETWORKING=yes

# Used by hotplug/pcmcia/ifplugd scripts to detect current network config
# subsystem.
CONFMETHOD=etcnet

# Used by rc.sysinit to setup system hostname at boot.
HOSTNAME=sw1-a.office.ssa2026.region

# This is used by ALTlinux ppp-common to decide if we want to install
# nameserver lines into /etc/resolv.conf or not.
RESOLV_MODS=yes
```

- Проверить можно с помощью команды **hostname** с ключём **-f**:

```
[root@sw1-a ~]# hostname -f
sw1-a.office.ssa2026.region
[root@sw1-a ~]#
```

Установка пакета Open vSwitch:

- Для установки пакета **openvswitch** необходим доступ в сеть Интернет, для этого необходимо на виртуальной машине **sw1-a**:
 - назначив средствами **iproute2** временно на интерфейс, смотрящий в сторону **rtr-a (ens19)**,
 - тегированный подинтерфейс с IP-адресом из подсети для **vlan300**, а также шлюзом по умолчанию и DNS

```
ip link add link ens19 name ens19.300 type vlan id 300
ip link set dev ens19.300 up
ip addr add 172.20.30.1/24 dev ens19.300
ip route add 0.0.0.0/0 via 172.20.30.254
echo "nameserver 77.88.8.8" > /etc/resolv.conf
```

- После чего обновляем список пакетов и устанавливаем **openvswitch**:

```
apt-get update && apt-get install -y openvswitch
```

- Включаем и добавляем в автозагрузку **openvswitch**:

```
systemctl enable --now openvswitch
```

- Правим основной файл **options** в котором по умолчанию сказано:
 - удалять настройки заданные через **ovs-vsctl**,
 - т.к. через **etcdnet** будет выполнено только создание интерфейса типа **internal**
 - с назначением необходимого IP-адреса, а настройка коммутации будет выполнена средствами **openvswitch**

```
sed -i "s/OVS_REMOVE=yes/OVS_REMOVE=no/g"  
/etc/net/ifaces/default/options
```

- Перезагрузить сервер - будет быстрее чем удалять параметры заданные в ручную через пакет **iproute2**:

```
reboot
```

- Проверяем интерфейсы и определяемся какой к кому направлен:
 - таким образом, имеем:
 - **ens19** - интерфейс в сторону **rtr-a**;
 - **ens20** - интерфейс в сторону **dc-a**;
 - **ens21** - интерфейс в сторону **sw2-a**;
 - **ens22** - интерфейс в сторону **sw2-a**;

```
[root@localhost ~]# ip -c -br a  
lo                UNKNOWN  
ens19             UP  
ens20             DOWN  
ens21             DOWN  
ens22             DOWN  
[root@localhost ~]#
```

- Поднимаем физические интерфейсы, создавая директорию для каждого интерфейса в **/etc/net/ifaces** и описывая файл **options**:
 - если в файле **options** для порты **ens19** есть параметры **TYPE=eth** и **BOOTPROTO=static**,
 - то можно рекурсивно выполнить копирование **ens19** во все необходимые каталоги

```
cp -r /etc/net/ifaces/ens19 /etc/net/ifaces/ens20
```

```
cp -r /etc/net/ifaces/ens19 /etc/net/ifaces/ens21
```

```
cp -r /etc/net/ifaces/ens19 /etc/net/ifaces/ens22
```

- Перезагружаем службу **network**:

```
systemctl restart network
```

- Проверить, что интерфейсы перешли из статуса **DOWN** в статус **UP**:

```
[root@sw1-a ~]# ip -c -br a  
lo                UNKNOWN  
ens19             UP  
ens20             UP  
ens21             UP  
ens22             UP  
[root@sw1-a ~]#
```

- Чтобы на **sw2-a** была возможность установить пакет **openvswitch**:
 - временно создадим простой коммутатор с именем, например **br0**
 - и добавим в него интерфейсы **ens19** в сторону **rtr-a** и **ens21** в сторону **sw2-a**

```
ovs-vsctl add-br br0
```

```
ovs-vsctl add-port br0 ens19
```

```
ovs-vsctl add-port br0 ens21
```

sw2-a (alt-server):

Назначение имени на устройство:

- Реализация аналогично **sw1-a**:

```
[root@sw2-a ~]# hostname -f
sw2-a.office.ssa2026.region
[root@sw2-a ~]#
```

- Для установки пакета **openvswitch** необходим доступ в сеть Интернет, для этого необходимо на виртуальной машине **sw2-a**:
 - назначив средствами **iproute2** временно на интерфейс, смотрящий в сторону **sw1-a (ens19)**,
 - тегированный подинтерфейс с IP-адресом из подсети для **vlan300**, а также шлюзом по умолчанию и DNS

```
ip link add link ens19 name ens19.300 type vlan id 300
ip link set dev ens19.300 up
ip addr add 172.20.30.2/24 dev ens19.300
ip route add 0.0.0.0/0 via 172.20.30.254
echo "nameserver 77.88.8.8" > /etc/resolv.conf
```

- После чего обновляем список пакетов и устанавливаем **openvswitch**:

```
apt-get update && apt-get install -y openvswitch
```

- Включаем и добавляем в автозагрузку **openvswitch**:

```
systemctl enable --now openvswitch
```

- Правим основной файл **options** в котором по умолчанию сказано:
 - удалять настройки заданные через **ovs-vsctl**,
 - т.к. через **etcdnet** будет выполнено только создание интерфейса типа **internal**
 - с назначением необходимого IP-адреса, а настройка коммутации будет выполнена средствами **openvswitch**

```
sed -i "s/OVS_REMOVE=yes/OVS_REMOVE=no/g"
/etc/net/ifaces/default/options
```

- Перезагрузить сервер - будет быстрее чем удалять параметры заданные вручную через пакет **iproute2**:

```
reboot
```

sw1-a (alt-server):

Настройка коммутации:

- Удаляем ранее созданный временный коммутатор с именем **br0**:

```
ovs-vsctl del-br br0
```

- Создадим коммутатор с именем **sw1-a**:

```
ovs-vsctl add-br sw1-a
```

- Проверить создание коммутатора можно с помощью команды **ovs-vsctl show**:

```
[root@sw1-a ~]# ovs-vsctl show
245b4b82-d2c3-4e24-a664-0df54cf70786
    Bridge sw1-a
        Port sw1-a
            Interface sw1-a
                type: internal
        ovs_version: "3.3.2"
[root@sw1-a ~]#
```

- Добавим интерфейс, направленный в сторону **dc-a** (ens20) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 100**:

```
ovs-vsctl add-port sw1-a ens20 tag=100
```

- Интерфейс в сторону **rtr-a** (ens19) добавляем в созданный коммутатор, но настраиваем как магистральный (trunk) порт:
 - также разрешаем пропуск только требуемых VLAN (100,200 и 300)

```
ovs-vsctl add-port sw1-a ens19 trunk=100,200,300
```

- Интерфейсы в стороны **sw2-a** (ens21, ens22) добавляем в созданный коммутатор, но настраиваем как магистральный (trunk) порт:
 - также разрешаем пропуск только требуемых VLAN (100,200 и 300)

```
ovs-vsctl add-port sw1-a ens21 trunk=100,200,300
```

```
ovs-vsctl add-port sw1-a ens22 trunk=100,200,300
```

- Проверить добавление портов в коммутатор можно с помощью команды **ovs-vsctl show**:

```
[root@sw1-a ~]# ovs-vsctl show
245b4b82-d2c3-4e24-a664-0df54cf70786
    Bridge sw1-a
        Port ens19
            trunks: [100, 200, 300]
            Interface ens19
        Port sw1-a
            Interface sw1-a
                type: internal
        Port ens21
            trunks: [100, 200, 300]
            Interface ens21
        Port ens22
            trunks: [100, 200, 300]
            Interface ens22
        Port ens20
            tag: 100
            Interface ens20
        ovs_version: "3.3.2"
[root@sw1-a ~]#
```

- Включаем модуль ядра отвечающий за тегированный трафик (**802.1Q**):

```
modprobe 8021q
```

```
echo "8021q" | tee -a /etc/modules
```

- Запускаем процесс STP на коммутаторе:
 - в режиме **802.1w** (rstp)

```
ovs-vsctl set bridge sw1-a rstp_enable=true
```

```
ovs-vsctl set bridge sw1-a other_config:stp-protocol=rstp
```

- Задаём данному коммутатору наименьший приоритет:

```
ovs-vsctl set bridge sw1-a other_config:rstp-priority=0
```

- Проверить заданный приоритет можно с помощью команды **ovs-appctl rstp/show**

```
[root@sw1-a ~]# ovs-appctl rstp/show
---- sw1-a ----
Root ID:
  stp-priority      0
  stp-system-id     bc:24:11:11:4c:4a
  stp-hello-time    2s
  stp-max-age       20s
  stp-fwd-delay     15s
  This bridge is the root

Bridge ID:
  stp-priority      0
  stp-system-id     bc:24:11:11:4c:4a
  stp-hello-time    2s
  stp-max-age       20s
  stp-fwd-delay     15s

Interface  Role          State          Cost      Pri.Nbr
-----
ens19      Designated Forwarding 2000      128.4
ens20      Designated Forwarding 2000      128.1
ens22      Designated Forwarding 2000      128.2
ens21      Designated Forwarding 2000      128.3

[root@sw1-a ~]#
```

- Проверить заданный протокол и приоритет можно с помощью команды **ovs-vsctl list bridge**:

```
[root@sw1-a ~]# ovs-vsctl list bridge
_uuid      : caa77584-1e9b-4fe9-8b62-f7c383f45010
auto_attach : []
controller : []
datapath_id : "0000bc2411114c4a"
datapath_type : ""
datapath_version : "<unknown>"
external_ids : {}
fail_mode   : []
flood_vlans : []
flow_tables : {}
ipfix       : []
mcast_snooping_enable: false
mirrors     : []
name        : sw1-a
netflow     : []
other_config : {rstp-priority="0", stp-priority="0",
ports      : [6d31f883-3097-49af-9c46-13e6a1f5e7ed,
-619310b3d52b, e0675f57-022e-4143-87c6-d06016585df0]
protocols   : []
rstp_enable : true
rstp_status : {rstp_bridge_id="0.000.bc2411114c4a",
rstp_root_id="0.000.bc2411114c4a", rstp_root_path_cost="0"}
sflow       : []
status      : {}
stp_enable  : false

[root@sw1-a ~]#
```

- Сетевая подсистема **etcdnet** будет взаимодействовать с **openvswitch**, для того чтобы корректно можно было назначить IP-адрес на интерфейс управления
 - создаём каталог для management интерфейса с именем **mgmt**:

```
mkdir /etc/net/ifaces/mgmt
```

- Описываем файл **options** для создания management интерфейса с именем **mgmt**:

```
vim /etc/net/ifaces/mgmt/options
```

- где:
 - **TYPE** - тип интерфейса (**internal**);
 - **BOOTPROTO** - определяет как будут назначаться сетевые параметры (статически);
 - **CONFIG_IPV4** - определяет использовать конфигурацию протокола IPv4 или нет;
 - **BRIDGE** - определяет к какому мосту необходимо добавить данный интерфейс;
 - **VID** - определяет принадлежность интерфейса к VLAN;

```
TYPE=ovsport
BOOTPROTO=static
CONFIG_IPV4=yes
BRIDGE=sw1-a
VID=300
```

- Назначаем IP-адрес и шлюз на созданный интерфейс **mgmt**:

```
echo "172.20.30.1/24" > /etc/net/ifaces/mgmt/ipv4address
```

```
echo "default via 172.20.30.254" > /etc/net/ifaces/mgmt/ipv4route
```

- Для применения настроек, необходимо перезагрузить службу **network**:

```
systemctl restart network
```

- Проверить назначенный IP-адрес можно командой **ip a**:

```
[root@sw1-a ~]# ip -c -br -4 a
lo                UNKNOWN    127.0.0.1/8
mgmt              UNKNOWN    172.20.30.1/24
[root@sw1-a ~]# ip -c r
default via 172.20.30.254 dev mgmt
```

- Проверить назначенный IP-адрес шлюза по умолчанию можно командой **ip r**:

```
[root@sw1-a ~]# ip -c r
default via 172.20.30.254 dev mgmt
172.20.30.0/24 dev mgmt proto kernel scope link src 172.20.30.1
[root@sw1-a ~]#
```

- Также стоит с помощью команды **ovs-vsctl show** проверить, что данный интерфейс добавился в коммутатор **sw1-a**:

```
[root@sw1-a ~]# ovs-vsctl show
245b4b82-d2c3-4e24-a664-0df54cf70786
    Bridge sw1-a
        Port ens19
            trunks: [100, 200, 300]
            Interface ens19
        Port sw1-a
            Interface sw1-a
            type: internal
        Port ens21
            trunks: [100, 200, 300]
            Interface ens21
        Port mgmt
            tag: 300
            Interface mgmt
            type: internal
        Port ens22
            trunks: [100, 200, 300]
            Interface ens22
        Port ens20
            tag: 100
            Interface ens20
    ovs_version: "3.3.2"
[root@sw1-a ~]#
```

- Помимо того, что интерфейс **mgmt** является портом доступа (access) необходимо использовать NativeVLAN:

```
ovs-vsctl set port mgmt vlan_mode=native-untagged
```

- Проверить можно с помощью команды **ovs-vsctl list port mgmt**:

```
[root@sw1-a ~]# ovs-vsctl list port mgmt
    _uuid          : d44ab8d5-5565-4894-8510-64bf7bc85bad
    bond_active_slave : []
    bond_downdelay   : 0
    bond_fake_iface  : false
    bond_mode        : []
    bond_updelay     : 0
    culans           : []
    external_ids     : {}
    fake_bridge      : false
    interfaces       : [f8e44b2f-9e4d-410d-8d80-38de806f9df1]
    lacp             : []
    mac              : []
    name             : mgmt
    other_config     : {}
    protected        : false
    qos              : []
    rstp_statistics  : {}
    rstp_status      : {}
    statistics       : {}
    status           : {}
    tag              : 300
    trunks           : []
    vlan_mode        : native-untagged
[root@sw1-a ~]#
```

sw2-a (alt-server):

- Проверяем интерфейсы и определяемся какой к кому направлен:
 - таким образом, имеем:
 - **ens19** - интерфейс в сторону **sw-1**;
 - **ens20** - интерфейс в сторону **sw-1**;
 - **ens21** - интерфейс в сторону **cli2-a**;
 - **ens22** - интерфейс в сторону **cli1-a**;

```
[root@sw2-a ~]# ip -c -br a
lo          UNKNOWN
ens19       UP
ens20       DOWN
ens21       DOWN
ens22       DOWN
[root@sw2-a ~]#
```


- Поднимаем физические интерфейсы, создавая директорию для каждого интерфейса в `/etc/net/ifaces` и описывая файл **options**:
 - если в файле **options** для порты **ens19** есть параметры **TYPE=eth** и **BOOTPROTO=static**,
 - то можно рекурсивно выполнить копирование **ens19** во все необходимые каталоги

```
cp -r /etc/net/ifaces/ens19 /etc/net/ifaces/ens20
```

```
cp -r /etc/net/ifaces/ens19 /etc/net/ifaces/ens21
```

```
cp -r /etc/net/ifaces/ens19 /etc/net/ifaces/ens22
```

- Перезагружаем службу **network**:

```
systemctl restart network
```

- Проверить, что интерфейсы перешли из статуса **DOWN** в статус **UP**:

```
[root@sw2-a ~]# ip -c -br a
lo                UNKNOWN
ens19             UP
ens20             UP
ens21             UP
ens22             UP
[root@sw2-a ~]#
```

Настройка коммутации:

- Создадим коммутатор с именем **sw2-a**:

```
ovs-vsctl add-br sw2-a
```

- Проверить создание коммутатора можно с помощью команды **ovs-vsctl show**:

```
[root@sw2-a ~]# ovs-vsctl show
3c07bdb7-8007-4fa1-b4dd-3df4face806e
  Bridge sw2-a
    Port sw2-a
      Interface sw2-a
        type: internal
    ovs_version: "3.3.2"
[root@sw2-a ~]#
```

- Добавим интерфейс, направленный в сторону **cli1-a** (ens22) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 200**:

```
ovs-vsctl add-port sw2-a ens22 tag=200
```

- Добавим интерфейс, направленный в сторону **cli2-a** (ens21) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 200**:

```
ovs-vsctl add-port sw2-a ens21 tag=200
```

- Интерфейсы в стороны **sw1-a** (ens19, ens20) добавляем в созданный коммутатор, но настраиваем как магистральный (trunk) порт:
 - также разрешаем пропуск только требуемых VLAN (100,200 и 300)

```
ovs-vsctl add-port sw1-a ens19 trunk=100,200,300
```

```
ovs-vsctl add-port sw1-a ens20 trunk=100,200,300
```

- Проверить добавление портов в коммутатор можно с помощью команды **ovs-vsctl show**:

```
[root@sw2-a ~]# ovs-vsctl show
3c07bdb7-8007-4fa1-b4dd-3df4faee806e
Bridge sw2-a
  Port ens22
    tag: 200
    Interface ens22
  Port ens20
    trunks: [100, 200, 300]
    Interface ens20
  Port sw2-a
    Interface sw2-a
      type: internal
  Port ens21
    tag: 200
    Interface ens21
  Port ens19
    trunks: [100, 200, 300]
    Interface ens19
  ovs_version: "3.3.2"
[root@sw2-a ~]#
```

- Включаем модуль ядра отвечающий за тегированный трафик (**802.1Q**):

```
modprobe 8021q
```

```
echo "8021q" | tee -a /etc/modules
```

- Запускаем процесс STP на коммутаторе:
 - в режиме **802.1w** (rstp)

```
ovs-vsctl set bridge sw2-a rstp_enable=true
```

```
ovs-vsctl set bridge sw2-a other_config:stp-protocol=rstp
```

- Проверить можно с помощью команды **ovs-appctl rstp/show**:

```
[root@sw2-a ~]# ovs-appctl rstp/show
---- sw2-a ----
Root ID:
  stp-priority      0
  stp-system-id     bc:24:11:11:4c:4a
  stp-hello-time    2s
  stp-max-age       20s
  stp-fwd-delay     15s
  root-port         ens20
  root-path-cost    2000
Bridge ID:
  stp-priority      32768
  stp-system-id     bc:24:11:5d:92:6d
  stp-hello-time    2s
  stp-max-age       20s
  stp-fwd-delay     15s

  Interface  Role      State      Cost    Pri.Nbr
  -----
  ens21      Designated Forwarding 2000    128.4
  ens20      Root      Forwarding 2000    128.1
  ens22      Designated Forwarding 2000    128.2
  ens19      Alternate Discarding 2000    128.3
[root@sw2-a ~]#
```

- Проверить заданный протокол и приоритет можно с помощью команды **ovs-vsctl list bridge**:

```
[root@sw2-a ~]# ovs-vsctl list bridge
_uuid                : aec1d564-ade1-461e-97f8-2e6635225138
auto_attach          : []
controller           : []
datapath_id          : "0000bc24115d926d"
datapath_type        : ""
datapath_version     : "<unknown>"
external_ids         : {}
fail_mode            : []
flood_vlans          : []
flow_tables          : {}
ipfix                : []
mcast_snooping_enable : false
mirrors              : []
name                 : sw2-a
netflow              : []
other_config         : {stp-protocol=rstp}
ports                : [5eb84cec-c8fe-42ef-90d4-67ec06e20403, 61b6
-001cc189acf3, eed4acda-c2b4-4828-85ed-68c38e27cec8]
protocols            : []
rstp_enable          : true
rstp_status          : {rstp_bridge_id="8.000.bc24115d926d", rstp_
rstp_root_id="8.000.bc241114c4a", rstp_root_path_cost="2000"}
sflow                : []
status               : {}
stp_enable           : false
[root@sw2-a ~]#
```

- Сетевая подсистема **etctnet** будет взаимодействовать с **openvswitch**, для того чтобы корректно можно было назначить IP-адрес на интерфейс управления
 - создаём каталог для management интерфейса с именем **mgmt**:

```
mkdir /etc/net/ifaces/mgmt
```

- Описываем файл **options** для создания management интерфейса с именем **mgmt**:

```
vim /etc/net/ifaces/mgmt/options
```

- где:
 - **TYPE** - тип интерфейса (**internal**);
 - **BOOTPROTO** - определяет как будут назначаться сетевые параметры (статически);
 - **CONFIG_IPV4** - определяет использовать конфигурацию протокола IPv4 или нет;
 - **BRIDGE** - определяет к какому мосту необходимо добавить данный интерфейс;
 - **VID** - определяет принадлежность интерфейса к VLAN;

```
TYPE=ovsport
BOOTPROTO=static
CONFIG_IPV4=yes
BRIDGE=sw2-a
VID=300
```

- Назначаем IP-адрес и шлюз на созданный интерфейс **mgmt**:

```
echo "172.20.30.2/24" > /etc/net/ifaces/mgmt/ipv4address
```

```
echo "default via 172.20.30.254" > /etc/net/ifaces/mgmt/ipv4route
```

- Для применения настроек, необходимо перезагрузить службу **network**:

```
systemctl restart network
```

- Проверить назначенный IP-адрес можно командой **ip a**:

```
[root@sw2-a ~]# ip -c -br -4 a
lo                UNKNOWN    127.0.0.1/8
mgmt              UNKNOWN    172.20.30.2/24
[root@sw2-a ~]#
```

- Проверить назначенный IP-адрес шлюза по умолчанию можно командой **ip r**:

```
[root@sw2-a ~]# ip -c r
default via 172.20.30.254 dev mgmt
172.20.30.0/24 dev mgmt proto kernel scope link src 172.20.30.2
[root@sw2-a ~]#
```

- Также стоит с помощью команды **ovs-vsctl show** проверить, что данный интерфейс добавился в коммутатор **sw2-a**:

```
[root@sw2-a ~]# ovs-vsctl show
3c07bdb7-8007-4fa1-b4dd-3df4faee806e
  Bridge sw2-a
    Port mgmt
      tag: 300
    Interface mgmt
      type: internal
    Port ens22
      tag: 200
    Interface ens22
    Port ens20
      trunks: [100, 200, 300]
    Interface ens20
    Port sw2-a
      Interface sw2-a
        type: internal
    Port ens21
      tag: 200
    Interface ens21
    Port ens19
      trunks: [100, 200, 300]
    Interface ens19
  ovs_version: "3.3.2"
[root@sw2-a ~]#
```

- Помимо того, что интерфейс **mgmt** является портом доступа (access) необходимо использовать NativeVLAN:

```
ovs-vsctl set port mgmt vlan_mode=native-untagged
```

- Проверить можно с помощью команды **ovs-vsctl list port mgmt**:

```
[root@sw2-a ~]# ovs-vsctl list port mgmt
_uuid                : 3c29bbb7-8223-4892-b8a3-7a3c671eaf30
bond_active_slave    : []
bond_downdelay       : 0
bond_fake_iface      : false
bond_mode            : []
bond_updelay         : 0
culans               : []
external_ids         : {}
fake_bridge          : false
interfaces           : [9150af57-a5e1-4c74-b0e1-358639b9dd3e]
lacp                 : []
mac                 : []
name                 : mgmt
other_config         : {}
protected            : false
qos                  : []
rstp_statistics      : {}
rstp_status          : {}
statistics           : {}
status              : {}
tag                  : 300
trunks               : []
vlan_mode            : native-untagged
[root@sw2-a ~]#
```

- Проверить доступность коммутаторов **sw1-a** и **sw2-a** можно как с маршрутизатора **rtr-a**:

```

rtr-a#ping 172.20.30.1
PING 172.20.30.1 (172.20.30.1) 56(84) bytes of data.
64 bytes from 172.20.30.1: icmp_seq=1 ttl=64 time=211 ms
64 bytes from 172.20.30.1: icmp_seq=2 ttl=64 time=22.4 ms
64 bytes from 172.20.30.1: icmp_seq=3 ttl=64 time=205 ms
64 bytes from 172.20.30.1: icmp_seq=4 ttl=64 time=29.1 ms

--- 172.20.30.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3002ms
rtt min/avg/max/mdev = 22.421/116.800/210.773/91.076 ms
rtr-a#ping 172.20.30.2
PING 172.20.30.2 (172.20.30.2) 56(84) bytes of data.
64 bytes from 172.20.30.2: icmp_seq=1 ttl=64 time=81.4 ms
64 bytes from 172.20.30.2: icmp_seq=2 ttl=64 time=24.0 ms
64 bytes from 172.20.30.2: icmp_seq=3 ttl=64 time=73.7 ms
64 bytes from 172.20.30.2: icmp_seq=4 ttl=64 time=198 ms

--- 172.20.30.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 24.005/94.301/198.166/63.879 ms
rtr-a#

```

- о так и с маршрутизатора **rtr-cod**:

```

rtr-cod#ping 172.20.30.1
PING 172.20.30.1 (172.20.30.1) 56(84) bytes of data.
64 bytes from 172.20.30.1: icmp_seq=1 ttl=64 time=121 ms
64 bytes from 172.20.30.1: icmp_seq=2 ttl=64 time=86.3 ms
64 bytes from 172.20.30.1: icmp_seq=3 ttl=64 time=109 ms
64 bytes from 172.20.30.1: icmp_seq=4 ttl=64 time=184 ms

--- 172.20.30.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 86.327/125.213/184.125/36.243 ms
rtr-cod#ping 172.20.30.2
PING 172.20.30.2 (172.20.30.2) 56(84) bytes of data.
64 bytes from 172.20.30.2: icmp_seq=1 ttl=64 time=197 ms
64 bytes from 172.20.30.2: icmp_seq=2 ttl=64 time=98.9 ms
64 bytes from 172.20.30.2: icmp_seq=3 ttl=64 time=105 ms
64 bytes from 172.20.30.2: icmp_seq=4 ttl=64 time=103 ms

--- 172.20.30.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 98.929/126.174/197.339/41.150 ms
rtr-cod#

```

- Также с коммутаторов **sw1-a** и **sw2-a** должна быть доступна сеть Интернет:

```

[root@sw1-a ~]# ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=48 time=90.7 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=48 time=92.3 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=48 time=88.6 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 88.647/90.537/92.284/1.488 ms
[root@sw1-a ~]#

```

```

[root@sw2-a ~]# ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=48 time=64.9 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=48 time=61.5 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=48 time=60.6 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 60.613/62.352/64.948/1.870 ms
[root@sw2-a ~]#

```

Предыдущий
элемент курса

6. Настройка
динамической
маршрутизации между
офисом «a» и «cod»

Перейти на...

Следующий элемент
курса

8. Настройка доступа к
веб-интерфейсу fw-cod и
лицензирование

Обратная связь

Подпишитесь

Вы используете гостевой
доступ (Вход)
Сводка хранения данных

Тема оформления сайта разработана
conecti.me

8. Настройка доступа к веб-интерфейсу fw-cod и лицензирование

[REG26](#)

- 8. Настройка доступа к веб-интерфейсу fw-cod и лицензирование

Вариант реализации:

fw-cod (ideco):

Проверка наличия временного маршрута по умолчанию:

- Выполнить вход на **fw-cod** из-под ранее созданного пользователя и перейти в консоль:

```
Ideco NGFW 21.5.99
-----
Вход в локальное меню.
Введите логин и нажмите Enter.
# admin
Введите пароль и нажмите Enter.
Введите 'b' и нажмите Enter для возврата.
#
Управление сервером
1. Консоль
2. Физические порты
3. Ethernet-интерфейсы
4. Отключить все статические маршруты и добавить новый
   Выбор типа кластерной сети
   ▶ Включить доступ к веб-интерфейсу из внешней сети
   Включить доступ к серверу по SSH из Интернет
5. Включить доступ к серверу по SSH из локальных сетей
6. Включить режим `Разрешить Интернет всем`
7. Сбросить блокировки по IP
8. Отключить пользовательский межсетевой экран
9. Создать новый бэкап
10. Восстановить из бэкапа
11. Мгновенно восстановить из бэкапа
12. Включить доступ Удаленного Помощника
13. Контакты технической поддержки
14. Восстановиться на предыдущую версию
15. Перезагрузка сервера
16. Отключить сервер
17. Выход
Введите номер пункта и нажмите Enter.
# 1_
```

- Проверить таблицу маршрутов можно с помощью команды **ip r**:

```
[admin@bez-nazvaniya-fddcbdbb-81a5-420d-bece-e00da33673ff ~]# ip -c r
169.254.1.0/29 dev lb_local_in proto kernel scope link src 169.254.1.1
169.254.1.0/29 dev lb_local_out proto kernel scope link src 169.254.1.3
169.254.1.4/30 dev Lvpn0 proto kernel scope link src 169.254.1.6
169.254.1.5 dev Lvpn0 scope link
172.16.1.0/30 dev Leth2 proto kernel scope link src 172.16.1.2
[admin@bez-nazvaniya-fddcbdbb-81a5-420d-bece-e00da33673ff ~]#
```

- Если отсутствует маршрут по умолчанию, его необходимо временно добавить средствами команды **ip route add ...**:


```
ip route add 0.0.0.0/0 via 172.16.1.1
```

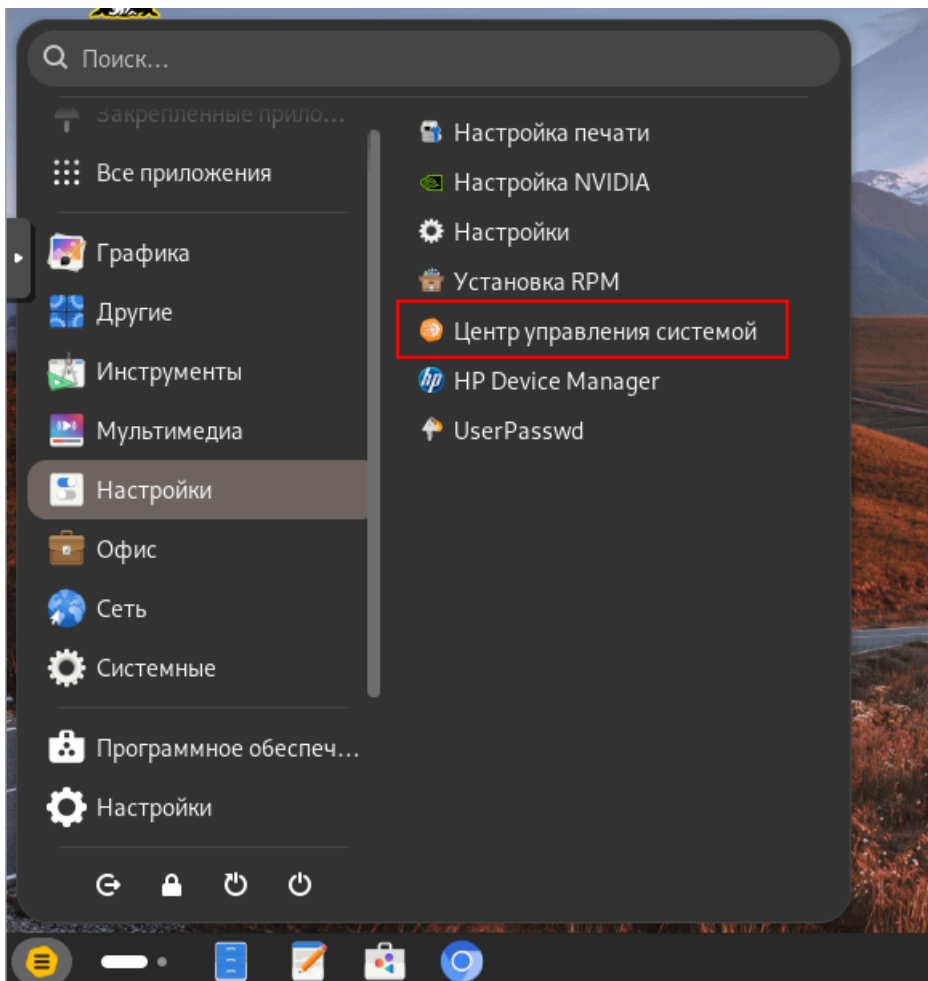
- Повторно проверить таблицу маршрутов:

```
[admin@bez-nazvaniya-fddcbdbb-81a5-420d-bece-e00da33673ff ~]# ip -c r
default via 172.16.1.1 dev Leth2
169.254.1.0/29 dev lb_local_in proto kernel scope link src 169.254.1.1
169.254.1.0/29 dev lb_local_out proto kernel scope link src 169.254.1.3
169.254.1.4/30 dev Lvpn0 proto kernel scope link src 169.254.1.6
169.254.1.5 dev Lvpn0 scope link
172.16.1.0/30 dev Leth2 proto kernel scope link src 172.16.1.2
[admin@bez-nazvaniya-fddcbdbb-81a5-420d-bece-e00da33673ff ~]# _
```

cli1-a (alt-workstation):

Базовая настройка устройства:

- Перейдём в Центр Управления Системой (ЦУС/acc):



- В ЦУС перейдём в раздел Ethernet-интерфейсы:



Система

Информация об установленной системе и её настройка

[Дата и время](#) [Информация о дистрибутиве](#) [Лицензионное соглашен](#)
[Настройка нескольких рабочих мест](#) [Настройка ограничений](#) [Обнов](#)
[Обновление ядра](#) [Сетевые каталоги](#) [Системные журналы](#) [Системн](#)
[Системные службы](#) [Управление ключами SSL](#)



Пользователи

Управление пользователями системы

[Администратор системы](#) [Аутентификация](#) [Использование диска](#) [Л](#)
[Локальные учётные записи](#)



Брандмауэр

Брандмауэр

[Внешние сети](#) [Перенаправление портов](#) [Список блокируемых хостов](#)



Сеть

Настройка подключения к сети

[Ethernet-интерфейсы](#) [OpenVPN-соединения](#) [PPPoE-соединения](#) [P](#)
[Прокси-сервер](#)



Графический интерфейс

Настройка устройств ввода-вывода

[Дисплей](#) [Клавиатура](#)

- Назначим имя на устройство в соответствие с топологией:

Ethernet-интерфейсы

Имя компьютера:

cli1-a.office.ssa2026.region

Интерфейсы

ens19

Сетевая карта:

провод подсоединён

MAC: bc:24:11:4e:3c:26

Интерфейс ВКЛЮЧЕН

Версия протокола IP: IPv4 ☒ Включить

Конфигурация:

IP-адреса:

Добавить ↑ IP:

Шлюз по умолчанию:

DNS-серверы:

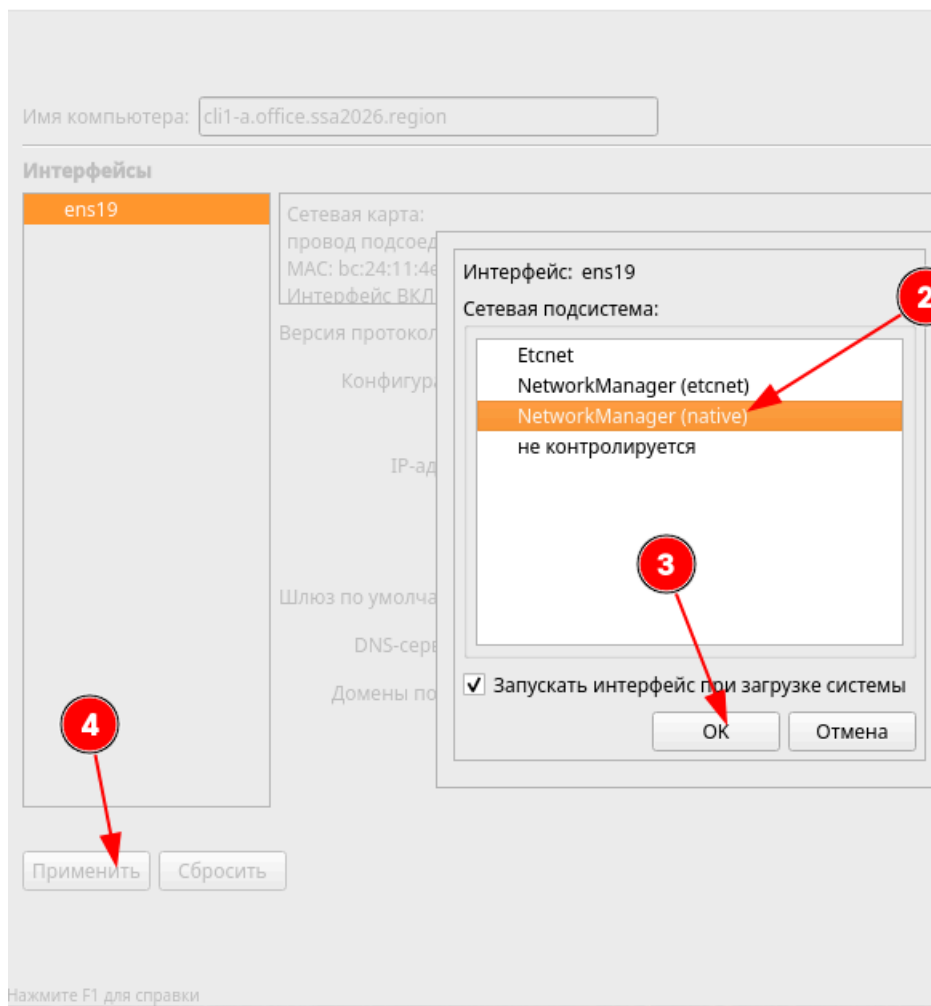
Домены поиска:

(несколько значений записываются через пр

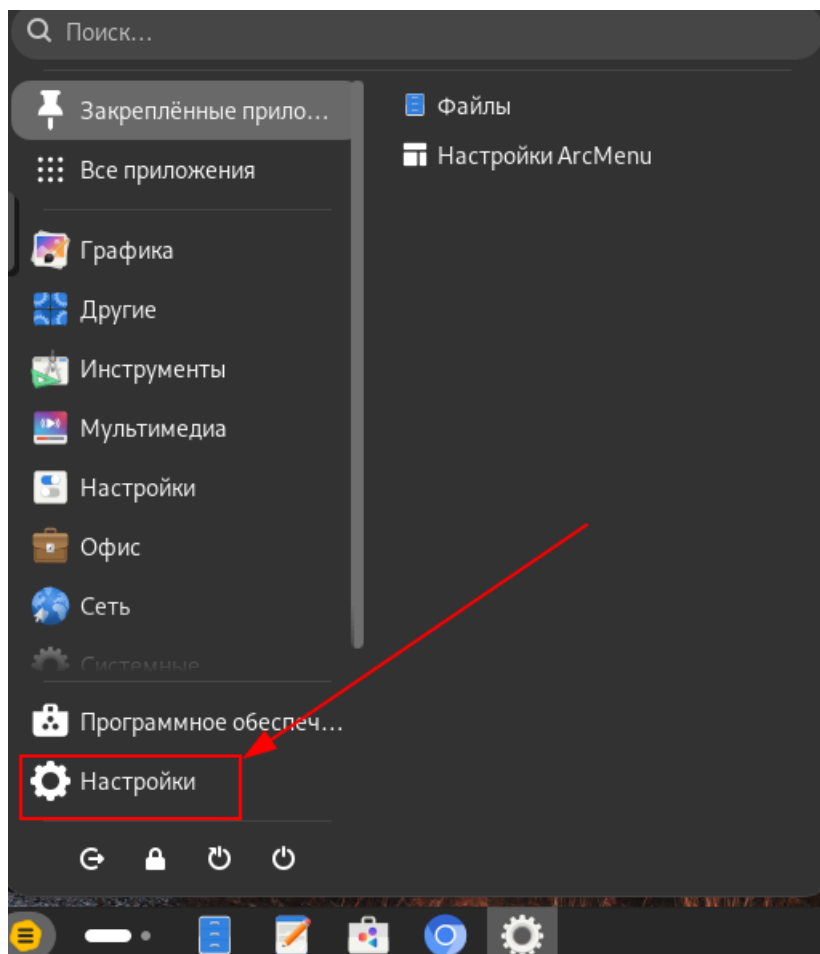
Применить

Сбросить

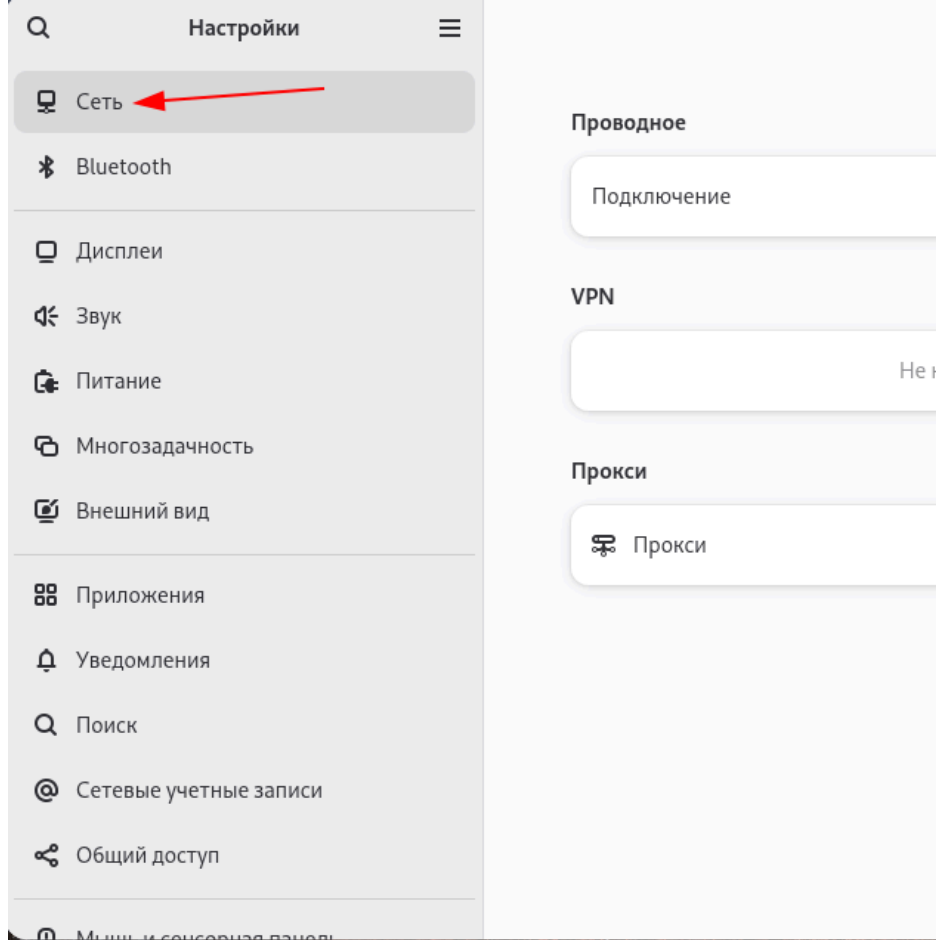
- В качестве режима работы сетевой подсистемы выберем **NetworkManager (native)**:



- Перейдём в обычный **Настройки** для назначения сетевых параметров:



- В разделе **Сеть** перейдём к настройке сетевого подключения:



- Задаём **Вручную** и нажимаем **Применить**:
 - **Адрес**
 - **Маску сети**
 - **Шлюз**
 - **DNS** (в качестве DNS-сервера указываем IP-адрес **dc-a**)

Отменить

Проводное подключение

ПодробнееИдентификацияIPv4IPv6Безопасность

Метод IPv4

☐ Автоматический (DHCP)

☒ Вручную

☐ Только для...

☐ Выключить

☐ Общий доступ другим компьютерам

Адреса

Адрес	Маска сети	Шлюз
172.20.20.1	255.255.255.0	172.20.20.254

DNS

172.20.10.10

Отделяйте IP-адреса запятыми

Маршруты

Адрес	Маска сети	Шлюз	

☐ Использовать это подключение только для ресурсов в этой сети

- Проверить можно на вкладке **Подробнее**:

Отменить

Проводное подключение

ПодробнееИдентификацияIPv4IPv6Безопасность

Адрес IPv4

172.20.20.1

Адрес IPv6

fe80::9eb7:43db:b272:14f3

Аппаратный адрес

BC:24:11:4E:3C:26

Маршрут по умолчанию

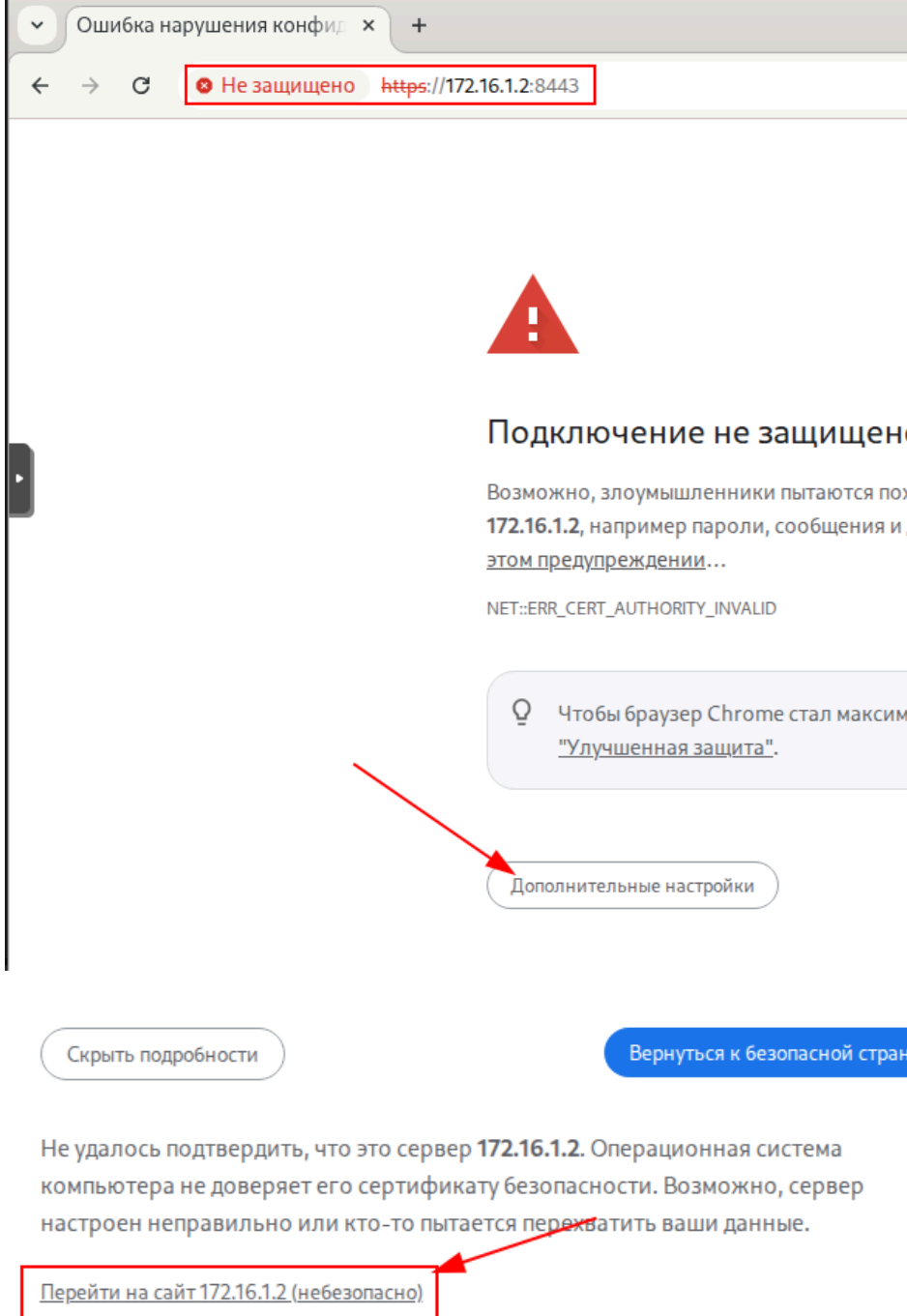
172.20.20.254

DNS

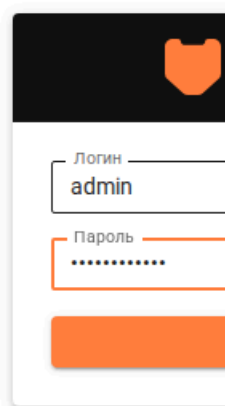
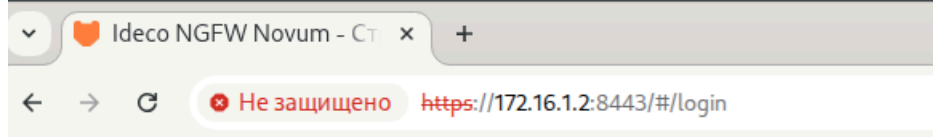
172.20.10.10

Доступ к веб-интерфейсу управления fw-cod:

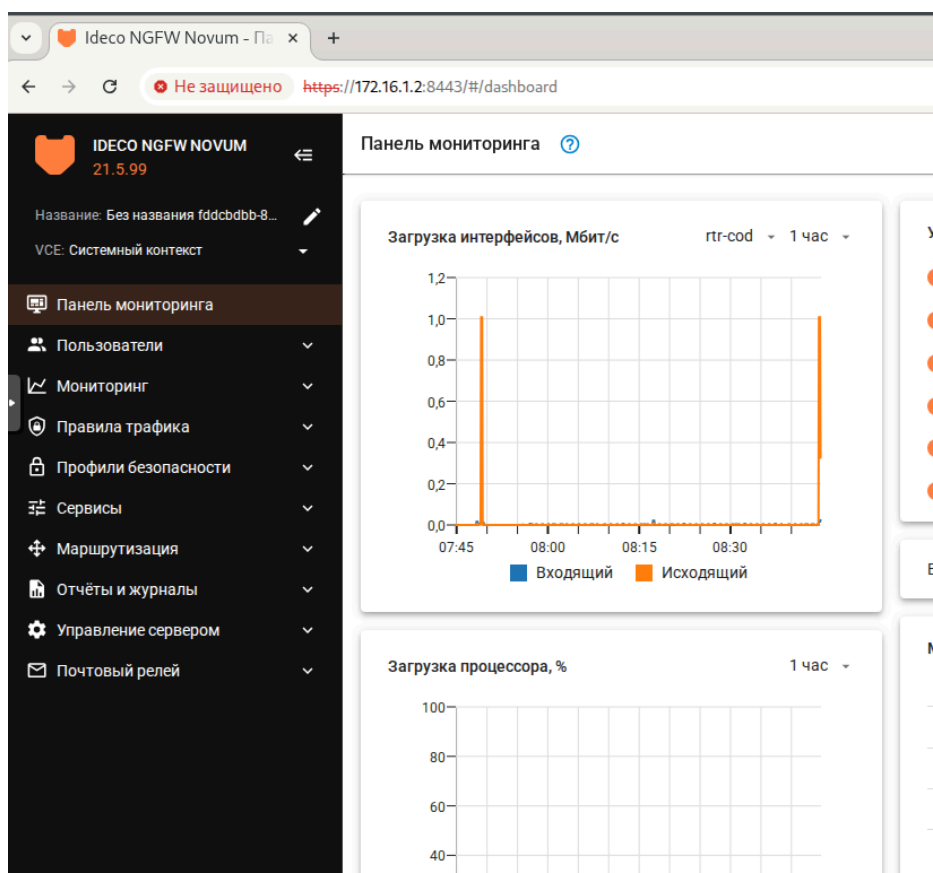
- Открываем браузер и переходим по <https://172.16.1.2:8443> (IP-адрес fw-cod):



- Выполняем вход в веб-интерфейс управления **fw-cod**:

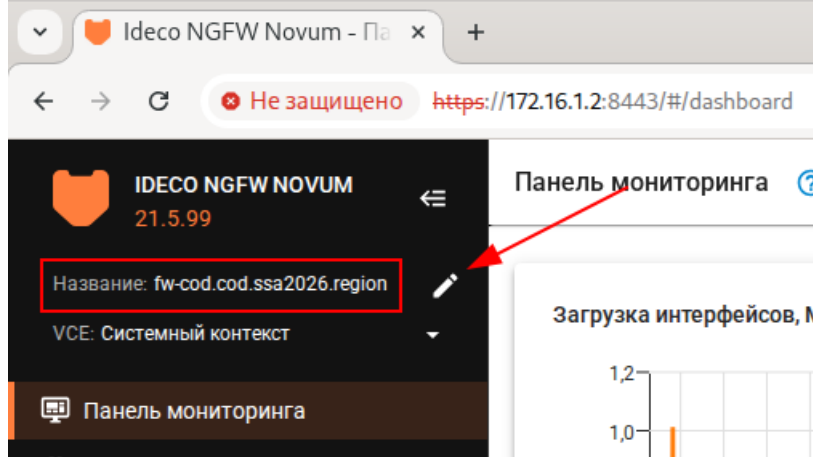


- Результат успешного входа:

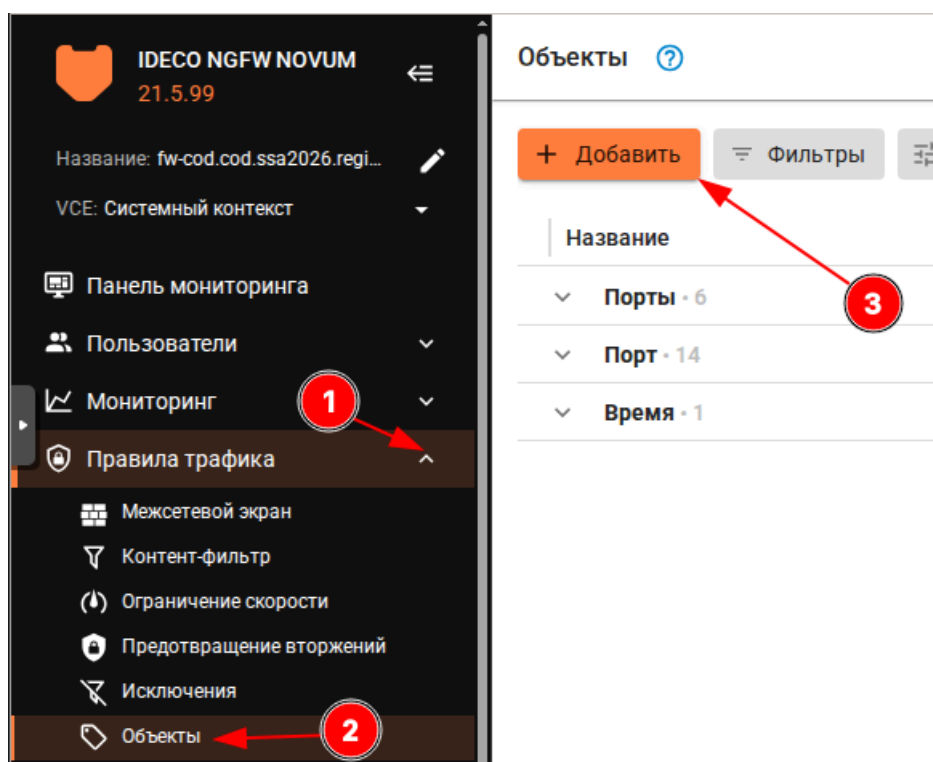


Базовая настройка fw-cod через веб-интерфейс управления:

- Зададим имя устройству в соответствии с топологией:



- Зададим адрес маршрута по умолчанию на постоянной основе, т.к. для первого доступа задавался временный:
 - для начала создадим **Объект** с типом **IP-адрес**, где и укажем значение, которое в дальнейшем будет использовано в качестве **шлюза**:



Добавление объекта

Тип

IP-адрес

Название

gateway

Значение

172.16.1.1

Комментарий

Добавить

Отмена

- Результат успешного добавления **Объекта** с типом **IP-адрес**:

+ Добавить		Фильтры	Отображение	Поиск
Название	Значение			
^ IP-адрес · 1				
IP gateway	172.16.1.1			
▼ Порты · 6				
▼ Порт · 14				
▼ Время · 1				

- после чего можем создать статический маршрут:

The screenshot shows the IDECO NGFW NOVUM interface. On the left sidebar, the 'Маршрутизация' (Routing) menu item is selected, indicated by a red circle with the number 1. Below it, the 'Статическая' (Static) option is highlighted, indicated by a red circle with the number 2. The main panel displays the 'Статическая' configuration screen, which includes a '+ Добавить' (Add) button highlighted by a red arrow and a red circle with the number 3. The interface also shows a table with columns 'Название' (Name) and 'Назначение' (Destination).

- Заполняем форму, указав:
 - произвольное имя в поле **Название**;
 - выбрав в качестве **Назначения** - **Любой**;
 - выбрав в качестве **Интерфейса** - единственный (на текущий момент) созданный интерфейс в сторону **rtr-cod**;
 - выбрав в поле **Шлюз** - ранее созданный объект с именем **gateway** (в значение этого объекта - IP-адрес шлюза по умолчанию, т.е. **rtr-cod**)

Добавление маршрута

Название

Поле необязательное

Назначение

Интерфейс

Шлюз

Приоритет


Поле необязательное. Укажите шлюз, если выбран интерфейс с режимом статической IP-конфигурации.

Введите число от 1 до 255. Чем меньше приоритет, тем приоритетнее маршрут

Дополнительно


Комментарий

0/256

Добавить

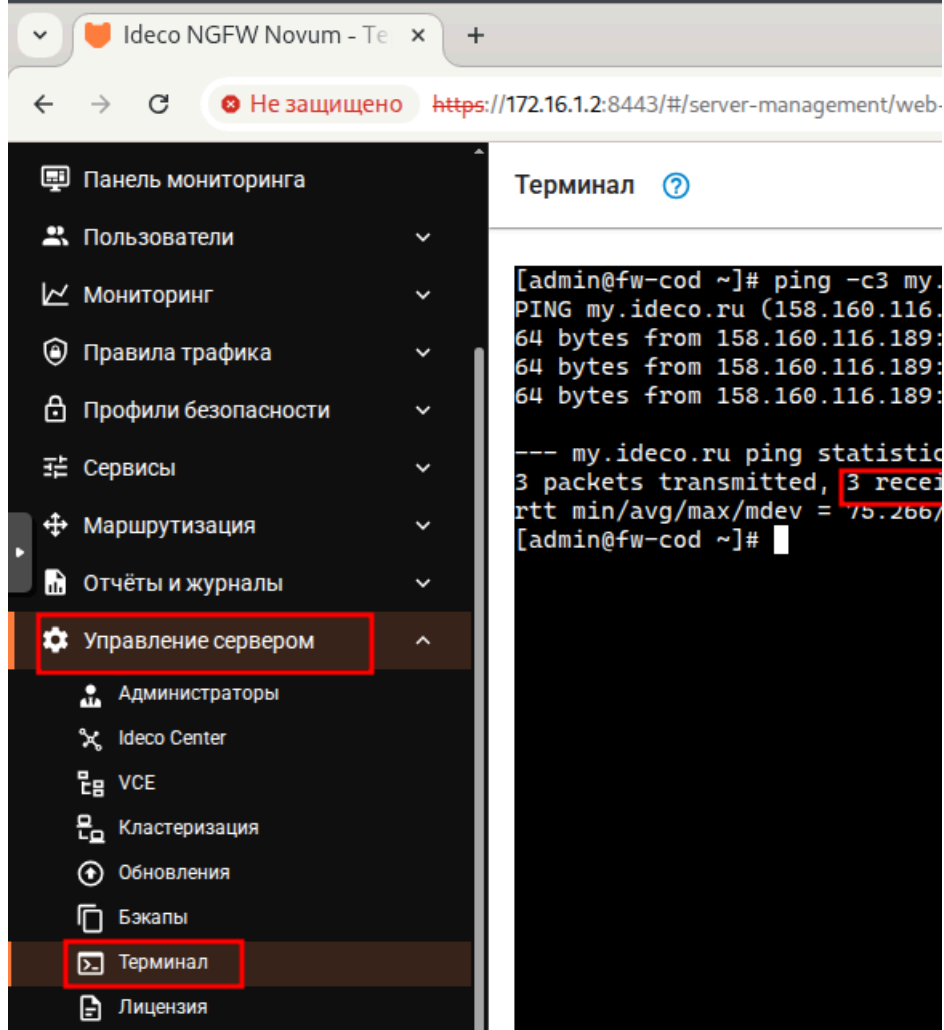
Отмена

- Результат успешного добавления статического маршрута:

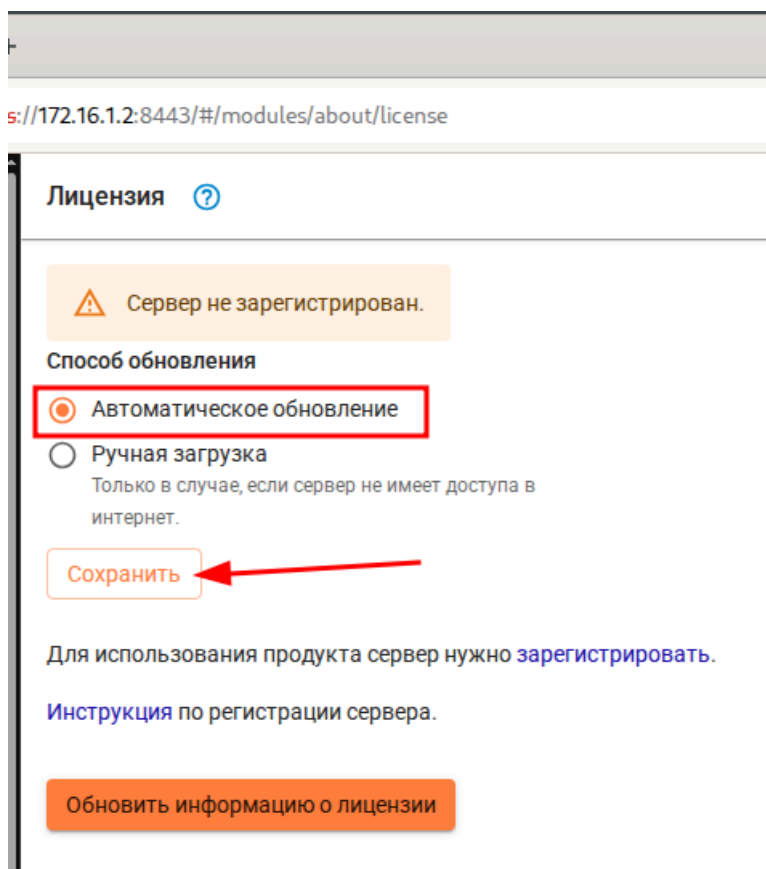
Название	Назначение	Интерфейс	Шлюз	Приоритет
default-gateway	* Любой	 rtr-cod	IP 172.16.1.1	1

Получение лицензии на fw-cod через веб-интерфейс управления:

- Для получения лицензии у fw-cod должен быть доступ в сеть Интернет, а именно до **my.ideco.ru**:



- Перейдите в **Управление сервером** → **Лицензия**, выберите **Автоматическое обновление** в качестве способа обновления, нажмите **Сохранить**:



- Перейдите в **MY.IDECO**, нажав **Зарегистрировать**:

⚠ Сервер не зарегистрирован.

Способ обновления

- ☒ Автоматическое обновление
- ☐ Ручная загрузка
Только в случае, если сервер не имеет доступа в интернет.

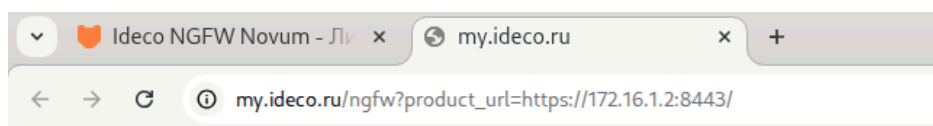
Сохранить

Для использования продукта сервер нужно **зарегистрировать**.

[Инструкция](#) по регистрации сервера.

Обновить информацию о лицензии

- Для того, чтобы имя **my.ideco.ru** с виртуальной машины **cli1-a** корректно было преобразовано DNS-сервером:



Не удается получить доступ к серверу

Не удалось найти IP-адрес сервера **my.i**

Попробуйте сделать следующее:

[Проверьте настройки прокси-сервера,](#)

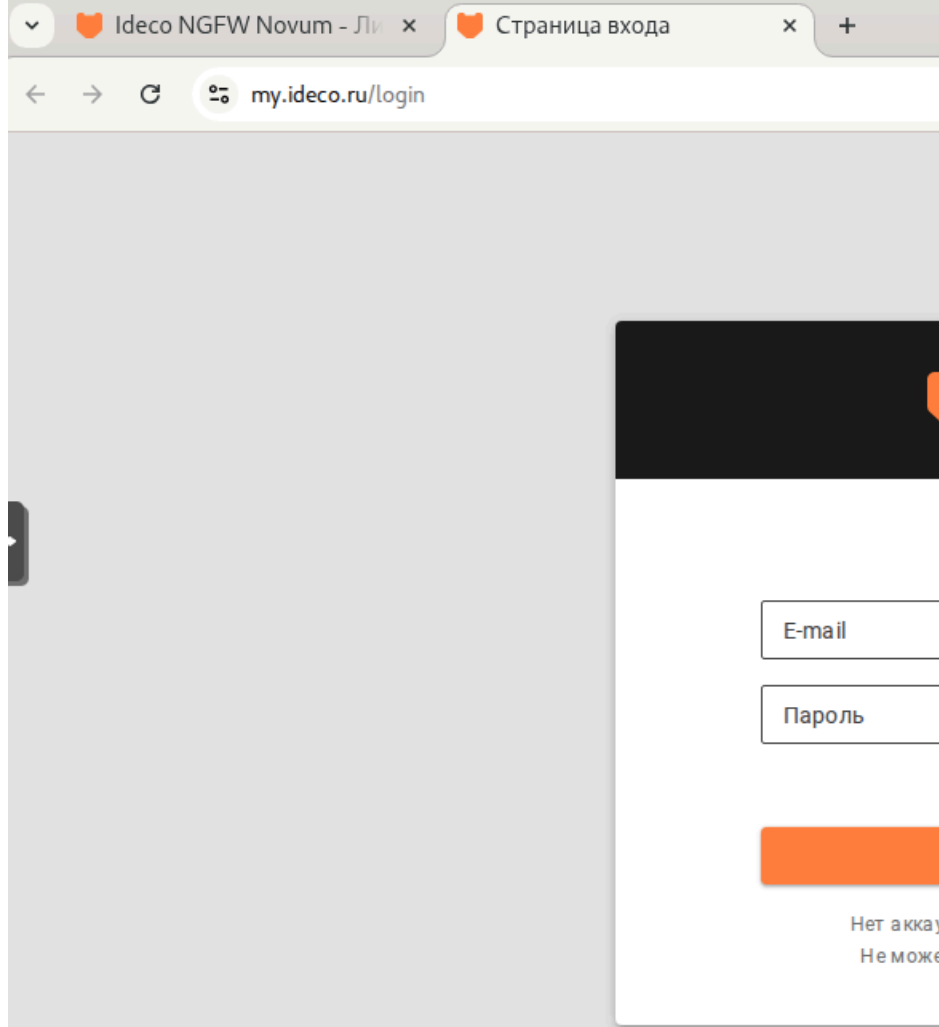
DNS_PROBE_FINISHED_BAD_CONFIG

Сведения

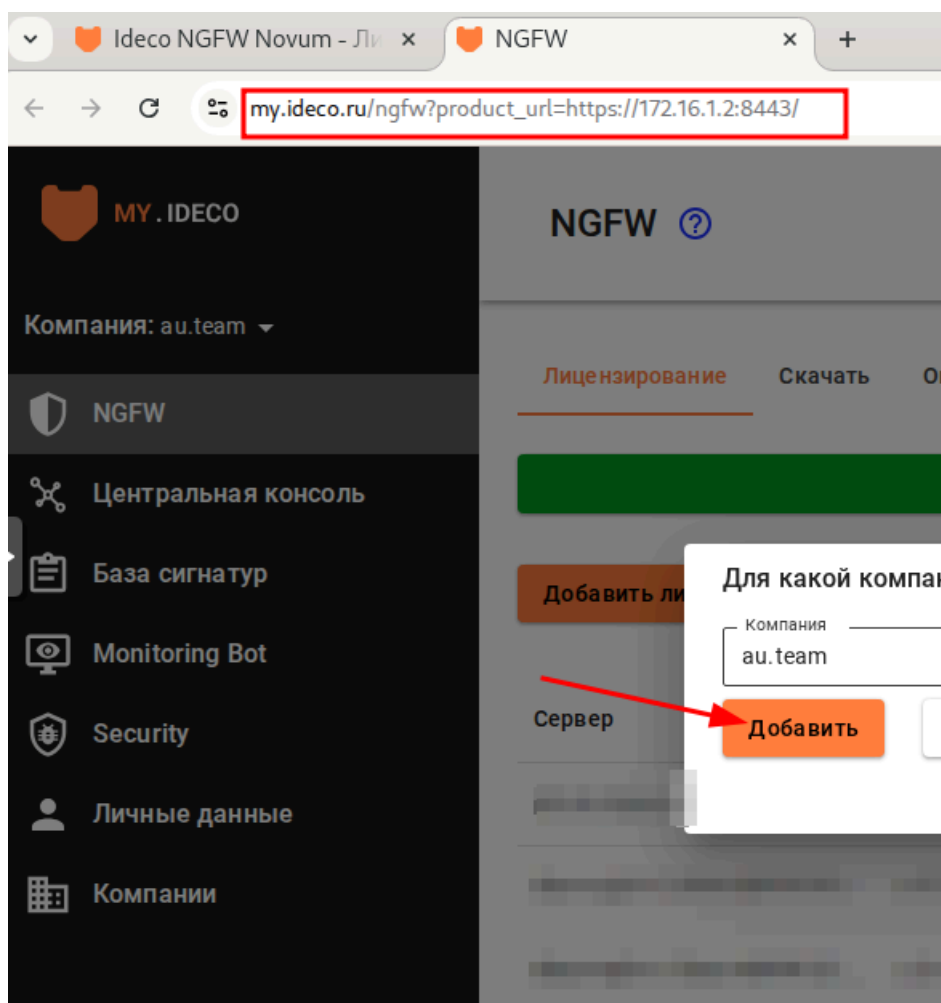
- необходимо временно назначить в качестве DNS-сервера, какой-либо публичный DNS, т.к. на **dc-a** пока DNS не реализован:

```
root@cli1-a: /root
[user@cli1-a ~]$ su -
Password:
[root@cli1-a ~]# echo "nameserver 77.88.8.8" > /etc/resolv.conf
[root@cli1-a ~]#
```

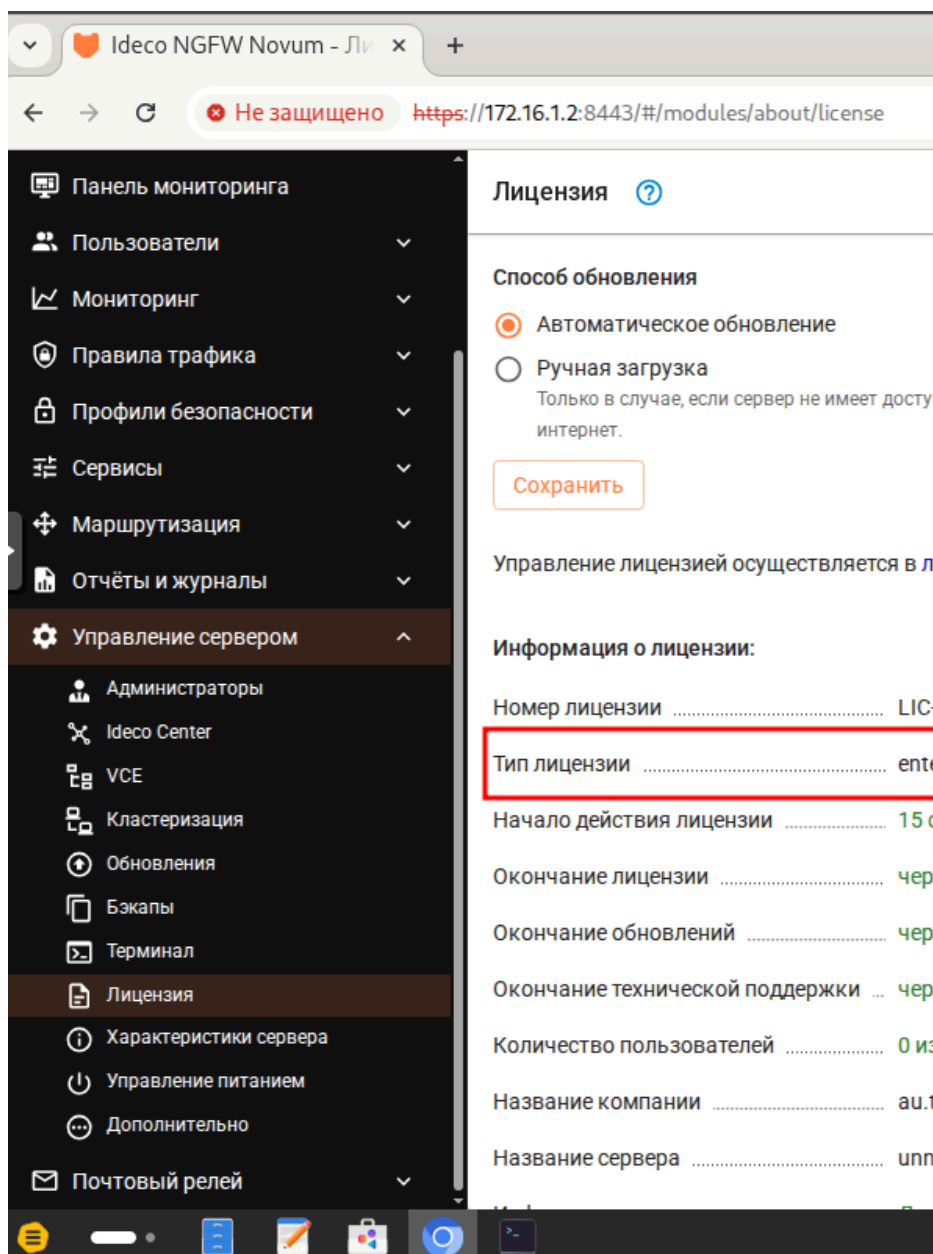
- Результат, выполните вход в личный кабинет MY.IDECO для дальнейшего лицензирования:



- В открывшемся окне выберите компанию и нажмите **Добавить**
 - после добавления нажмите **Обновить** информацию о лицензии для проверки состояния лицензии:



- Результат успешного лиценсирования fw-cod:



Последнее изменение: четверг, 13 ноября 2025, 12:12

Предыдущий
элемент курса

7. Настройка
коммутации между sw1-
а и sw2-а

Перейти на...

Следующий элемент
курса

9. Создание
интерфейсов типа VLAN
на fw-cod для
маршрутизации между
VLAN

Обратная связь

Подпишитесь

Вы используете гостевой
доступ (Вход)

Сводка хранения данных

Тема оформления сайта разработана

9. Создание интерфейсов типа VLAN на fw-cod для маршрутизации между VLAN

[REG26](#)

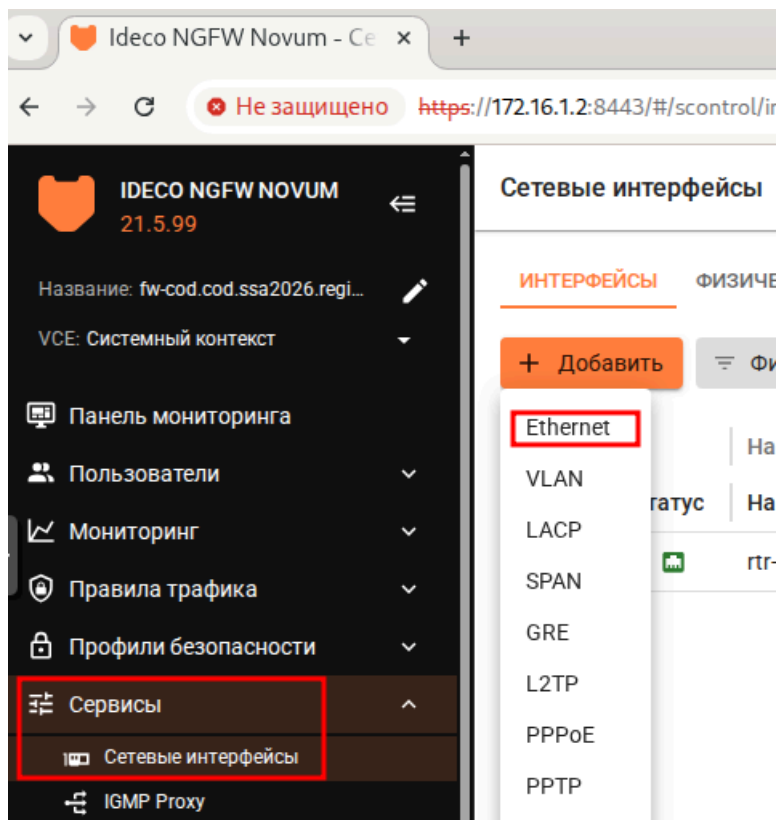
- 9. Создание интерфейсов
- > типа VLAN на fw-cod для маршрутизации между VLAN

Вариант реализации:

cli1-a (alt-workstation):

Создание интерфейсов типа VLAN на fw-cod для маршрутизации между VLAN

- Открываем браузер и переходим по <https://172.16.1.2:8443> (IP-адрес fw-cod), выполняем вход в веб-интерфейс управления fw-cod
- Перейдём в Сервисы -> Сетевые интерфейсы -> Добавить -> VLAN:



- Заполняем форму **Добавление Ethernet-интерфейса**:
 - задаём произвольное имя;
 - роль **LAN**;
 - физический порт, направленный в сторону **fw1-cod**;
 - режим - **Без конфигурации**, т.к. IP-адреса будут назначаться на интерфейсы **VLAN**, созданные на базе данного интерфейса

Добавление Ethernet-интерфейса

Название
fw1-cod

Настройки

Роль
LAN

Зона

Поле необязательное

VCE
Системный контекст

Виртуальный контекст (VCE), в котором будет использоваться интерфейс

Физический порт
eth1_da

IP-конфигурация

Режим
Без конфигурации

- Нажимаем **Добавить**, результат добавления интерфейса:

Сетевые интерфейсы ?

ИНТЕРФЕЙСЫ

ФИЗИЧЕСКИЕ ПОРТЫ

+ Добавить

Фильтры

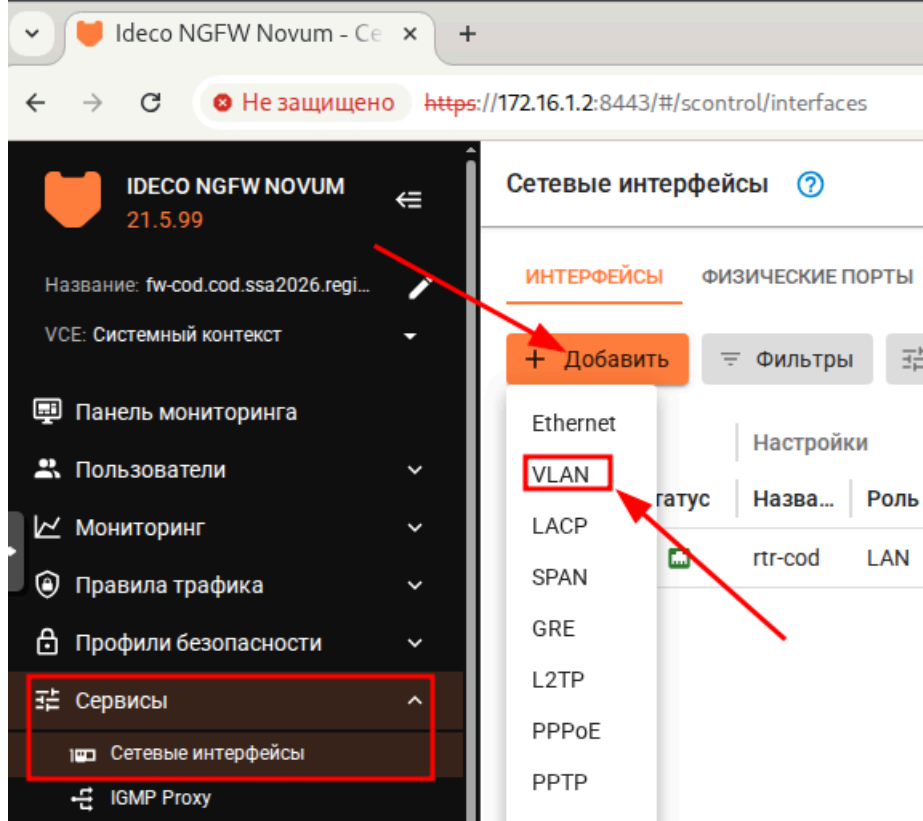
Отображение

Поиск

Настройки

Тип	Статус	Назва...	Роль	Зона	VCE	Интерфейс/порт	Те
Ethern...		rtr-cod	LAN	—	● Сис...	eth0_01	—
Ethern...		fw1-cod	LAN	—	● Сис...	eth1_da	—

- Создаём интерфейсы типа **VLAN**:



- Заполняем форму **Добавление VLAN-интерфейса**:
 - задаём произвольное имя;
 - выбираем ранее созданный интерфейс в сторону **sw1-cod**;
 - указываем в качестве тега **vlan 100**;
 - задаём статическую конфигурацию IP-адреса и нажимаем добавить

Добавление VLAN-интерфейса

Название

Настройки

Роль

Зона

Поле необязательное

VCE

Виртуальный контекст (VCE), в котором будет использоваться интерфейс

Интерфейс

На выбранном интерфейсе создаётся VLAN

Тег VLAN

Целое число от 1 до 4094

IP-конфигурация

Режим

Статический

IP-адрес/маска

192.168.10.254/24

+ Добавить IP-адрес с маской

i Для доступа к сети интернет настройте Балансировку и резервирование.

Дополнительно

Индекс интерфейса для Netflow

0

Целое число от 0 до 65 535

Комментарий

0/256

Добавить

Отмена

- Результат добавления интерфейса типа **VLAN**:

Тип	Статус	Название	Интерфейс/порт	Ter VLAN
Ethernet		rtr-cod	eth0_01	—
Ethernet		fw1-cod	eth1_da	—
VLAN		srv-cod	fw1-cod	100

- Аналогичным образом необходимо создать интерфейсы типа **VLAN** для:
 - vlan300
 - vlan400
 - vlan500
 - для **vlan200** создавать интерфейс нет необходимости, т.к. по условиям задания трафик данного vlan-а не должен маршрутизироваться

Тип	Статус	Название	Интерфейс/порт	Ter VLAN
Ethernet		rtr-cod	eth0_01	—
Ethernet		fw1-cod	eth1_da	—
VLAN		srv-cod	fw1-cod	100
VLAN		mgmt-cod	fw1-cod	300
VLAN		cli	fw1-cod	400
VLAN		voip	fw1-cod	500

- Доступ в сеть Интернет можно проверить с виртуальной машины **sw1-cod** используя команды временного назначения IP-адресов и шлюза:
 - назначив средствами **iproute2** временно на интерфейс, смотрящий в сторону **fw-cod (ens19)**,
 - создав тегированный подинтерфейс с IP-адресом из подсети для **vlan300**

```
ip link add link ens19 name ens19.300 type vlan id 300
ip link set dev ens19 up
ip link set dev ens19.300 up
ip addr add 192.168.30.1/24 dev ens19.300
ip route add 0.0.0.0/0 via 192.168.30.254
```

- Доступ в сеть Интернет с **sw1-cod**:

```
[root@localhost ~]# ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data:
64 bytes from 77.88.8.8: icmp_seq=1 ttl=47 time=92.1 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=47 time=89.3 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=47 time=91.4 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 89.266/90.936/92.147/1.220 ms
[root@localhost ~]#
```

Последнее изменение: четверг, 13 ноября 2025, 13:38

Предыдущий
элемент курса

[8. Настройка доступа к
веб-интерфейсу fw-cod и
лицензирование](#)

Перейти на...

Следующий элемент
курса

[10. Настройка
коммутации между sw1-
cod и sw2-cod](#)

Обратная связь

Вы используете гостевой
доступ (Вход)

Подпишитесь

Сводка хранения данных

Тема оформления сайта разработана

conecti.me

10. Настройка коммутации между sw1-cod и sw2-cod

[REG26](#)

> 10. Настройка коммутации между sw1-cod и sw2-cod

Вариант реализации:

sw1-cod (alt-server):

Назначение имени на устройство:

- Для назначения имени устройства согласно топологии используем следующую команду:

```
hostnamectl set-hostname sw1-cod.cod.ssa2026.region; exec bash
```

- Так же рекомендуется указать имя в файле `/etc/sysconfig/network`:

```
vim /etc/sysconfig/network
```

- указать имя в параметре **HOSTNAME**:

```
# When set to no, this may cause most daemons' initscripts skip starting
NETWORKING=yes

# Used by hotplug/pccmcia/ifplugd scripts to detect current network config
# subsystem.
CONFMETHOD=etcnet

# Used by rc.susinit to setup system hostname at boot.
HOSTNAME=sw1-cod.cod.ssa2026.region

# This is used by ALTlinux ppp-common to decide if we want to install
# nameserver lines into /etc/resolv.conf or not.
RESOLV_MODS=yes
```

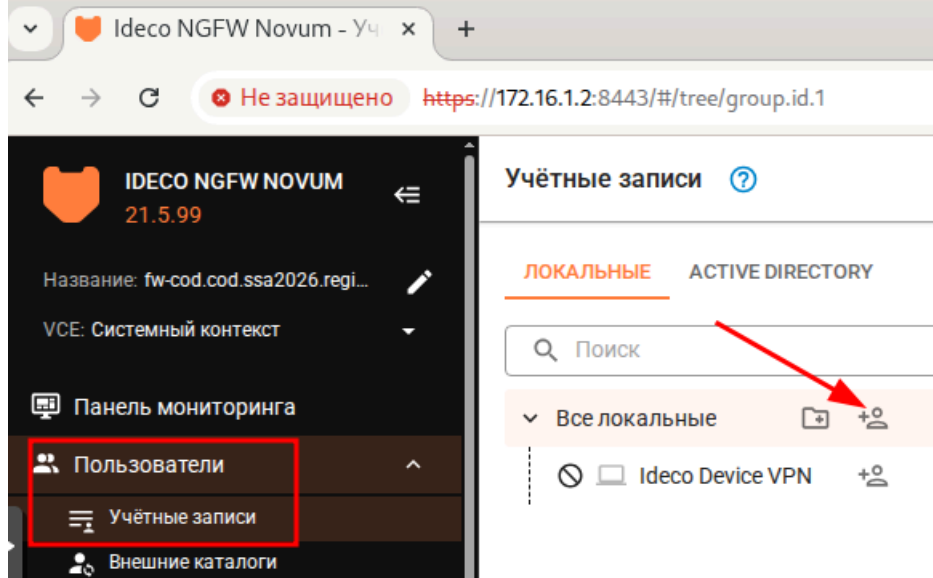
- Проверить можно с помощью команды **hostname** с ключём **-f**:

```
[root@sw1-cod ~]# hostname -f
sw1-cod.cod.ssa2026.region
[root@sw1-cod ~]# _
```

cli1-a (alt-workstation):

Настройка авторизации на fw-cod для доступа в сеть Интернет:

- Авторизация - необходимое условие для доступа пользователя в интернет. Для работы в пределах локальной сети авторизация не требуется
- Для доступа сетевого устройства (хоста) в интернет через NGFW Novum с возможностью контроля трафика, хост должен быть авторизован в системе под учетной записью пользователя
- Открываем браузер и переходим по <https://172.16.1.2:8443> (IP-адрес fw-cod), выполняем вход в веб-интерфейс управления fw-cod
- Для авторизации необходимо создать пользователя, для этого перейдём в **Пользователи** -> **Учётные записи** и нажмём **Добавить пользователя**:



- В форме **Добавления пользователя** заполняем только **Имя пользователя** (произвольное) и **Логин** (произвольный)
 - пароль можно оставить сгенерированный случайным образом, он нам не потребуется

Добавить пользователя в группу «Все локальные»

Основные настройки

Имя пользователя
network

Логин
network

Пароль
.....

Повторите пароль
.....

Рекомендуется использовать комбинацию цифр и букв

Надежный

Телефон
.....

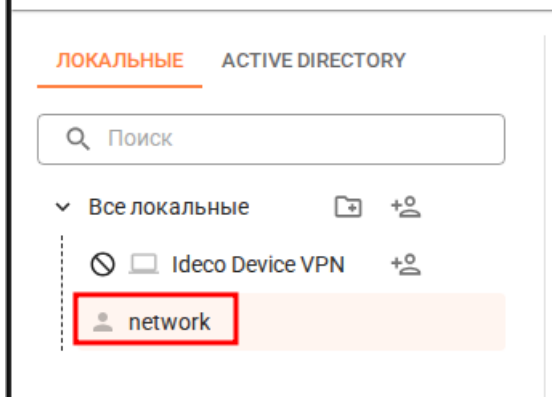
Формат: знак «плюс» (+), код страны, код региона и номер телефона

Дополнительные настройки

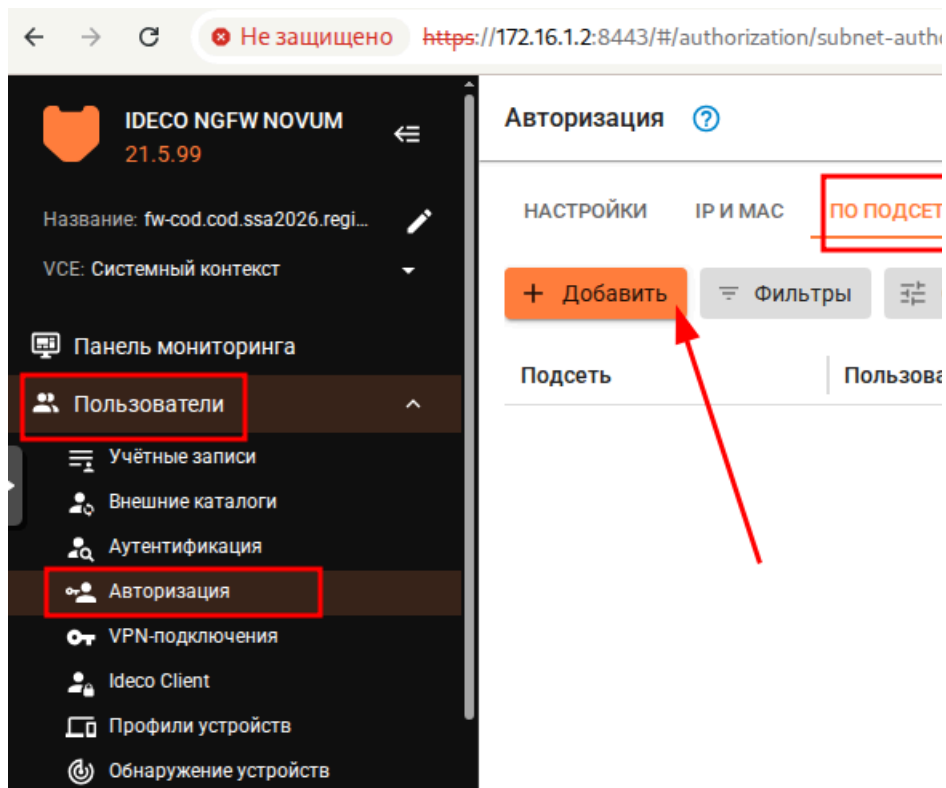
IP-адрес
.....

Поле необязательное

- Результат успешного добавления пользователя:



- Далее перейдёт в **Пользователи** -> **Авторизация** -> **ПО ПОДСЕТЯД** и нажмём **Добавить**:



- В форме **Добавление правила авторизации** указываем только что созданного пользователя и **подсеть**
 - в данном случае в сеть **192.168.0.0/16** входят все подсети **COD-a** (именно 192.168.10.0/24, 192.168.20.0/24, 192.168.30.0/24, 192.168.40.0/24, 192.168.50.0/24);

Добавление правила авторизации

Пользователь
network

Подсеть
192.168.0.0/16

Комментарий
0/256

Добавить
Отмена

- Результат успешного добавления Авторизации по подсетям:

+ Добавить	Фильтры	Отображение	Поиск
Подсеть	Пользователь	Каталог	
192.168.0.0/16	network	Локальные пользо...	

Настройка коммутации:

- Проверяем интерфейсы и определяемся какой к кому направлен (сверка производится по MAC-адресам):
 - таким образом, имеем (в данном конкретном случае):
 - ens19 - интерфейс в сторону fw-cod;
 - ens21 - интерфейс в сторону sw2-cod;
 - ens22 - интерфейс в сторону sw2-cod;
 - enp2s1 - интерфейс в сторону srv2-cod;
 - enp2s29 - интерфейс в сторону admin-cod;
 - enp3s12 - интерфейс в сторону srv2-cod

```
[root@sw1-cod ~]# ip -c -br a
lo UNKNOWN
ens19 DOWN
ens21 DOWN
ens22 DOWN
enp2s1 UP
enp2s29 DOWN
enp3s12 UP
[root@sw1-cod ~]#
```

- Для каждого интерфейса необходимо создать директорию по пути /etc/net/ifaces/ с помощью команды mkdir:

```
[root@sw1-cod ~]# ls /etc/net/ifaces/
default enp2s1 enp2s29 enp3s12 ens19 ens20 ens21 ens22 lo unknow
[root@sw1-cod ~]#
```

- Для каждого интерфейса в директории `/etc/net/ifaces/` **<ИМЯ_ИНТЕРФЕЙСА>** необходимо создать файл **options**
 - указав в нём два основных параметра:
 - **TYPE=eth**
 - **BOOTPROTO=static**
 - после чего необходимо перезагрузить службу **network**
 - все интерфейсы должны перейти в статус **UP**:

```
(root@sw1-cod ~)# ip -c -br a
lo                UNKNOWN
ens19             UP
ens21             UP
ens22             UP
enp2s1            UP
enp2s29           UP
enp3s12           UP
(root@sw1-cod ~)#
```

- Временно на базе интерфейса в сторону **fw-cod** создадим подинтерфейс с указанием **vlan300** для дальнейшей установки пакета **openvswitch**:

```
ip link add link ens19 name ens19.300 type vlan id 300
ip link set dev ens19.300 up
ip addr add 192.168.30.1/24 dev ens19.300
ip route add 0.0.0.0/0 via 192.168.30.254
echo "nameserver 77.88.8.8" > /etc/resolv.conf
```

- После чего обновляем список пакетов и устанавливаем **openvswitch**:

```
apt-get update && apt-get install -y openvswitch
```

- Включаем и добавляем в автозагрузку **openvswitch**:

```
systemctl enable --now openvswitch
```

- Правим основной файл **options** в котором по умолчанию сказано:
 - удалять настройки заданные через **ovs-vsctl**,
 - т.к. через **etcdnet** будет выполнено только создание интерфейса типа **internal**
 - с назначением необходимого IP-адреса, а настройка коммутации будет выполнена средствами **openvswitch**

```
sed -i "s/OVS_REMOVE=yes/OVS_REMOVE=no/g"
/etc/net/ifaces/default/options
```

- Перезагрузить сервер - будет быстрее чем удалять параметры заданные в ручную через пакет **iproute2**:

```
reboot
```

- Чтобы на **sw2-cod** была возможность установить пакет **openvswitch**:
 - временно создадим простой коммутатор с именем, например **br0**
 - и добавим в него интерфейсы **ens19** в сторону **fw-cod** и **ens21** в сторону **sw2-cod**

```
ovs-vsctl add-br br0
```

```
ovs-vsctl add-port br0 ens19
```

```
ovs-vsctl add-port br0 ens21
```

sw2-cod (alt-server):

Назначение имени на устройство:

- Реализация аналогично **sw1-cod**:

```
root@sw2-cod ~# hostname -f
sw2-cod.cod.ssa2026.region
root@sw2-cod ~#
```

- Для установки пакета **openvswitch** необходим доступ в сеть Интернет, для этого необходимо на виртуальной машине **sw2-cod**:
 - назначив средствами **iproute2** временно на интерфейс, смотрящий в сторону **sw1-cod (ens19)**,
 - тегированный подинтерфейс с IP-адресом из подсети для **vlan300**, а также шлюзом по умолчанию и DNS

```
ip link add link ens19 name ens19.300 type vlan id 300
ip link set up ens19
ip link set up ens19.300
ip addr add 192.168.30.2/24 dev ens19.300
ip route add 0.0.0.0/0 via 192.168.30.254
echo "nameserver 77.88.8.8" > /etc/resolv.conf
```

- После чего обновляем список пакетов и устанавливаем **openvswitch**:

```
apt-get update && apt-get install -y openvswitch
```

- Включаем и добавляем в автозагрузку **openvswitch**:

```
systemctl enable --now openvswitch
```

- Правим основной файл **options** в котором по умолчанию сказано:
 - удалять настройки заданные через **ovs-vsctl**,
 - т.к. через **etcdnet** будет выполнено только создание интерфейса типа **internal**
 - с назначением необходимого IP-адреса, а настройка коммутации будет выполнена средствами **openvswitch**

```
sed -i "s/OVS_REMOVE=yes/OVS_REMOVE=no/g"
/etc/net/ifaces/default/options
```

- Перезагрузить сервер - будет быстрее чем удалять параметры заданные в ручную через пакет **iproute2**:

```
reboot
```

sw1-cod (alt-server):

Настройка коммутации:

- Удаляем ранее созданный временный коммутатор с именем **br0**:

```
ovs-vsctl del-br br0
```

- Создадим коммутатор с именем **sw1-cod**:

```
ovs-vsctl add-br sw1-cod
```

- Проверить создание коммутатора можно с помощью команды **ovs-vsctl show**:

```
[root@sw1-cod ~]# ovs-vsctl show
ca12e05a-0291-425e-8290-ac709fdaf53f
    Bridge sw1-cod
        Port sw1-cod
            Interface sw1-cod
                type: internal
        ovs_version: "3.3.2"
[root@sw1-cod ~]#
```

- Добавим интерфейс, направленный в сторону **admin-cod** (enp2s29) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 300**:

```
ovs-vsctl add-port sw1-cod enp2s29 tag=300
```

- Добавим интерфейс, направленный в сторону **srv2-cod** (enp2s1) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 100**:

```
ovs-vsctl add-port sw1-cod enp2s1 tag=100
```

- Добавим интерфейс, направленный в сторону **srv2-cod** (enp3s12) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 200**:

```
ovs-vsctl add-port sw1-cod enp3s12 tag=200
```

- Интерфейс в сторону **fw-cod** (ens19) добавляем в созданный коммутатор, но настраиваем как магистральный (trunk) порт:
 - также разрешаем пропуск только требуемых VLAN (100,200,300,400 и 500)

```
ovs-vsctl add-port sw1-cod ens19 trunk=100,200,300,400,500
```

- Проверить добавление портов в коммутатор можно с помощью команды **ovs-vsctl show**:

```
[root@sw1-cod ~]# ovs-vsctl show
ca12e05a-0291-425e-8290-ac709fdaf53f
    Bridge sw1-cod
        Port enp3s12
            tag: 200
            Interface enp3s12
        Port enp2s1
            tag: 100
            Interface enp2s1
        Port sw1-cod
            Interface sw1-cod
                type: internal
        Port ens19
            trunks: [100, 200, 300, 400, 500]
            Interface ens19
        Port enp2s29
            tag: 300
            Interface enp2s29
    ovs_version: "3.3.2"
[root@sw1-cod ~]#
```

- Включаем модуль ядра отвечающий за тегированный трафик (**802.1Q**):

```
modprobe 8021q
```

```
echo "8021q" | tee -a /etc/modules
```

- На базе интерфейсов **ens21** и **ens22**, направленных в сторону **sw2-cod**, создадим **bond**-интерфейс в режиме **active-backup**:

```
ovs-vsctl add-bond sw1-cod bond0 ens21 ens22 bond_mode=active-backup
```

- Интерфейс типа **bond** в сторону **sw2-cod** (**bond0**) настраиваем как магистральный (trunk) порт:
 - также разрешаем пропуск только требуемых VLAN (100,200,300,400 и 500)

```
ovs-vsctl set port bond0 trunk=100,200,300,400,500
```

- Проверить можно с помощью команды **ovs-vsctl show**:

```
(root@sw1-cod ~)# ovs-vsctl show
ca12e05a-0291-425e-8290-ac709fdaf53f
Bridge sw1-cod
  Port enp3s12
    tag: 200
    Interface enp3s12
  Port enp2s1
    tag: 100
    Interface enp2s1
  Port bond0
    trunks: [100, 200, 300, 400, 500]
    Interface ens22
    Interface ens21
  Port sw1-cod
    Interface sw1-cod
    type: internal
  Port ens19
    trunks: [100, 200, 300, 400, 500]
    Interface ens19
  Port enp2s29
    tag: 300
    Interface enp2s29
  ovs_version: "3.3.2"
(root@sw1-cod ~)# _
```

- Режим работы **bond** интерфейса проверить можно с помощью команды **ovs-appctl bond/show**:

```
(root@sw1-cod ~)# ovs-appctl bond/show
---- bond0 ----
bond_mode: active-backup
bond may use recirculation: no, Recirc-ID : -1
bond-hash-basis: 0
lb_output action: disabled, bond-id: -1
updelay: 0 ms
downdelay: 0 ms
lacp_status: off
lacp_fallback_ab: false
active-backup primary: <none>
active member mac: bc:24:11:81:8e:b2(ens22)

member ens21: enabled
  may_enable: true

member ens22: enabled
  active member
  may_enable: true
oot@sw1-cod ~)# _
```

- Сетевая подсистема **etcdnet** будет взаимодействовать с **openvswitch**, для того чтобы корректно можно было назначить IP-адрес на интерфейс управления
 - создаём каталог для management интерфейса с именем **mgmt-cod**:

```
mkdir /etc/net/ifaces/mgmt-cod
```

- Описываем файл **options** для создания management интерфейса с именем **mgm-codt**:

```
vim /etc/net/ifaces/mgmt-cod/options
```

◦ где:

- **TYPE** - тип интерфейса (**internal**);
- **BOOTPROTO** - определяет как будут назначаться сетевые параметры (статически);
- **CONFIG_IPV4** - определяет использовать конфигурацию протокола IPv4 или нет;
- **BRIDGE** - определяет к какому мосту необходимо добавить данный интерфейс;
- **VID** - определяет принадлежность интерфейса к VLAN;

```
TYPE=ovsport
BOOTPROTO=static
CONFIG_IPV4=yes
BRIDGE=sw1-cod
VID=300
```

- Назначаем IP-адрес и шлюз на созданный интерфейс **mgmt**:

```
echo "192.168.30.1/24" > /etc/net/ifaces/mgmt-cod/ipv4address
```

```
echo "default via 192.168.30.254" > /etc/net/ifaces/mgmt-cod/ipv4route
```

- Для применения настроек, необходимо перезагрузить службу **network**:

```
systemctl restart network
```

- Проверить назначенный IP-адрес можно командой **ip a**:

```
[root@sw1-cod ~]# ip -c -br -4 a
lo                UNKNOWN    127.0.0.1/8
mgmt-cod          UNKNOWN    192.168.30.1/24
root@sw1-cod ~#
```

- Проверить назначенный IP-адрес шлюза по умолчанию можно командой **ip r**:

```
[root@sw1-cod ~]# ip -c r
default via 192.168.30.254 dev mgmt-cod
192.168.30.0/24 dev mgmt-cod proto kernel scope link src 192.168.30.1
[root@sw1-cod ~]#
```

- Также стоит с помощью команды **ovs-vsctl show** проверить, что данный интерфейс добавился в коммутатор **sw1-cod**:

```
[root@sw1-cod ~]# ovs-vsctl show
ca12e05a-0291-425e-8290-ac709fdaf53f
    Bridge sw1-cod
        Port enp3s12
            tag: 200
            Interface enp3s12
        Port mgmt-cod
            tag: 300
            Interface mgmt-cod
                type: internal
        Port enp2s1
            tag: 100
            Interface enp2s1
        Port bond0
            trunks: [100, 200, 300, 400, 500]
            Interface ens22
            Interface ens21
        Port sw1-cod
            Interface sw1-cod
                type: internal
        Port ens19
            trunks: [100, 200, 300, 400, 500]
            Interface ens19
        Port enp2s29
            tag: 300
            Interface enp2s29
    ovs_version: "3.3.2"
[root@sw1-cod ~]#
```

- Помимо того, что интерфейс **mgmt-cod** является портом доступа (access) необходимо использовать NativeVLAN:

```
ovs-vsctl set port mgmt-cod vlan_mode=native-untagged
```

- Проверить можно с помощью команды **ovs-vsctl list port mgmt-cod**:

```
[root@sw1-cod ~]# ovs-vsctl list port mgmt-cod
_uuid                : 1434ac51-88f6-4ea8-8718-15a0bbec2714
bond_active_slave    : []
bond_downdelay       : 0
bond_fake_iface      : false
bond_mode            : []
bond_updelay         : 0
culans               : []
external_ids         : {}
fake_bridge          : false
interfaces           : [b865de0e-0251-4c5a-942a-14710cd69fa5]
lacp                 : []
mac                 : []
name                 : mgmt-cod
other_config         : {}
protected            : false
qos                  : []
rstp_statistics      : {}
rstp_status          : {}
statistics           : {}
status              : {}
tag                  : 300
trunks               : []
vlan_mode            : native-untagged
[root@sw1-cod ~]#
```

sw2-cod (alt-server):

- Проверяем интерфейсы и определяемся какой к кому направлен:
 - таким образом, имеем (в данном конкретном случае):
 - **ens19** - интерфейс в сторону **sw1-cod**;
 - **ens20** - интерфейс в сторону **sw1-cod**;
 - **ens21** - интерфейс в сторону **srv1-cod**;
 - **ens22** - интерфейс в сторону **srv1-cod**;
 - **enp2s29** - интерфейс в сторону **sip-cod**;
 - **enp3s12** - интерфейс в сторону **cli-cod**;


```
[root@sw2-cod ~]# ip -c -br a
lo                UNKNOWN
ens19             DOWN
ens20             DOWN
ens21             DOWN
ens22             DOWN
enp2s29           UP
enp3s12           DOWN
[root@sw2-cod ~]#
```

- Для каждого интерфейса в директории `/etc/net/ifaces/` **<ИМЯ_ИНТЕРФЕЙСА>** необходимо создать файл **options**
 - указав в нём два основных параметра:
 - **TYPE=eth**
 - **BOOTPROTO=static**
 - после чего необходимо перезагрузить службу **network**
 - все интерфейсы должны перейти в статус **UP**:

```
[root@sw2-cod ~]# ip -c -br a
lo                UNKNOWN
ens19             UP
ens20             UP
ens21             UP
ens22             UP
enp2s29           UP
enp3s12           UP
[root@sw2-cod ~]#
```

- Создадим коммутатор с именем **sw2-cod**:

```
ovs-vsctl add-br sw2-cod
```

- Проверить создание коммутатора можно с помощью команды **ovs-vsctl show**:

```
[root@sw2-cod ~]# ovs-vsctl show
a83d6bb0-91b6-49ec-be50-3c19857b3947
    Bridge sw2-cod
        Port sw2-cod
            Interface sw2-cod
                type: internal
        ovs_version: "3.3.2"
[root@sw2-cod ~]#
```

- Добавим интерфейс, направленный в сторону **srv1-cod** (ens21) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 100**:

```
ovs-vsctl add-port sw2-cod ens21 tag=100
```

- Добавим интерфейс, направленный в сторону **srv1-cod** (ens22) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 200**:

```
ovs-vsctl add-port sw2-cod ens22 tag=200
```

- Добавим интерфейс, направленный в сторону **sip-cod** (enp2s29) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 500**:

```
ovs-vsctl add-port sw2-cod enp2s29 tag=500
```

- Добавим интерфейс, направленный в сторону **cli-cod** (enp3s12) в созданный коммутатор и назначим его в качестве порта доступа (access), указав принадлежность к **VLAN 400**:

```
ovs-vsctl add-port sw2-cod enp3s12 tag=400
```

- Проверить добавление портов в коммутатор можно с помощью команды **ovs-vsctl show**:

```
[root@sw2-cod ~]# ovs-vsctl show
a83d6bb0-91b6-49ec-be50-3c19857b3947
  Bridge sw2-cod
    Port ens21
      tag: 100
    Interface ens21
    Port enp2s29
      tag: 500
    Interface enp2s29
    Port enp3s12
      tag: 400
    Interface enp3s12
    Port sw2-cod
    Interface sw2-cod
      type: internal
    Port ens22
      tag: 200
    Interface ens22
  ovs_version: "3.3.2"
[root@sw2-cod ~]# _
```

- Включаем модуль ядра отвечающий за тегированный трафик (802.1Q):

```
modprobe 8021q
```

```
echo "8021q" | tee -a /etc/modules
```

- На базе интерфейсов **ens19** и **ens20**, направленных в сторону **sw1-cod**, создадим **bond**-интерфейс в режиме **active-backup**:

```
ovs-vsctl add-bond sw2-cod bond0 ens19 ens20 bond_mode=active-backup
```

- Интерфейс типа **bond** в сторону **sw2-cod** (**bond0**) настраиваем как магистральный (trunk) порт:
 - также разрешаем пропуск только требуемых VLAN (100,200,300,400 и 500)

```
ovs-vsctl set port bond0 trunk=100,200,300,400,500
```

- Проверить можно с помощью команды **ovs-vsctl show**:

```
[root@sw2-cod ~]# ovs-vsctl show
a83d6bb0-91b6-49ec-be50-3c19857b3947
  Bridge sw2-cod
    Port ens21
      tag: 100
    Interface ens21
    Port enp2s29
      tag: 500
    Interface enp2s29
    Port enp3s12
      tag: 400
    Interface enp3s12
    Port sw2-cod
    Interface sw2-cod
      type: internal
    Port ens22
      tag: 200
    Interface ens22
    Port bond0
      trunks: [100, 200, 300, 400, 500]
      Interface ens19
      Interface ens20
  ovs_version: "3.3.2"
[root@sw2-cod ~]# _
```

- Режим работы **bond** интерфейса проверить можно с помощью команды **ovs-appctl bond/show**:

```
[root@sw2-cod ~]# ovs-appctl bond/show
---- bond0 ----
bond_mode: active-backup
bond may use recirculation: no, Recirc-ID : -1
bond-hash-basis: 0
lb_output action: disabled, bond-id: -1
updelay: 0 ms
downdelay: 0 ms
lACP_status: off
lACP_fallback_ab: false
active-backup primary: <none>
active member mac: bc:24:11:2b:a9:69(ens20)

member ens19: enabled
  may_enable: true

member ens20: enabled
  active member
  may_enable: true

[root@sw2-cod ~]#
```

- Сетевая подсистема **etcdnet** будет взаимодействовать с **openvswitch**, для того чтобы корректно можно было назначить IP-адрес на интерфейс управления
 - создаём каталог для management интерфейса с именем **mgmt-cod**:

```
mkdir /etc/net/ifaces/mgmt-cod
```

- Описываем файл **options** для создания management интерфейса с именем **mgm-codt**:

```
vim /etc/net/ifaces/mgmt-cod/options
```

- где:
 - **TYPE** - тип интерфейса (**internal**);
 - **BOOTPROTO** - определяет как будут назначаться сетевые параметры (статически);
 - **CONFIG_IPV4** - определяет использовать конфигурацию протокола IPv4 или нет;
 - **BRIDGE** - определяет к какому мосту необходимо добавить данный интерфейс;
 - **VID** - определяет принадлежность интерфейса к VLAN;

```
TYPE=ovsport
BOOTPROTO=static
CONFIG_IPV4=yes
BRIDGE=sw2-cod
VID=300
```

- Назначаем IP-адрес и шлюз на созданный интерфейс **mgmt**:

```
echo "192.168.30.2/24" > /etc/net/ifaces/mgmt-cod/ipv4address
```

```
echo "default via 192.168.30.254" > /etc/net/ifaces/mgmt-cod/ipv4route
```

- Для применения настроек, необходимо перезагрузить службу **network**:

```
systemctl restart network
```

- Проверить назначенный IP-адрес можно командой **ip a**:

```

[root@sw2-cod ~]# ip -c -br -4 a
lo UNKNOWN 127.0.0.1/8
mgmt-cod UNKNOWN 192.168.30.2/24
[root@sw2-cod ~]# _

```

- Проверить назначенный IP-адрес шлюза по умолчанию можно командой **ip r**:

```

[root@sw2-cod ~]# ip -c r
default via 192.168.30.254 dev mgmt-cod
192.168.30.0/24 dev mgmt-cod proto kernel scope link src 192.168.30.2
[root@sw2-cod ~]#

```

- Также стоит с помощью команды **ovs-vsctl show** проверить, что данный интерфейс добавился в коммутатор **sw1-cod**:

```

[root@sw2-cod ~]# ovs-vsctl show
a83d6bb0-91b6-49ec-be50-3c19857b3947
    Bridge sw2-cod
        Port ens21
            tag: 100
        Interface ens21
        Port mgmt-cod
            tag: 300
        Interface mgmt-cod
            type: internal
        Port enp2s29
            tag: 500
        Interface enp2s29
        Port enp3s12
            tag: 400
        Interface enp3s12
        Port sw2-cod
            Interface sw2-cod
                type: internal
        Port ens22
            tag: 200
        Interface ens22
        Port bond0
            trunks: [100, 200, 300, 400, 500]
            Interface ens19
            Interface ens20
    ovs_version: "3.3.2"
[root@sw2-cod ~]# _

```

- Помимо того, что интерфейс **mgmt-cod** является портом доступа (access) необходимо использовать NativeVLAN:

```
ovs-vsctl set port mgmt-cod vlan_mode=native-untagged
```

- Проверить можно с помощью команды **ovs-vsctl list port mgmt-cod**:

```

[root@sw2-cod ~]# ovs-vsctl list port mgmt-cod
_uuid          : 4db47dc6-c94c-491d-9b53-8642eb526930
bond_active_slave : []
bond_downdelay   : 0
bond_fake_iface  : false
bond_mode        : []
bond_updelay     : 0
culans          : []
external_ids     : {}
fake_bridge      : false
interfaces       : [69720e2f-9ba0-44f0-8cfb-e460f6f23b94]
lACP            : []
mac             : []
name            : mgmt-cod
other_config     : {}
protected       : false
qos             : []
rstp_statistics  : {}
rstp_status      : {}
statistics       : {}
status          : {}
    ▸ g          : 300
    trunks       : []
    vlan_mode     : native-untagged
[root@sw2-cod ~]# _

```

Предыдущий
элемент курса

9. Создание
интерфейсов типа VLAN
на fw-cod для
маршрутизации между
VLAN

Перейти на...

Следующий элемент
курса

11. Настройка имён и IP-
адресов на всех
остальных устройствах

Обратная связь

Подпишитесь

Вы используете гостевой
доступ (Вход)

Сводка хранения данных

Тема оформления сайта разработана

conecti.me

11. Настройка имён и IP-адресов на всех остальных устройствах

[REG26](#)

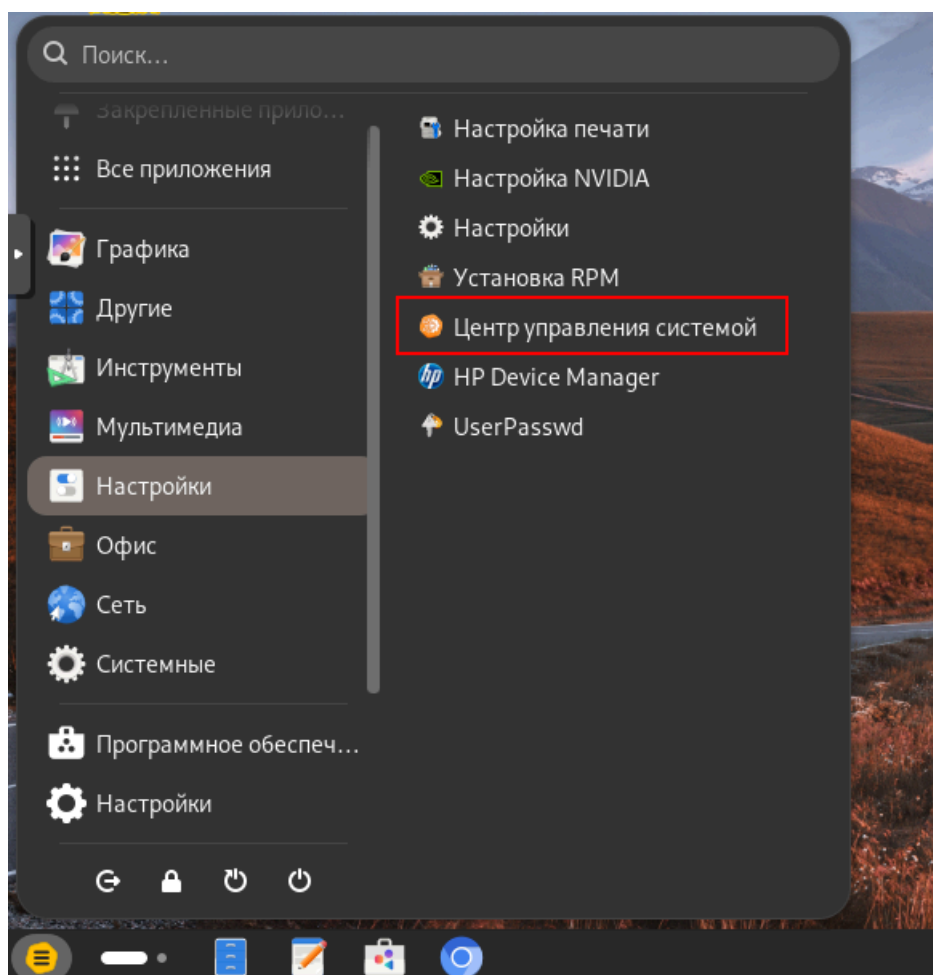
- 11. Настройка имён и IP-адресов на всех остальных устройствах

Вариант реализации:

cli-cod (alt-workstation):

Базовая настройка устройства:

- Перейдём в Центр Управления Системой (ЦУС/acc):



- В ЦУС перейдём в раздел Ethernet-интерфейсы:



Система

Информация об установленной системе и её настройка

[Дата и время](#) [Информация о дистрибутиве](#) [Лицензионное соглашени](#)
[Настройка нескольких рабочих мест](#) [Настройка ограничений](#) [Обнов](#)
[Обновление ядра](#) [Сетевые каталоги](#) [Системные журналы](#) [Системни](#)
[Системные службы](#) [Управление ключами SSL](#)



Пользователи

Управление пользователями системы

[Администратор системы](#) [Аутентификация](#) [Использование диска](#) [Лс](#)
[Локальные учётные записи](#)



Брандмауэр

Брандмауэр

[Внешние сети](#) [Перенаправление портов](#) [Список блокируемых хостов](#)



Сеть

Настройка подключения к сети

[Ethernet-интерфейсы](#) [OpenVPN-соединения](#) [PPPoE-соединения](#) [PF](#)
[Прокси-сервер](#)



Графический интерфейс

Настройка устройств ввода-вывода

[Дисплей](#) [Клавиатура](#)

- Назначим имя на устройство в соответствии с топологией:

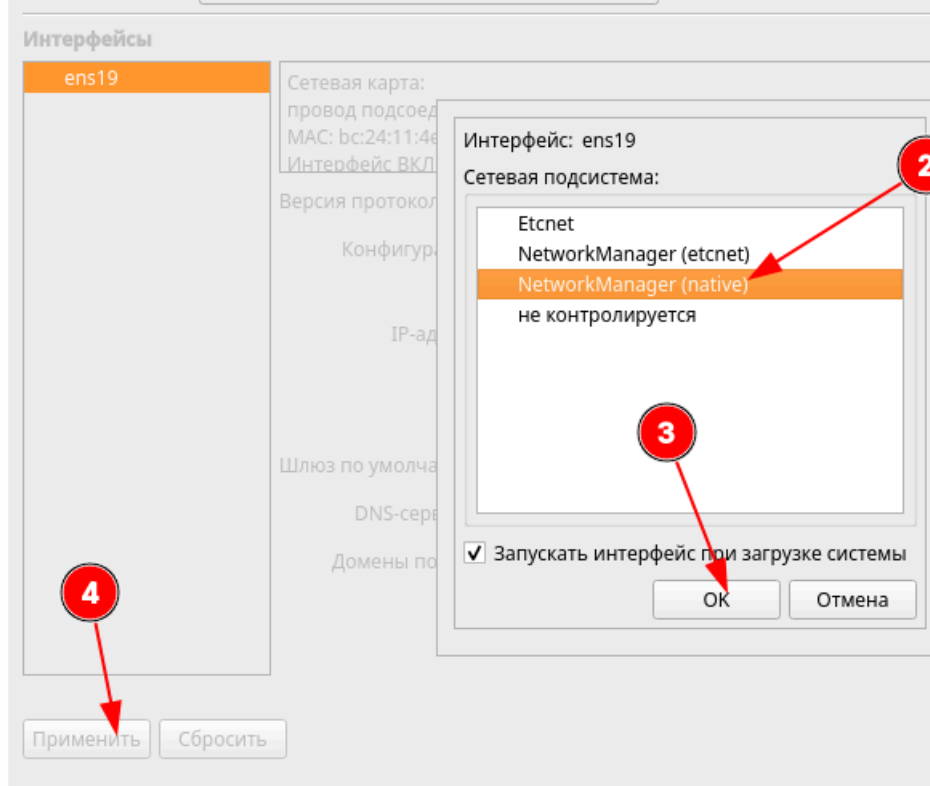
Имя компьютера: cli-cod.cod.ssa2026.region

Интерфейсы

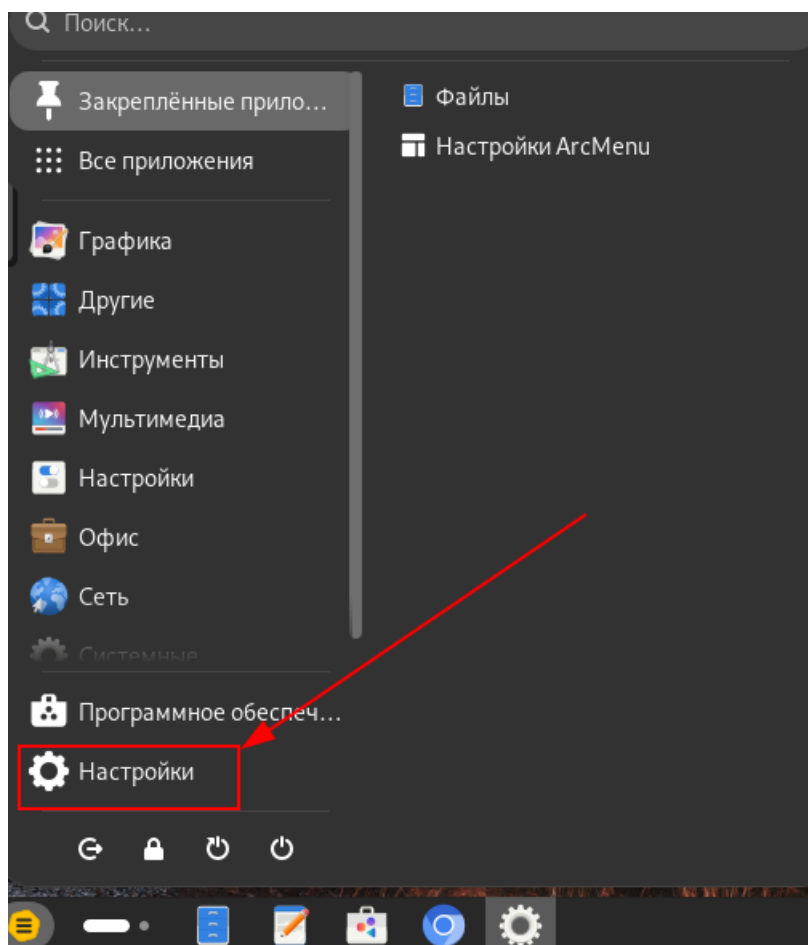
ens19	Сетевая карта:
	провод подсоединён
	MAC: bc:24:11:c8:17:aa
	Интерфейс ВКЛЮЧЕН
	Версия протокола IP: IP
	Конфигурация: И
	IP-адреса:
	Дс
	Шлюз по умолчанию:
	DNS-серверы:
	Домены поиска:
	(не

Применить Сбросить

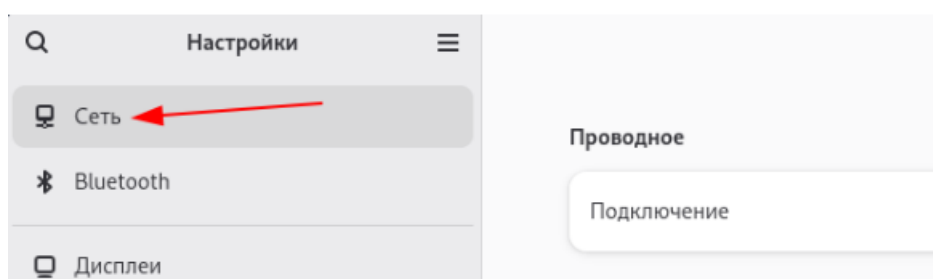
- В качестве режима работы сетевой подсистемы выберем **NetworkManager (native)**:



- Перейдём в обычный **Настройки** для назначения сетевых параметров:



- В разделе **Сеть** перейдём к настройке сетевого подключения:



- Задаём **Вручную** и нажимаем **Применить**:

- Адрес
- Маску сети
- Шлюз
- DNS (в качестве DNS-сервера указываем IP-адрес **srv1-cod**)

Подробнее

Идентификация

IPv4

IPv6

Безопасность

Метод IPv4

☐ Автоматический (DHCP)
 ☒ Вручную
 ☐ Только для л
 ☐ Выключить

☐ Общий доступ другим компьютерам

Адреса

Адрес	Маска сети	Шлюз
192.168.40.40	255.255.255.0	192.168.40.254

DNS

192.168.10.1	
--------------	--

Отделяйте IP-адреса запятыми

- Проверить можно на вкладке **Подробнее**:

Отменить

Проводное подключение

Подробнее

Идентификация

IPv4

IPv6

Безопасность

Адрес IPv4 192.168.40.40

Адрес IPv6 fe80::c921:51f0:524:6ee3

Аппаратный адрес BC:24:11:C8:17:AA

Маршрут по умолчанию 192.168.40.254

DNS 192.168.10.1

- Также можно проверить доступ в сеть Интернет:

```
[user@cli-cod ~]$ ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=47 time=188 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=47 time=83.9 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=47 time=86.4 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 83.874/119.478/188.147/48.567 ms
[user@cli-cod ~]$
```

cli2-a (alt-workstation):

Базовая настройка устройства:

- Реализация аналогично **cli-cod**:
 - имя должно быть:

```
[user@cli2-a ~]$ hostname -f
cli2-a.office.ssa2026.region
[user@cli2-a ~]$
```

- сетевые параметры должны быть:

Проводное подключение	
Отменить	
Подробности	Идентификация IPv4 IPv6 Безопасность
Адрес IPv4	172.20.20.2
Адрес IPv6	fe80::4974:f3a7:5a7a:10d9
Аппаратный адрес	BC:24:11:19:9F:0F
Маршрут по умолчанию	172.20.20.254
DNS	172.20.10.10

- доступ в сеть Интернет должен быть:

```
[user@cli2-a ~]$ ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=48 time=48.3 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=48 time=46.5 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=48 time=43.5 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 43.456/46.097/48.345/2.015 ms
[user@cli2-a ~]$
```

- связность с устройствами **COD-a** должна быть, например с **cli-cod**:

```
[user@cli2-a ~]$ ping -c3 192.168.40.40
PING 192.168.40.40 (192.168.40.40) 56(84) bytes of data.
64 bytes from 192.168.40.40: icmp_seq=1 ttl=61 time=104 ms
64 bytes from 192.168.40.40: icmp_seq=2 ttl=61 time=107 ms
64 bytes from 192.168.40.40: icmp_seq=3 ttl=61 time=106 ms

--- 192.168.40.40 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 103.655/105.400/106.921/1.342 ms
[user@cli2-a ~]$ tracepath -n 192.168.40.40
1?: [LOCALHOST] pmtu 1500
 1: 172.20.20.254 13.304ms
 1: 172.20.20.254 68.337ms
 2: 172.20.20.254 14.964ms pmtu 1476
 2: 10.10.10.1 100.561ms
 3: 169.254.1.1 104.682ms asymm
 4: 192.168.40.40 108.025ms reached
Resume: pmtu 1476 hops 4 back 4
[user@cli2-a ~]$
```

admin-cod (alt-workstation):

Базовая настройка устройства:

- Реализация аналогично **cli-cod**:

```
[user@admin-cod ~]$ hostname -f
admin-cod.cod.ssa2026.region
[user@admin-cod ~]$ ip -c -br -4 a
lo                UNKNOWN    127.0.0.1/8
ens19             UP          192.168.30.30/24
[user@admin-cod ~]$ ip -c r
default via 192.168.30.254 dev ens19 proto static metric 100
192.168.30.0/24 dev ens19 proto kernel scope link src 192.168.30.30 metr
[user@admin-cod ~]$ cat /etc/resolv.conf
# Generated by resolvconf
# Do not edit manually, use
# /etc/net/ifaces/<interface>/resolv.conf instead.
search cod.ssa2026.region
nameserver 192.168.10.1
[user@admin-cod ~]$ ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=47 time=81.1 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=47 time=84.7 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=47 time=79.9 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 79.890/81.915/84.728/2.052 ms
[user@admin-cod ~]$
```

srv1-cod (alt-server):

Базовая настройка устройства:

- Для назначения имени устройства согласно топологии используем следующую команду:

```
hostnamectl set-hostname srv1-cod.cod.ssa2026.region; exec bash
```

- Так же рекомендуется указать имя в файле `/etc/sysconfig/network`:

```
vim /etc/sysconfig/network
```

- указать имя в параметре **HOSTNAME**:

```
# When set to no, this may cause most daemons' initscripts skip starting.
NETWORKING=yes

# Used by hotplug/pcmcia/ifplugd scripts to detect current network config
# subsystem.
CONFMETHOD=etcnet

# Used by rc.susinit to setup system hostname at boot.
HOSTNAME=srv1-cod.cod.ssa2026.region

# This is used by ALTlinux ppp-common to decide if we want to install
# nameserver lines into /etc/resolv.conf or not.
RESOLV_MODS=yes
```

- Проверить можно с помощью команды **hostname** с ключём **-f**:

```
(root@srv1-cod ~)# hostname -f
srv1-cod.cod.ssa2026.region
(root@srv1-cod ~)#
```

Проверяем интерфейсы и определяемся какой к кому направлен (сверка производится по MAC-адресам):

- таким образом, имеем (в данном конкретном случае):
 - **ens19** - интерфейс в сторону **sw2-cod (vlan100)**;
 - **ens21** - интерфейс в сторону **sw2-cod (vlan200)**;

```
[root@srv1-cod ~]# ip -c -br a
lo                UNKNOWN
ens19             UP
ens20             DOWN
[root@srv1-cod ~]#
```

Для каждого интерфейса в директории `/etc/net/ifaces/`
<ИМЯ_ИНТЕРФЕЙСА>/ необходимо создать файл `options`

- указав в нём два основных параметра:
 - `TYPE=eth`
 - `BOOTPROTO=static`

```
[root@srv1-cod ~]# ls /etc/net/ifaces/
default ens19 ens20 lo unknown
[root@srv1-cod ~]# cat /etc/net/ifaces/ens19/options
TYPE=eth
BOOTPROTO=static
[root@srv1-cod ~]# cat /etc/net/ifaces/ens20/options
TYPE=eth
BOOTPROTO=static
[root@srv1-cod ~]#
```

- Задаём IP-адрес на интерфейс **ens19 (vlan100)**:

```
echo "192.168.10.1/24" > /etc/net/ifaces/ens19/ipv4address
```

- Задаём IP-адрес шлюза по умолчанию для интерфейса **ens19 (vlan100)**:

```
echo "default via 192.168.10.254" > /etc/net/ifaces/ens19/ipv4route
```

- Задаём IP-адрес на интерфейс **ens20 (vlan200)**:
 - шлюз не задаётся, по заданию данный vlan не должен маршрутизироваться;

```
echo "192.168.20.1/24" > /etc/net/ifaces/ens20/ipv4address
```

- Перезагружаем службу **network** для применения всех настроек:

```
systemctl restart network
```

- Проверить:

```
[root@srv1-cod ~]# ip -c -br -4 a
lo                UNKNOWN    127.0.0.1/8
ens19             UP          192.168.10.1/24
ens20             UP          192.168.20.1/24
[root@srv1-cod ~]# ip -c r
default via 192.168.10.254 dev ens19
192.168.10.0/24 dev ens19 proto kernel scope link src 192.168.10.1
192.168.20.0/24 dev ens20 proto kernel scope link src 192.168.20.1
[root@srv1-cod ~]# ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=47 time=47.2 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=47 time=40.6 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=47 time=42.7 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 40.648/43.502/47.177/2.727 ms
[root@srv1-cod ~]#
```

srv2-cod (alt-server):

Базовая настройка устройства:

- Реализация аналогично **srv1-cod**:

```

[root@srv2-cod ~]# hostname -f
srv2-cod.cod.ssa2026.region
[root@srv2-cod ~]# ip -c -br -4 a
lo                UNKNOWN          127.0.0.1/8
ens19             UP                192.168.20.2/24
ens20             UP                192.168.10.2/24
[root@srv2-cod ~]# ip -c r
default via 192.168.10.254 dev ens20
192.168.10.0/24 dev ens20 proto kernel scope link src 192.168.10.2
192.168.20.0/24 dev ens19 proto kernel scope link src 192.168.20.2
[root@srv2-cod ~]# ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=47 time=77.7 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=47 time=72.5 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=47 time=72.7 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 72.536/74.323/77.719/2.402 ms
[root@srv2-cod ~]#

```

dc-a (alt-server):

Базовая настройка устройства:

- Реализация аналогично `srv1-cod`:

```

[root@dc-a ~]# hostname -f
dc-a.office.ssa2026.region
[root@dc-a ~]# ip -c -br -4 a
lo                UNKNOWN          127.0.0.1/8
ens19             UP                172.20.10.10/24
[root@dc-a ~]# ip -c r
default via 172.20.10.254 dev ens19
172.20.10.0/24 dev ens19 proto kernel scope link src 172.20.10.10
[root@dc-a ~]# ping -c3 77.88.8.8
PING 77.88.8.8 (77.88.8.8) 56(84) bytes of data.
64 bytes from 77.88.8.8: icmp_seq=1 ttl=48 time=53.0 ms
64 bytes from 77.88.8.8: icmp_seq=2 ttl=48 time=55.5 ms
64 bytes from 77.88.8.8: icmp_seq=3 ttl=48 time=50.4 ms

--- 77.88.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 50.419/52.974/55.536/2.089 ms
[root@dc-a ~]#

```

Последнее изменение: пятница, 14 ноября 2025, 13:39

Предыдущий
элемент курса

[10. Настройка
коммутации между sw1-
cod и sw2-cod](#)

Перейти на...

Следующий элемент
курса

[12. Настройка
административного
доступа \(RADIUS\)](#)

Обратная связь

Вы используете гостевой
доступ (Вход)

Подпишитесь

Сводка хранения данных

Тема оформления сайта разработана

conecti.me

12. Настройка административного доступа (RADIUS)

[REG26](#)

- 12. Настройка
- > административного доступа (RADIUS)

Вариант реализации:

srv1-cod (alt-server):

- Временно для возможности установки необходимых пакетов зададим публичный DNS-сервер:

```
echo "nameserver 77.88.8.8" > /etc/resolv.conf
```

- Устанавливаем пакеты **freeradius** и **freeradius-utils**:

```
apt-get update && apt-get install -y freeradius freeradius-utils
```

- Включаем службу **radiusd** в автозагрузку и запускаем ее:

```
systemctl enable --now radiusd
```

- Создаем клиентов добавляя в файле **/etc/raddb/clients.conf** следующее содержимое:
 - пример конфигурации файла **/etc/raddb/clients.conf** для любого клиента

```
client ALL {
    ipaddr = 0.0.0.0
    netmask = 0
    secret = P@ssw0rd
}
```

- Редактируем файл с пользователями **/etc/raddb/users** добавляя в самый конец следующее содержимое:

```
#####
# You should add test accounts to the TOP of this file! #
# See the example user "bob" above.                      #
#####
netuser Cleartext-Password := "P@ssw0rd"
      Service-Type = Administrative-User,
      Cisco-AVPair = "shell:roles=admin"
```

- Перезапускаем службу **radiusd**:

```
systemctl restart radiusd
```

rtr-cod (ecorouter):

- Выполняем подключение к RADIUS-серверу:

```
rtr-cod(config)#security none
rtr-cod(config)#aaa radius-server 192.168.10.1 port 1812 secret
P@ssw0rd auth
rtr-cod(config)#
```

- Задаём приоритет:

```
rtr-cod(config)#aaa precedence local radius
rtr-cod(config)#write memory
Building configuration...

rtr-cod(config)#
```

- Проверить возможность входа из-под пользователя **netuser** с паролем **P@ssw0rd**:

```
rtr-cod login: netuser
Password:
User Access Verification

EcoRouterOS version Jasmine 26/12/2024 23:46:47
rtr-cod>enable
rtr-cod#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
rtr-cod(config)#
```

- Проверить вход под локальной учетной записью даже при доступности RADIUS-сервера:

```
rtr-cod login: admin
Password:
User Access Verification

EcoRouterOS version Jasmine 26/12/2024 23:46:47
rtr-cod>
```

- Проверить доступ по **SSH** из-под пользователя **netuser** с паролем **P@ssw0rd**:
 - например с **srv1-cod**

```
[root@srv1-cod ~]# ssh netuser@172.16.1.1
The authenticity of host '172.16.1.1 (172.16.1.1)' can't be established.
ED25519 key fingerprint is SHA256:k5mMnJ+uMSuWIO8/QWzwU/CJFUsu/i46KD3nYx
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.16.1.1' (ED25519) to the list of known h
netuser@172.16.1.1's password:
User Access Verification

EcoRouterOS version Jasmine 26/12/2024 23:46:47
rtr-cod>enable
rtr-cod#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
rtr-cod(config)#
```

sw1-cod и sw2-cod (alt-server):

- Временно для возможности установки необходимых пакетов зададим публичный DNS-сервер:

```
echo "nameserver 77.88.8.8" > /etc/resolv.conf
```

- Устанавливаем пакет **pam_radius**:

```
apt-get update && apt-get install -y pam_radius
```

- Редактируем конфигурационный файл **/etc/pam_radius_auth.conf**:

```
# server[:port] shared secret timeout (s) source_ip
192.168.10.1 P@ssw0rd 3
#127.0.0.1 secret 3
#other-server other-secret 5 192.168.1.1
#[2001:0db8:85a3::4]:1812 other6-secret 3 [2001:0db8:
#other-other-server other-other-secret 5 0
#
# having localhost in your radius configuration is a Good Thing.
#
:wq?
```

- Редактируем конфигурационный файл **/etc/pam.d/sshd**:

```
#%PAM-1.0
auth required pam_userpass.so
auth sufficient pam_radius_auth.so
auth include common-login-use_first_pass
account include common-login
password include common-login
session include common-login
```

- Редактируем конфигурационный файл **/etc/pam.d/system-auth-local**:

```
#%PAM-1.0
auth sufficient pam_radius_auth.so
auth include system-auth-local-only
auth include system-auth-common
account include system-auth-local-only
account include system-auth-common
password include system-auth-local-only
password include system-auth-common
session include system-auth-local-only
session include system-auth-common
```

- Также в случае с **Linux**, данного пользователя необходимо создать локально:

```
useradd netuser
```

- Проверить возможность входа из-под пользователя **netuser** с паролем **P@ssw0rd**:

```
Welcome to ALT Server 11.0 (Mendeleevium)!
Hostname: sw2-cod.cod.ssa2026.region
IP: 192.168.30.2
sw2-cod login: netuser
Password:
[netuser@sw2-cod ~]$
```

- Проверить доступ по **SSH** из-под пользователя **netuser** с паролем **P@ssw0rd**:
 - например с **srv1-cod**

```
Last login: Mon Nov 17 13:00:16 MSK 2025 on tt01
[root@srv1-cod ~]# ssh netuser@192.168.30.2
The authenticity of host '192.168.30.2 (192.168.30.2)' can't be established
ED25519 key fingerprint is SHA256:A8TkadFGEBXdLvLNzB5Bgm2Jtf6g0UdS0sYusrC
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.30.2' (ED25519) to the list of known
netuser@192.168.30.2's password:
Last login: Mon Nov 17 13:06:04 2025
[netuser@sw2-cod ~]$
```


Предыдущий
элемент курса

11. Настройка имён и IP-адресов на всех остальных устройствах

Перейти на...

Следующий элемент
курса

13. Настройка синхронизации времени между сетевыми устройствами

Обратная связь

Подпишитесь

Вы используете гостевой доступ (Вход)

Сводка хранения данных

Тема оформления сайта разработана

conecti.me

13. Настройка синхронизации времени между сетевыми устройствами

[REG26](#)

13. Настройка
> синхронизации времени
 между сетевыми
 устройствами

Вариант реализации:

rtr-cod и rtr-a (ecorouter):

- Устанавливаем корректный часовой пояс:

```
rtr-cod(config)#ntp timezone utc+3
rtr-cod(config)#
```

- Устанавливаем синхронизацию времени с 100.100.100.100:

```
rtr-cod(config)#ntp server 100.100.100.100
rtr-cod(config)#write memory
Building configuration...

rtr-cod(config)#
```

- Проверить текущий часовой пояс можно командой **show ntp timezone** из режима администрирования (**enable**):

```
rtr-cod#show ntp timezone
System Time zone: Europe/Moscow
rtr-cod#
```

- Проверить адреса ntp-серверов для синхронизации можно командой **show ntp status** из режима администрирования (**enable**):

```
rtr-cod#show ntp status
Status Description
* best
+ sync
- failed
? unknown

-----
Status | VR name | Server | Stratum | Delay
-----
* | default | 100.100.100.100 | 5 | 0.2069
rtr-cod#
```

sw1-cod, sw2-cod, cli-cod, srv1-cod, srv2-cod, admin-cod, sw1-a, sw2-a, dc-a, cli1-a и cli2-a (alt-server, alt-workstation):

- Устанавливаем корректный часовой пояс:

```
timedatectl set-timezone Europe/Moscow
```

- Редактируем конфигурационный файл `/etc/chrony.conf`:

```
# Use public servers from the pool.ntp.org project.
# Please consider joining the pool (https://www.pool.ntp.org/join.html)
#pool pool.ntp.org iburst
pool 100.100.100.100 iburst

# Record the rate at which the system clock gains/losses time.
```

- Перезапускаем службу `chronyd`:

```
systemctl restart chronyd
```

- Проверяем с каким сервером синхронизировалось время с помощью команды `chronyc tracking`:

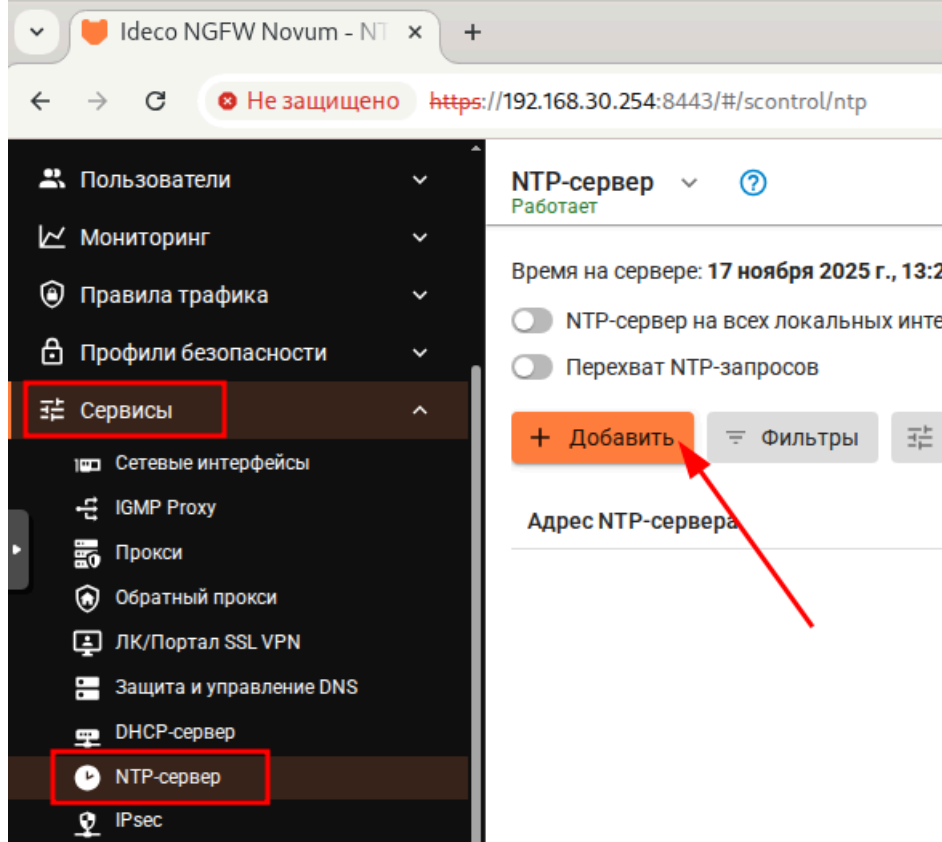
```
[root@sw1-cod ~]# chronyc tracking
Reference ID    : 64646464 (100.100.100.100)
Stratum        : 6
Ref time (UTC)  : Mon Nov 17 10:15:41 2025
System time     : 0.000004234 seconds slow of NTP time
Last offset     : +0.004109129 seconds
RMS offset      : 0.004109129 seconds
Frequency       : 9.536 ppm slow
Residual freq   : -2878.252 ppm
Skew            : 11.929 ppm
Root delay      : 0.096411645 seconds
Root dispersion : 0.071875237 seconds
Update interval : 2.0 seconds
Leap status     : Normal
[root@sw1-cod ~]# _
```

- Проверяем часовой пояс с помощью команды `timedatectl`:

```
[root@sw1-cod ~]# timedatectl
          Local time: Mon 2025-11-17 13:16:43 MSK
          Universal time: Mon 2025-11-17 10:16:43 UTC
           RTC time: Mon 2025-11-17 10:16:43
          Time zone: Europe/Moscow (MSK, +0300)
System clock synchronized: yes
              NTP service: active
           RTC in local TZ: no
[root@sw1-cod ~]# _
```

admin-cod (alt-workstation):

- Открываем браузер и переходим по <https://192.168.30.254:8443> (IP-адрес `fw-cod`):
 - выполняем вход в веб-интерфейс управления `fw-cod`
 - переходим Сервисы -> NTP-сервер и нажимаем **Добавить**:



- Указываем **100.100.100.100** и нажимаем **Добавить**:

NTP-сервер Работает

Добавление NTP-сервера

NTP-сервер 100.100.100.100

IP-адрес или доменное имя

Комментарий

0/256

Добавить **Отмена**

- Результат успешного добавления NTP-сервера:

NTP-сервер Работает

Время на сервере: **17 ноября 2025 г., 13:30:29 (Москва)**

☐ NTP-сервер на всех локальных интерфейсах (порт 123/UDP)

☐ Перехват NTP-запросов

+ Добавить **Фильтры** **Отображение**

Адрес NTP-сервера	Комментарий
100.100.100.100	

- Проверяем с каким сервером синхронизировалось время с помощью команды **chronyc tracking**:

Терминал

```
[admin@fw-cod ~]# chronyc tracking
Reference ID      : 64646464 (100.100.100.100)
Stratum          : 0
Ref time (UTC)   : Mon Nov 17 10:31:03 2025
System time      : 0.007790932 seconds fast of NTP time
Last offset      : +0.008523416 seconds
RMS offset       : 0.008937536 seconds
Frequency        : 10.231 ppm slow
Residual freq    : +358.057 ppm
Skew             : 5.477 ppm
Root delay       : 0.080065280 seconds
Root dispersion  : 0.027800325 seconds
Update interval  : 16.4 seconds
Leap status      : Normal
[admin@fw-cod ~]#
```

- Проверяем часовой пояс с помощью команды **timedatectl**:

Терминал

```
[admin@fw-cod ~]# timedatectl
      Local time: Пн 2025-11-17 13:32:00 MSK
      Universal time: Пн 2025-11-17 10:32:00 UTC
      RTC time: Пн 2025-11-17 10:32:00
      Time zone: Europe/Moscow (MSK, +0300)
System clock synchronized: yes
      NTP service: active
      RTC in local TZ: no
[admin@fw-cod ~]#
```

Последнее изменение: понедельник, 17 ноября 2025, 13:31

Предыдущий
элемент курса

[12. Настройка
административного
доступа \(RADIUS\)](#)

Перейти на...

Следующий элемент
курса

[14. Настройка службы
доменных имен в COD-e](#)

Обратная связь

Подпишитесь

Вы используете гостевой
доступ (Вход)

Сводка хранения данных

Тема оформления сайта разработана

conecti.me

14. Настройка службы доменных имен в COD-е

[REG26](#)

> 14. Настройка службы доменных имен в COD-е

Вариант реализации:

srv1-cod (alt-server):

- Установим пакет `bind` и `bind-utils`:

```
apt-get install -y bind bind-utils
```

- Отредактируем конфигурационный файл `/etc/bind/options.conf`:

```
options {
    version "unknown";
    directory "/etc/bind/zone";
    dump-file "/var/run/named/named_dump.db";
    statistics-file "/var/run/named/named.stats";
    recursing-file "/var/run/named/named.recursing";
    secroots-file "/var/run/named/named.secroots";

    // disables the use of a PID file
    pid-file none;

    /*
     * Oftenly used directives are listed below.
     */

    listen-on { any; };
    listen-on-v6 { none; };

    /*
     * If the forward directive is set to "only", the server will only
     * query the forwarders.
     */
    forward first;
    forwarders { 100.100.100.100; };

    /*
     * Specifies which hosts are allowed to ask ordinary questions.
     */
    allow-query { any; };

    /*
     * This lets "allow-query" be used to specify the default zone access
     * level rather than having to have every zone override the global
     * value. "allow-query-cache" can be set at both the options and
     * zone levels. If "allow-query-cache" is not set then "allow-recursion"
     * is used if set, otherwise "allow-query" is used if set unless
     * "recursion no;" is set in which case "none;" is used, otherwise
     * default (localhost; localnets;) is used.
     */
    allow-query-cache { any; };

    /*
     * Specifies which hosts are allowed to make recursive queries
     * through this server. If not specified, the default is to allow
     * recursive queries from all hosts. Note that disallowing recursive
     * queries for a host does not prevent the host from retrieving data
     * that is already in the server's cache.
     */
    allow-recursion { any; };
```

- Добавить в конфигурационный файл `/etc/bind/rfc1912.local` информацию о файлах зон прямого и обратного просмотра:

```

zone "cod.ssa2026.region" {
    type master;
    file "cod.ssa2026.region";
    allow-transfer { 172.20.10.10; };
};

zone "168.192.in-addr.arpa" {
    type master;
    file "168.192.in-addr.arpa";
    allow-transfer { 172.20.10.10; };
};

zone "office.ssa2026.region" {
    type forward;
    forward only;
    forwarders { 172.20.10.10; };
};

zone "20.172.in-addr.arpa" {
    type forward;
    forward only;
    forwarders { 172.20.10.10; };
};

```

- Скопировать файл шаблона для зоны прямого просмотра:

```
cp /etc/bind/zone/localhost /etc/bind/zone/cod.ssa2026.region
```

- Выдать права на файл зоны прямого просмотра:

```
chown root:named /etc/bind/zone/cod.ssa2026.region
```

- Привести файл `/etc/bind/zone/cod.ssa2026.region` зоны прямого просмотра к следующему виду:

```

$TTL      1D
;
; SOA cod.ssa2026.region. root.cod.ssa2026.region.
; serial
; refresh
; retry
; expire
; mcache
)

IN      NS      cod.ssa2026.region.
IN      A       192.168.10.1
rtr-cod IN      A       172.16.1.1
fw-cod  IN      A       192.168.10.254
sw1-cod IN      A       192.168.30.1
sw2-cod IN      A       192.168.30.2
cli-cod IN      A       192.168.40.40
srv1-cod IN     A       192.168.10.1
srv2-cod IN     A       192.168.10.2
sip-cod IN     A       192.168.50.50
admin-cod IN    A       192.168.30.30
monitoring IN   CNAME   srv1-cod.cod.ssa2026.region.

```

- Скопировать файл шаблона для зоны обратного просмотра:

```
cp /etc/bind/zone/localhost /etc/bind/zone/168.192.in-addr.arpa
```

- Выдать права на файл зоны прямого просмотра:

```
chown root:named /etc/bind/zone/168.192.in-addr.arpa
```

- Привести файл `/etc/bind/zone/168.192.in-addr.arpa` зоны обратного просмотра к следующему виду:

```

$TTL      1D      IN      SOA      cod.ssa2026.region. root.cod.ssa2026.region. (
                                2025110500      ; serial
                                12H              ; refresh
                                1H              ; retry
                                1W              ; expire
                                1H              ; ncache
                                )
254.10    IN      NS      cod.ssa2026.region.
1.30      IN      PTR     fw-cod.cod.ssa2026.region.
2.30      IN      PTR     sw1-cod.cod.ssa2026.region.
40.40     IN      PTR     sw2-cod.cod.ssa2026.region.
1.10      IN      PTR     cli-cod.cod.ssa2026.region.
2.10      IN      PTR     srv1-cod.cod.ssa2026.region.
50.50     IN      PTR     srv2-cod.cod.ssa2026.region.
30.30     IN      PTR     sip-cod.cod.ssa2026.region.
30.30     IN      PTR     admin-cod.cod.ssa2026.region.

```

- Включить и добавить в автозагрузку службу **bind**:

```
systemctl enable --now bind
```

- Задаём DNS-сервер и домен поиска:

```
cat <<EOF > /etc/net/ifaces/ens19/resolv.conf
search cod.ssa2026.region
nameserver 127.0.0.1
EOF
```

- Перезагружаем сервер:

```
reboot
```

- Проверяем записи типа **A**:

```

[root@srv1-cod ~]# host fw-cod
fw-cod.cod.ssa2026.region has address 192.168.10.254
[root@srv1-cod ~]# host sw1-cod
sw1-cod.cod.ssa2026.region has address 192.168.30.1
[root@srv1-cod ~]# host sw2-cod
sw2-cod.cod.ssa2026.region has address 192.168.30.2
[root@srv1-cod ~]# host cli-cod
cli-cod.cod.ssa2026.region has address 192.168.40.40
[root@srv1-cod ~]# host srv1-cod
srv1-cod.cod.ssa2026.region has address 192.168.10.1
[root@srv1-cod ~]# host srv2-cod
srv2-cod.cod.ssa2026.region has address 192.168.10.2
[root@srv1-cod ~]# host sip-cod
sip-cod.cod.ssa2026.region has address 192.168.50.50
[root@srv1-cod ~]# host admin-cod
admin-cod.cod.ssa2026.region has address 192.168.30.30
[root@srv1-cod ~]# _

```

- Проверяем записи типа **PTR**:

```

[root@srv1-cod ~]# host 192.168.10.254
254.10.168.192.in-addr.arpa domain name pointer fw-cod.cod.ssa2026.region
[root@srv1-cod ~]# host 192.168.30.1
1.30.168.192.in-addr.arpa domain name pointer sw1-cod.cod.ssa2026.region
[root@srv1-cod ~]# host 192.168.30.2
2.30.168.192.in-addr.arpa domain name pointer sw2-cod.cod.ssa2026.region
[root@srv1-cod ~]# host 192.168.40.40
40.40.168.192.in-addr.arpa domain name pointer cli-cod.cod.ssa2026.region
[root@srv1-cod ~]# host 192.168.10.1
1.10.168.192.in-addr.arpa domain name pointer srv1-cod.cod.ssa2026.region
[root@srv1-cod ~]# host 192.168.10.2
2.10.168.192.in-addr.arpa domain name pointer srv2-cod.cod.ssa2026.region
[root@srv1-cod ~]# host 192.168.50.50
50.50.168.192.in-addr.arpa domain name pointer sip-cod.cod.ssa2026.region
[root@srv1-cod ~]# host 192.168.30.30
30.30.168.192.in-addr.arpa domain name pointer admin-cod.cod.ssa2026.regi

```

sw1-cod и sw2-cod (alt-server):

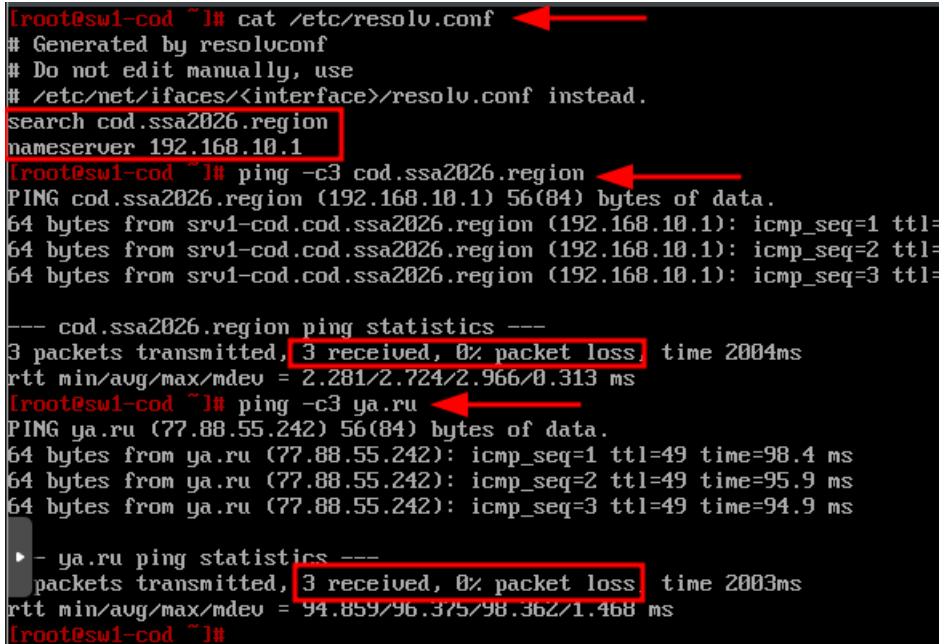
- Задаём в качестве DNS-сервера **srv1-cod**:

```
cat <<EOF > /etc/net/ifaces/mgmt-cod/resolv.conf
search cod.ssa2026.region
nameserver 192.168.10.1
EOF
```

- Перезагружаем службу **network**:

```
systemctl restart network
```

- Проверить:



The terminal screenshot shows the following commands and output:

```
[root@sw1-cod ~]# cat /etc/resolv.conf
# Generated by resolvconf
# Do not edit manually, use
# /etc/net/ifaces/<interface>/resolv.conf instead.
search cod.ssa2026.region
nameserver 192.168.10.1

[root@sw1-cod ~]# ping -c3 cod.ssa2026.region
PING cod.ssa2026.region (192.168.10.1) 56(84) bytes of data.
64 bytes from srv1-cod.cod.ssa2026.region (192.168.10.1): icmp_seq=1 ttl=
64 bytes from srv1-cod.cod.ssa2026.region (192.168.10.1): icmp_seq=2 ttl=
64 bytes from srv1-cod.cod.ssa2026.region (192.168.10.1): icmp_seq=3 ttl=

--- cod.ssa2026.region ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 2.281/2.724/2.966/0.313 ms

[root@sw1-cod ~]# ping -c3 ya.ru
PING ya.ru (77.88.55.242) 56(84) bytes of data.
64 bytes from ya.ru (77.88.55.242): icmp_seq=1 ttl=49 time=98.4 ms
64 bytes from ya.ru (77.88.55.242): icmp_seq=2 ttl=49 time=95.9 ms
64 bytes from ya.ru (77.88.55.242): icmp_seq=3 ttl=49 time=94.9 ms

- ya.ru ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 94.859/96.375/98.362/1.468 ms

[root@sw1-cod ~]#
```

Red boxes highlight the DNS configuration in `/etc/resolv.conf` and the ping statistics for both `cod.ssa2026.region` and `ya.ru`. Red arrows point to the command prompts and the IP address `192.168.10.1`.

srv2-cod (alt-server):

- Задаём в качестве DNS-сервера **srv1-cod**:

```
cat <<EOF > /etc/net/ifaces/ens20/resolv.conf
search cod.ssa2026.region
nameserver 192.168.10.1
EOF
```

- Перезагружаем службу **network**:

```
systemctl restart network
```

- Проверить:

```
[root@sru2-cod ~]# cat etc/resolv.conf
# Generated by resolvconf
# Do not edit manually, use
# /etc/net/ifaces/<interface>/resolv.conf instead.
search cod.ssa2026.region
nameserver 192.168.10.1
[root@sru2-cod ~]# ping -c3 cod.ssa2026.region
PING cod.ssa2026.region (192.168.10.1) 56(84) bytes of data.
64 bytes from sru1-cod.cod.ssa2026.region (192.168.10.1): icmp_seq=1 ttl=
64 bytes from sru1-cod.cod.ssa2026.region (192.168.10.1): icmp_seq=2 ttl=
64 bytes from sru1-cod.cod.ssa2026.region (192.168.10.1): icmp_seq=3 ttl=

--- cod.ssa2026.region ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 1.413/2.001/2.493/0.446 ms
[root@sru2-cod ~]# ping -c3 ya.ru
PING ya.ru (77.88.44.242) 56(84) bytes of data.
64 bytes from ya.ru (77.88.44.242): icmp_seq=1 ttl=47 time=44.4 ms
64 bytes from ya.ru (77.88.44.242): icmp_seq=2 ttl=47 time=43.4 ms
64 bytes from ya.ru (77.88.44.242): icmp_seq=3 ttl=47 time=41.5 ms

--- ya.ru ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 41.497/43.098/44.352/1.191 ms
[root@sru2-cod ~]#
```

Последнее изменение: пятница, 21 ноября 2025, 10:54

Предыдущий
элемент курса

[13. Настройка
синхронизации времени
между сетевыми
устройствами](#)

Перейти на...

Следующий элемент
курса

[15. Развёртывание
контроллера домена на
базе SambaAD](#)

Обратная связь

Вы используете гостевой
доступ (Вход)

Подпишитесь

Сводка хранения данных

Тема оформления сайта разработана

connecti.me

15. Развёртывание контроллера домена на базе SambaAD

[REG26](#)

- 15. Развёртывание
- > контроллера домена на базе SambaAD

Вариант реализации:

dc-a (alt-server):

- Временно для возможности установки необходимых пакетов зададим публичный DNS-сервер:

```
echo "nameserver 77.88.8.8" > /etc/resolv.conf
```

- Устанавливаем пакеты **task-samba-dc**, **bind** и **bind-utils**:

```
apt-get update && apt-get install -y task-samba-dc bind bind-utils
```

- Если при установке системы в настройках сети было указано полное имя домена (например, dc-a.office.ssa2026.region), система может автоматически создать зону office.ssa2026.region, что приведёт к конфликту с Samba при запуске bind
- Для решения проблемы необходимо закомментировать все строки в файле **/etc/bind/local.conf**, это предотвратит автозагрузку конфликтующих зон:

```
//include "/etc/bind/rfc1912.conf";  
// Consider adding the 1918 zones here,  
// if they are not used in your organization.  
//     include "/etc/bind/rfc1918.conf";  
// Add other zones here
```

- Отключите chroot:

```
control bind-chroot disabled
```

- Отключите KRB5RCACHETYPE:

```
echo 'KRB5RCACHETYPE="none"' >> /etc/sysconfig/bind
```

- Подключите плагин BIND_DLZ:

```
echo 'include "/var/lib/samba/bind-dns/named.conf";' >>  
/etc/bind/named.conf
```

- Отредактируйте файл **/etc/bind/options.conf**:
 - в раздел **options** добавьте строки:
 - tkey-gssapi-keytab** — путь к ключевой таблице для GSS-API (интеграция с Kerberos);
 - minimal-responses** — уменьшает объём ответов;
 - listen-on** — IP-адреса, на которых принимаются запросы;
 - allow-query** — разрешённые подсети для DNS-запросов;

- **allow-recursion** — подсети, которым разрешены рекурсивные запросы;
- **forwarders** — внешние DNS-серверы для пересылки;;
- **forward first** — сначала пересылать, затем кешировать;

```
dump-file "/var/run/named/named_dump.db";
statistics-file "/var/run/named/named.stats";
recursing-file "/var/run/named/named.recursing";
secroots-file "/var/run/named/named.secrets";

key-gssapi-keytab "/var/lib/samba/bind-dns/dns.keytab";
minimal-responses yes;

// disables the use of a PID file
pid-file none;

/*
 * Oftenly used directives are listed below.
 */

listen-on { any; };
listen-on-v6 { none; };

/*
 * If the forward directive is set to "only", the server will only
 * query the forwarders.
 */
forward first;
forwarders { 100.100.100.100; };

/*
 * Specifies which hosts are allowed to ask ordinary questions.
 */
allow-query { any; };

/*
 * This lets "allow-query" be used to specify the default zone access
 * level rather than having to have every zone override the global
 * value. "allow-query-cache" can be set at both the options and view
 * levels. If "allow-query-cache" is not set then "allow-recursion"
 * is used if set, otherwise "allow-query" is used if set unless
 * "recursion no;" is set in which case "none;" is used, otherwise
 * default (localhost; localnets;) is used.
 */
allow-query-cache { any; };

/*
 * Specifies which hosts are allowed to make recursive queries
 * through this server. If not specified, the default is to allow
 * recursive queries from all hosts. Note that disallowing recursive
 * queries for a host does not prevent the host from retrieving data
 * that is already in the server's cache.
 */
allow-recursion { any; };
```

- в раздел **logging** добавьте строку:

- **logging** — подавление предупреждений о «lame servers»

```
logging {
    category lame-servers { null; };
    // The default debug channel has the special property that it only
    // produces output when the server's debug level is non-zero. It
    // normally writes to a file called named.run in the server's work
    // directory.
```

- Восстановить к начальному состоянию Samba:

```
rm -f /etc/samba/smb.conf
```

```
rm -f /etc/samba/smb.conf
```

```
rm -rf /var/cache/samba
```

```
mkdir -p /var/lib/samba/sysvol
```

- Интерактивное создание домена:

```
samba-tool domain provision
```

- в качестве **DNS backend** указать **BIND9_DLZ**

```
[root@dc-a ~]# samba-tool domain provision
Realm [OFFICE.SSA2026.REGION]:
Domain [OFFICE]:
Server Role (dc, member, standalone) [dc]:
DNS backend (SAMBA_INTERNAL, BIND9_FLATFILE, BIND9_DLZ, NONE) [SAMBA_INT
Administrator password:
Retype password: _
```

- результат успешного создания домена в интерактивном режиме:

```
INFO 2025-11-19 09:22:18,503 pid:1376 /usr/lib64/samba-dc/python3.12/samba/provision/__init__.py #2414:
9f-9bee195a63ef
INFO 2025-11-19 09:22:18,506 pid:1376 /usr/lib64/samba-dc/python3.12/samba/provision/__init__.py #2425:
generated at /var/lib/samba/private/krb5.conf
INFO 2025-11-19 09:22:18,507 pid:1376 /usr/lib64/samba-dc/python3.12/samba/provision/__init__.py #2427:
nf or replace it with this one. Do not create a symlink!
INFO 2025-11-19 09:22:18,744 pid:1376 /usr/lib64/samba-dc/python3.12/samba/provision/__init__.py #492: D
will be ready to use
INFO 2025-11-19 09:22:18,746 pid:1376 /usr/lib64/samba-dc/python3.12/samba/provision/__init__.py #497: S
INFO 2025-11-19 09:22:18,746 pid:1376 /usr/lib64/samba-dc/python3.12/samba/provision/__init__.py #498: H
INFO 2025-11-19 09:22:18,748 pid:1376 /usr/lib64/samba-dc/python3.12/samba/provision/__init__.py #499: N
INFO 2025-11-19 09:22:18,749 pid:1376 /usr/lib64/samba-dc/python3.12/samba/provision/__init__.py #500: D
INFO 2025-11-19 09:22:18,749 pid:1376 /usr/lib64/samba-dc/python3.12/samba/provision/__init__.py #501: D
2411318
[root@dc-a ~]#
```

- Включить в автозагрузку службы **samba** и **bind**, также запустить их:

```
systemctl enable --now samba
```

```
systemctl enable --now bind
```

- При создании домена Samba автоматически генерирует корректный файл **krb5.conf** для домена в каталоге **/var/lib/samba/private/**
 - Можно просто заменить этим файлом файл, находящийся в каталоге **/etc/**:

```
cp /var/lib/samba/private/krb5.conf /etc/krb5.conf
```

- Перезагрузить службу **samba**:

```
systemctl restart samba
```

- Просмотр общей информации о домене можно выполнить с помощью команды **samba-tool domain info 127.0.0.1**:

```
[root@dc-a ~]# samba-tool domain info 127.0.0.1
Forest      : office.ssa2026.region
Domain     : office.ssa2026.region
Netbios domain : OFFICE
DC name    : dc-a.office.ssa2026.region
DC netbios name : DC-A
Server site : Default-First-Site-Name
Client site : Default-First-Site-Name
[root@dc-a ~]#
```

- Просмотр предоставляемых служб можно выполнить с помощью команды **smbclient -L localhost -Uadministrator**:
 - **netlogon** и **sysvol** создаются автоматически и необходимы для работы контроллера домена

```
[root@dc-a ~]# smbclient -L localhost -U administrator
Password for [OFFICE\administrator]:

      Sharename      Type      Comment
      -----
      sysvol          Disk
      netlogon         Disk
      IPC$            IPC        IPC Service (Samba 4.21.9-alt1)
SMB1 disabled -- no workgroup available
[root@dc-a ~]#
```

- Проверка конфигурации DNS:
 - проверка наличия `nameserver 127.0.0.1` в `/etc/resolv.conf`:

```
[root@dc-a ~]# cat /etc/resolv.conf
# Generated by resolvconf
# Do not edit manually, use
# /etc/net/ifaces/<interface>/resolv.conf instead.
search office.ssa2026.region
nameserver 127.0.0.1
[root@dc-a ~]#
```

- проверка имён хостов `"_kerberos._udp."`:

```
[root@dc-a ~]# host -t SRV _kerberos._udp.office.ssa2026.region.
_kerberos._udp.office.ssa2026.region has SRV record 0 100 88 dc-a.office.ssa2026.region.
[root@dc-a ~]#
```

- проверка имён хостов `"_ldap._tcp."`:

```
[root@dc-a ~]# host -t SRV _ldap._tcp.office.ssa2026.region.
_ldap._tcp.office.ssa2026.region has SRV record 0 100 389 dc-a.office.ssa2026.region.
[root@dc-a ~]#
```

- проверка имён хостов `"адрес хоста."`:

```
[root@dc-a ~]# host -t A dc-a.office.ssa2026.region.
dc-a.office.ssa2026.region has address 172.20.10.10
[root@dc-a ~]#
```

- Проверка Kerberos-аутентификации (имя домена должно быть в верхнем регистре):

```
[root@dc-a ~]# kinit administrator@OFFICE.SSA2026.REGION
Password for administrator@OFFICE.SSA2026.REGION:
Warning: Your password will expire in 41 days on Wed Dec 31 09:22:12 2025
[root@dc-a ~]#
```

- Просмотр полученного билета:

```
[root@dc-a ~]# klist
Ticket cache: FILE:/tmp/krb5cc_0
Default principal: administrator@OFFICE.SSA2026.REGION

Valid starting    Expires          Service principal
11/19/25 09:49:33 11/19/25 19:49:33 krbtgt/OFFICE.SSA2026.REGION@OFFICE.SSA2026.REGION
renew until 11/20/25 09:45:24
[root@dc-a ~]#
```

cli1-a и cli2-a (alt-workstation):

- Для ввода в домен установим пакет `task-auth-ad-sssd`:

```
apt-get update && apt-get install -y task-auth-ad-sssd
```

- Для ввода компьютера в домен в **ЦУС** необходимо выбрать пункт **Пользователи → Аутентификация**
- В окне модуля **Аутентификация** следует выбрать пункт **Домен Active Directory**, заполнить поля (Домен, Рабочая группа, Имя компьютера), выбрать пункт SSSD (в единственном домене) и нажать кнопку **Применить**

Аутентификация

☐ Локальная база пользователей

☐ Домен ALT Linux или Astra Linux Directory

Домен:

☐ Кэшировать аутентификацию при недоступности сервера домена

☒ Домен Active Directory

Домен:

Рабочая группа:

Имя компьютера:

☒ SSSD (в единственном домене)

☐ Winbind (в сложных доменах)

☐ Домен FreeIPA

Внимание: Не установлен пакет task-auth-freeipa. Аутентификация в домене FreeIPA не работает.

Домен:

Имя компьютера:

Внимание!

Изменение домена заработает только после перезагрузки компьютера

☐ Восстановить файлы конфигурации по умолчанию (smb.conf, krb5.conf, sssd.conf).

- В открывшемся окне необходимо ввести имя пользователя, имеющего право вводить машины в домен, и его пароль и нажать кнопку **ОК**:

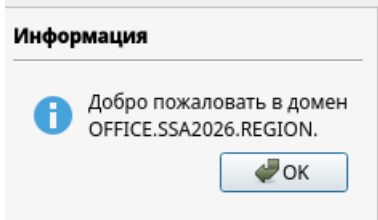
Введите пароль для учётной записи с правами подключения к домену.

Имя пользователя:

Пароль:

☐ Использовать уже полученный билет Kerberos

- Результат успешного присоединения к домену:



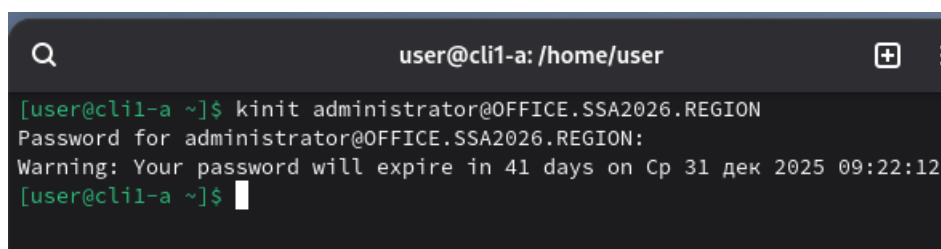
- Перезагрузить рабочую станцию для применения всех настроек

cli1-a (alt-workstation):

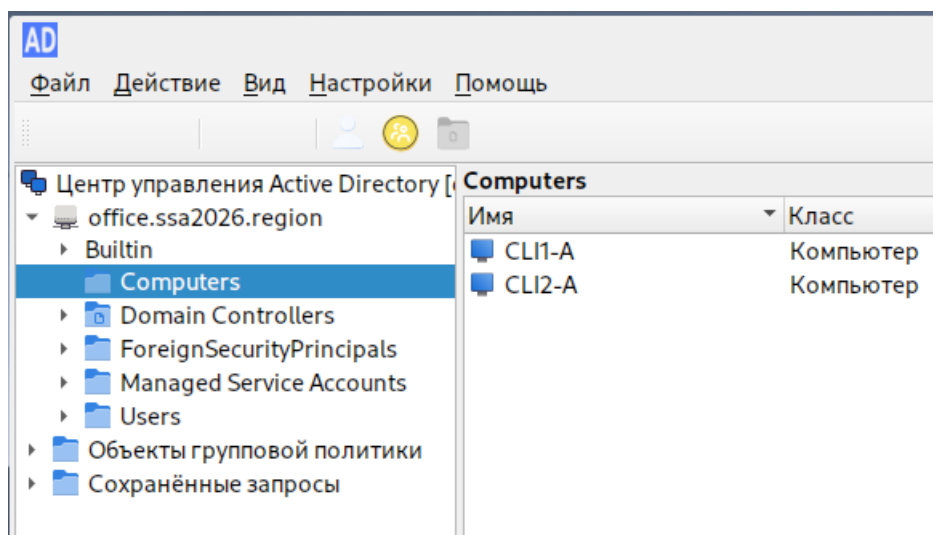
- Установить пакет **admc**:

```
apt-get install -y admc
```

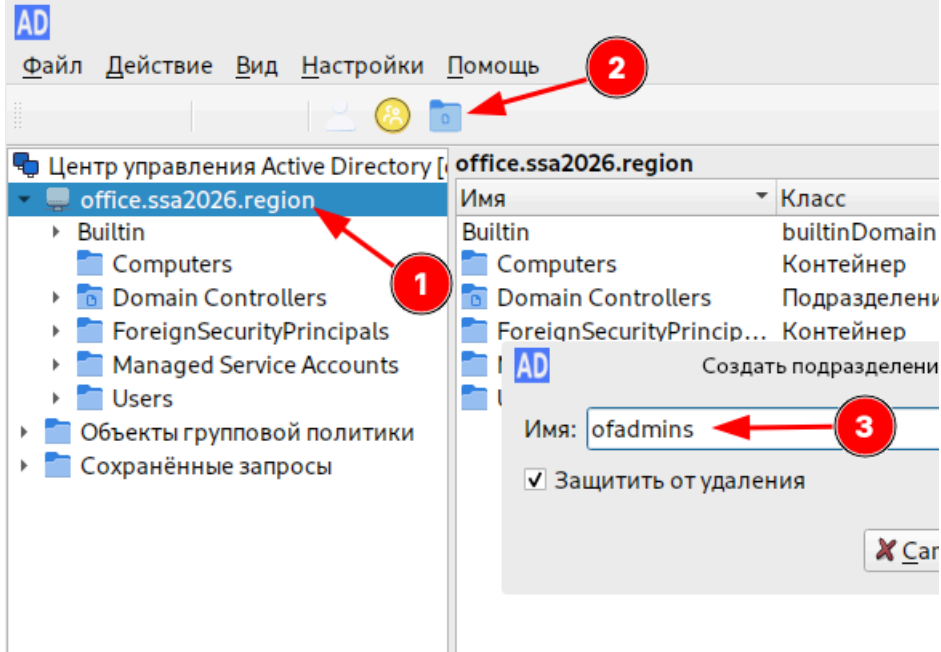
- Запуск **ADMC** осуществляется из меню запуска приложений: пункт **Системные** → **ADMC** или из командной строки (команда **admc**)
- Для запуска ADCM необходимо предварительно получить ключ Kerberos для администратора домена:



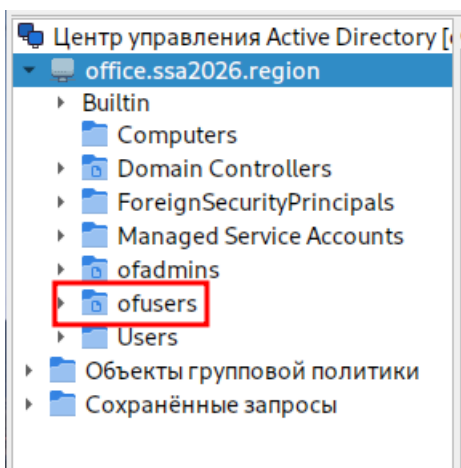
- Результат успешного запуска **admc**:



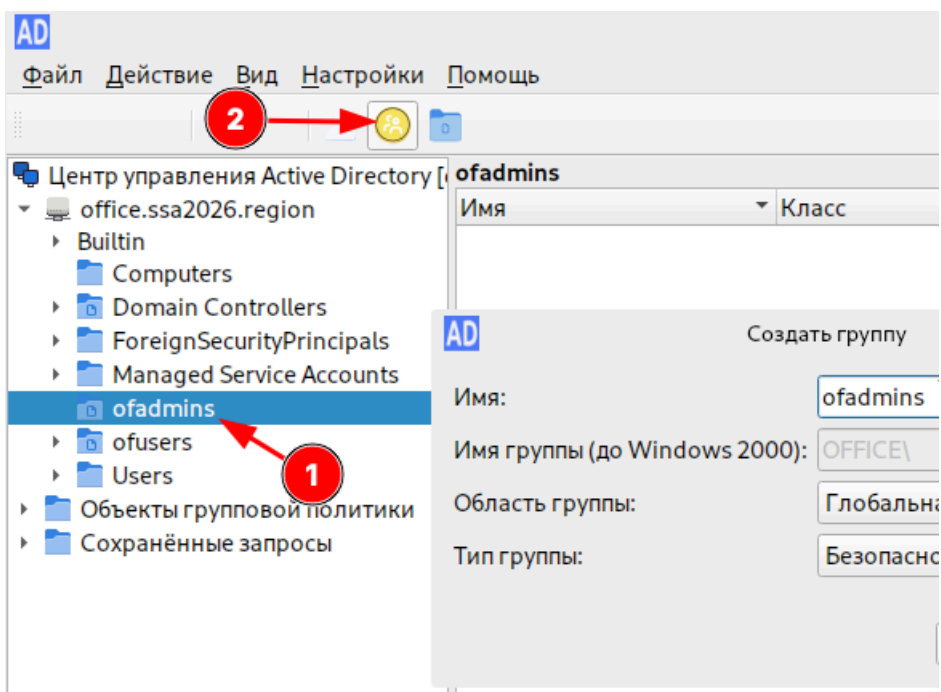
- Создать **ofadmins**:



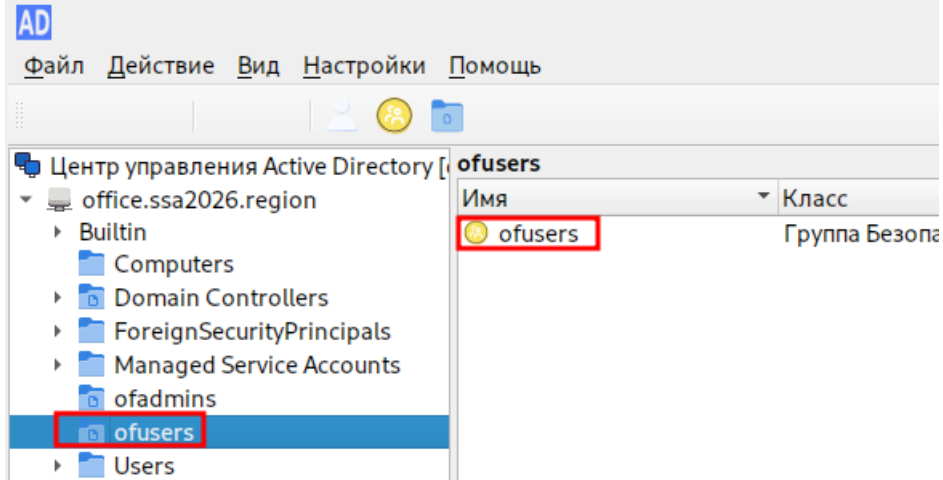
- Создать **ofusers**:



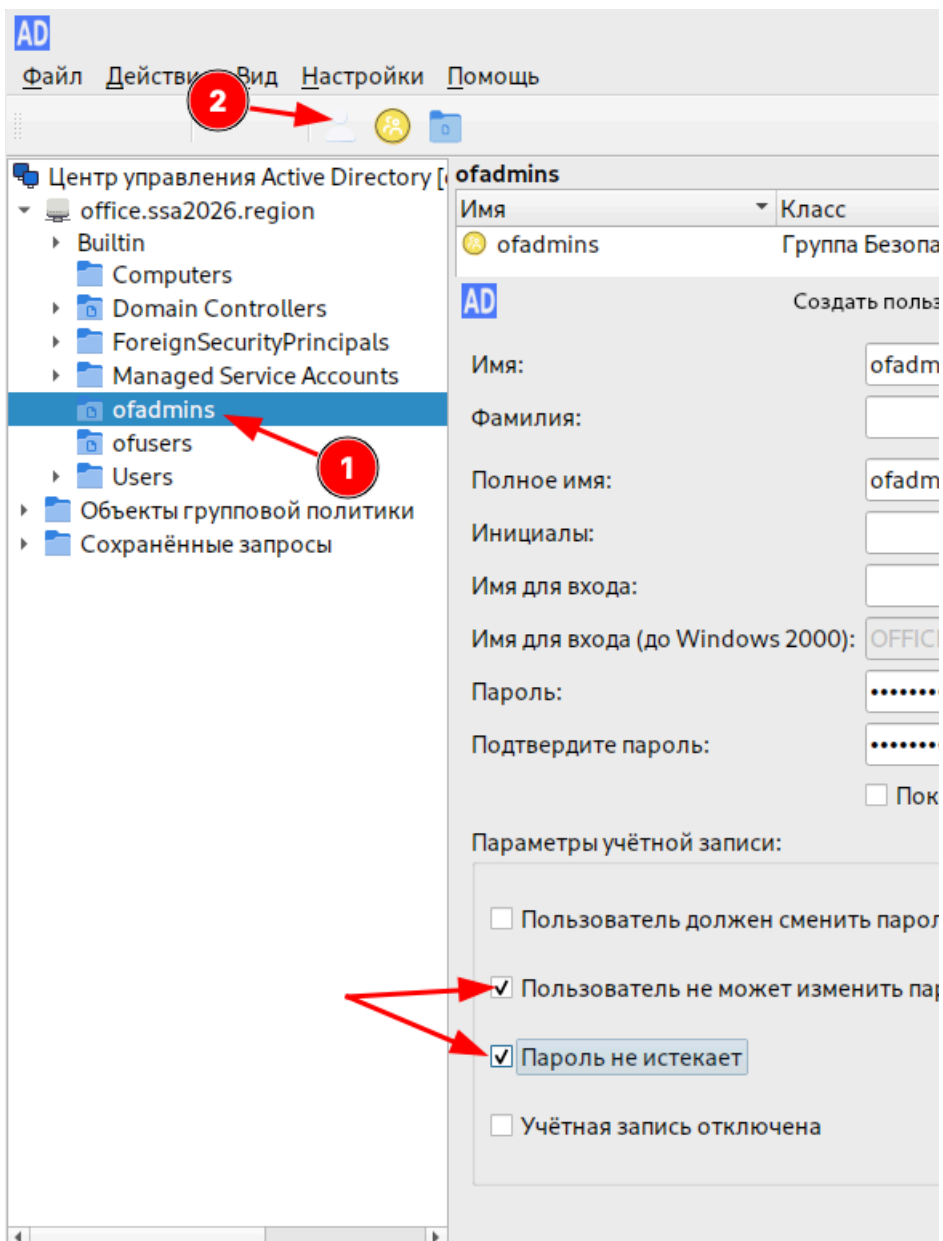
- Создать группу **ofadmins** в подразделение **ofadmins**:



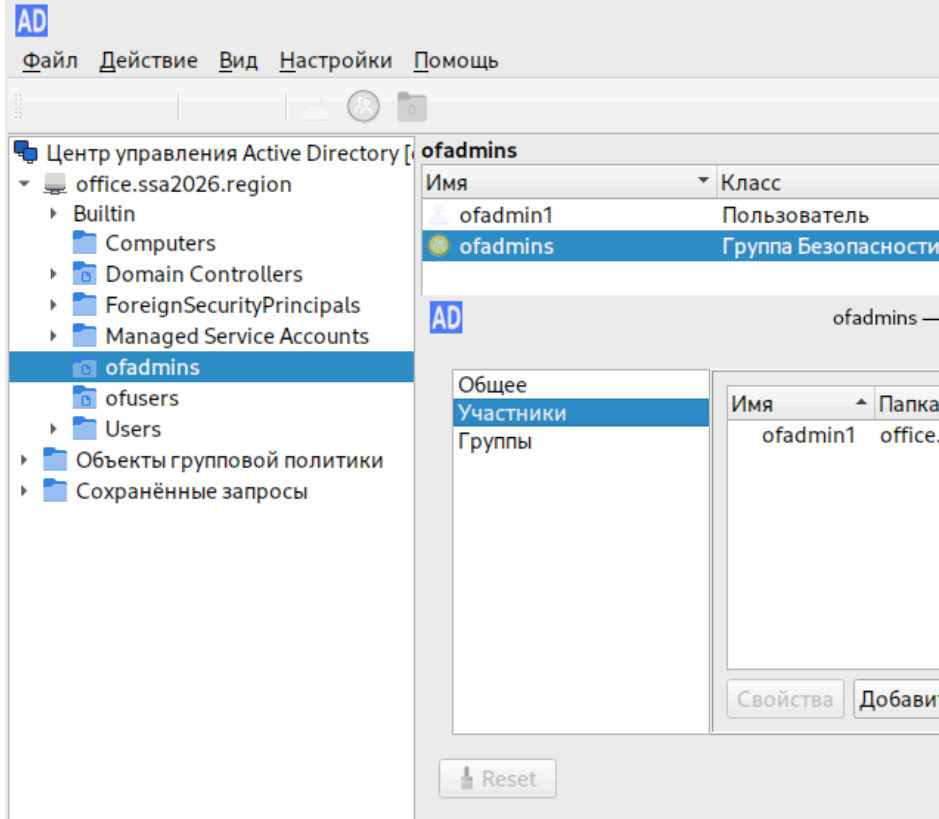
- Создать группу **ofusers** в подразделение **ofusers**:



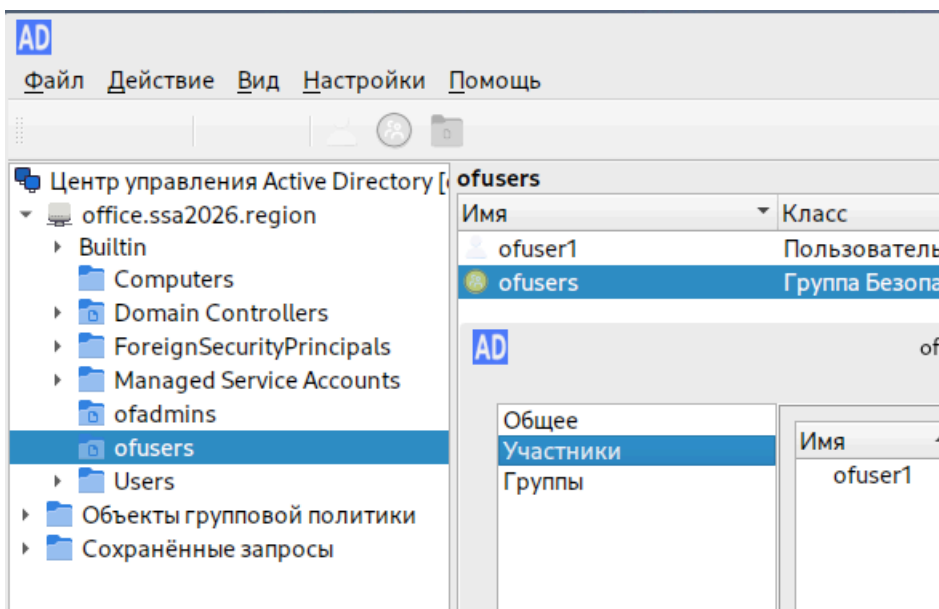
- Создать пользователя **ofadmin1** в подразделение **ofadmins**:



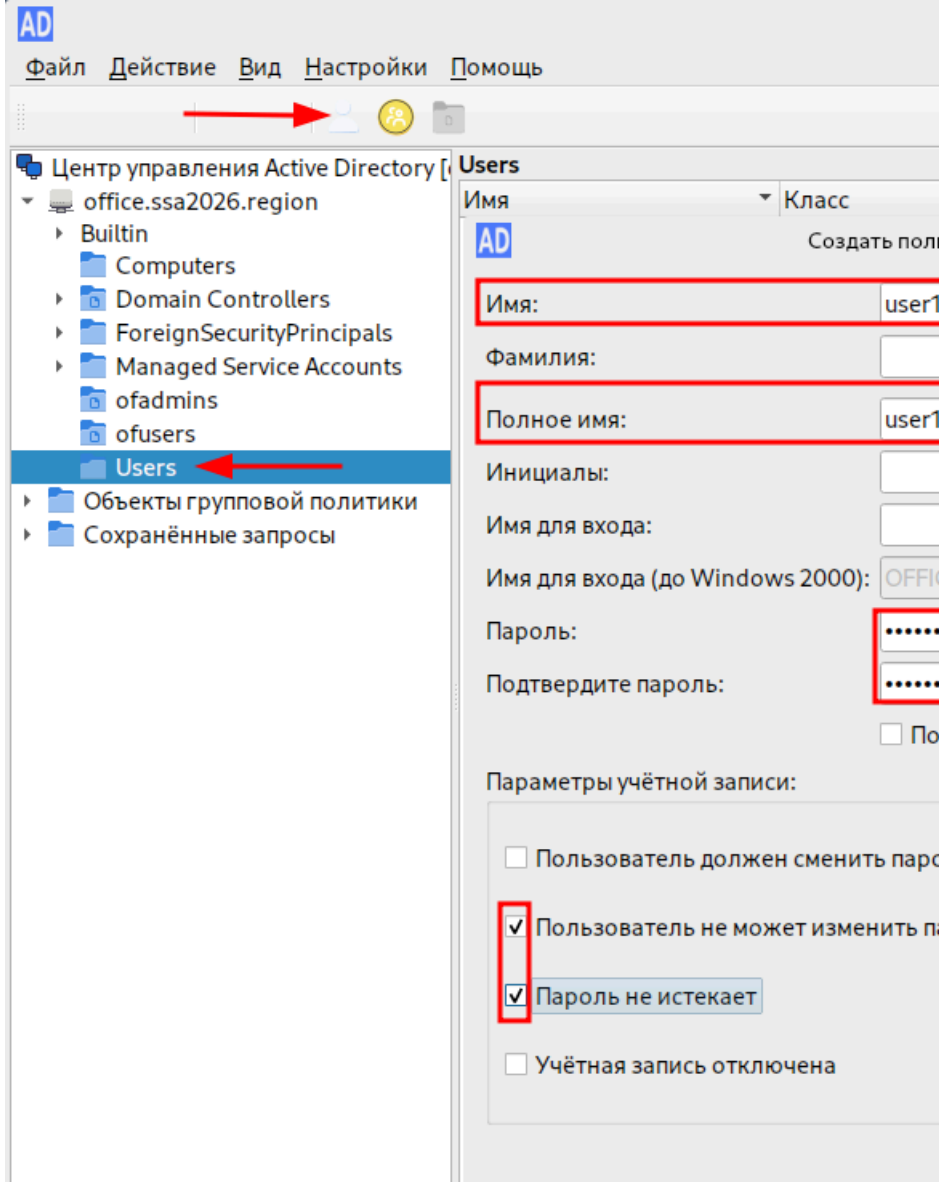
- Добавить созданного пользователя в группу **ofadmins**:



- Создать пользователя **ofuser1** в подразделение **ofusers**
 - Добавить созданного пользователя в группу **ofusers**



- Создать пользователя **user1**:



cli2-a или cli1-a (alt-workstation):

- Проверить возможность входа из-под созданных пользователей:
 - ofadmin1:

```
ofadmin1@cli2-a: /home/OFFICE.SSA2026.REGION/ofadmin1
[ofadmin1@cli2-a ~]$ id
uid=29801108(ofadmin1) gid=29800513(domain users) группы=29800513(domain users),29801108(ofadmin1),29801109(ofusers),29801110(ofadmins),29801111(ofadmins),29801112(ofadmins),29801113(ofadmins),29801114(uucp),19(proc),22(cdrom),71(floppy),80(cdwriter),81(audio),83(radio),100(disk),951(usershares),959(camera),965(fuse),978(video),988(vboxsf),989(vboxsf),990(xgrp),998(scanner),29801106(ofadmins)
[ofadmin1@cli2-a ~]$
```

- ofuser1:

```
ofuser1@cli2-a: /home/OFFICE.SSA2026.REGION/ofuser1
[ofuser1@cli2-a ~]$ id
uid=29801109(ofuser1) gid=29800513(domain users) группы=29800513(domain users),29801108(ofadmin1),29801109(ofuser1),29801110(ofadmins),29801111(ofadmins),29801112(ofadmins),29801113(ofadmins),29801114(uucp),19(proc),22(cdrom),71(floppy),80(cdwriter),81(audio),83(radio),100(disk),951(usershares),959(camera),965(fuse),978(video),988(vboxsf),989(vboxsf),990(xgrp),998(scanner),29801107(ofusers)
[ofuser1@cli2-a ~]$
```

- user1:

```
user1@cli2-a /home/OFFICE.SSA2026.REGION/user1
[user1@cli2-a ~]$ id
uid=29801110(user1) gid=29800513(domain users) группы=29800513(domain us
uucp),19(proc),22(cdrom),71(floppy),80(cdwriter),81(audio),83(radio),100
951(usershares),959(camera),965(fuse),978(video),988(vboxsf),989(vboxadd
rp),998(scanner)
[user1@cli2-a ~]$
```

cli1-a и cli2-a (alt-workstation):

- Необходимо установить пакет **gpubdate**:

```
apt-get install -y gpubdate
```

- Включить модуль групповых политик:

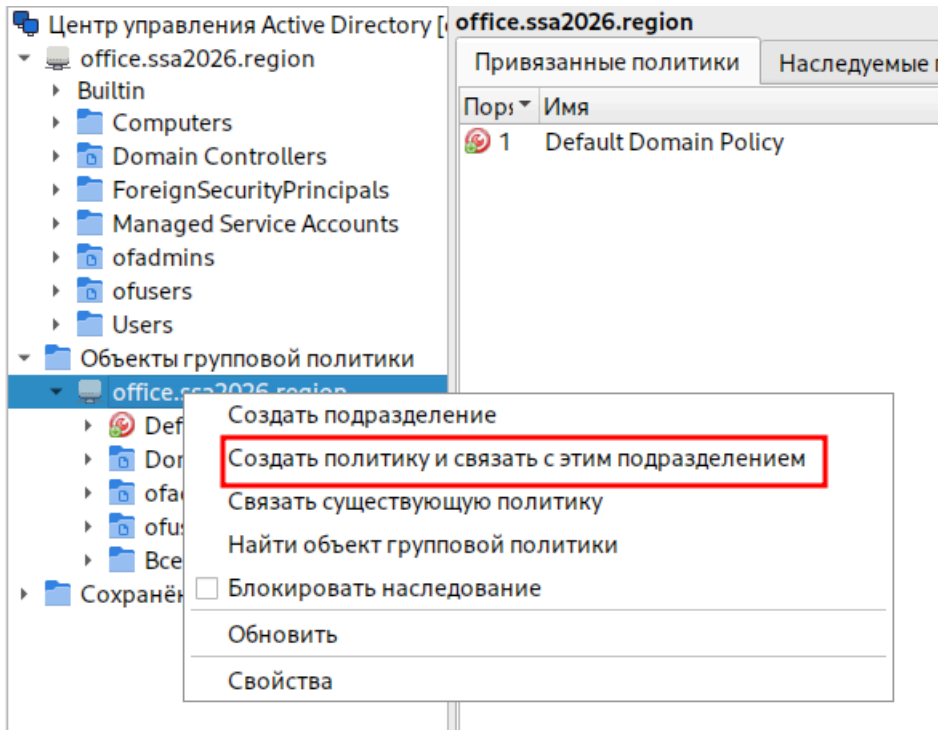
```
gpubdate-setup enable
```

cli1-a (alt-workstation):

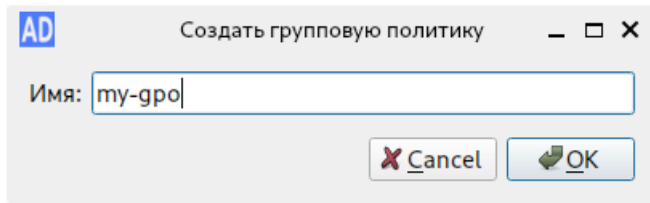
- Установим пакет **gpui**, для редактирования настроек клиентской конфигурации:

```
apt-get install -y gpui
```

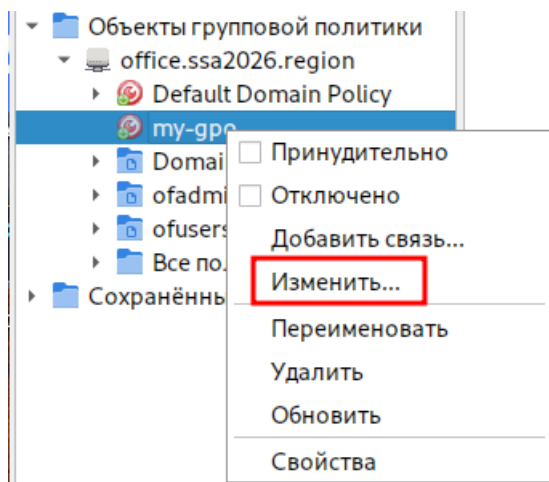
- В оснастке ADMC - переходим в раздел **Объекты групповой политики -> ПКМ по office.ssa2026.region** и нажимаем **Создать политику и связать с этим подразделением**:



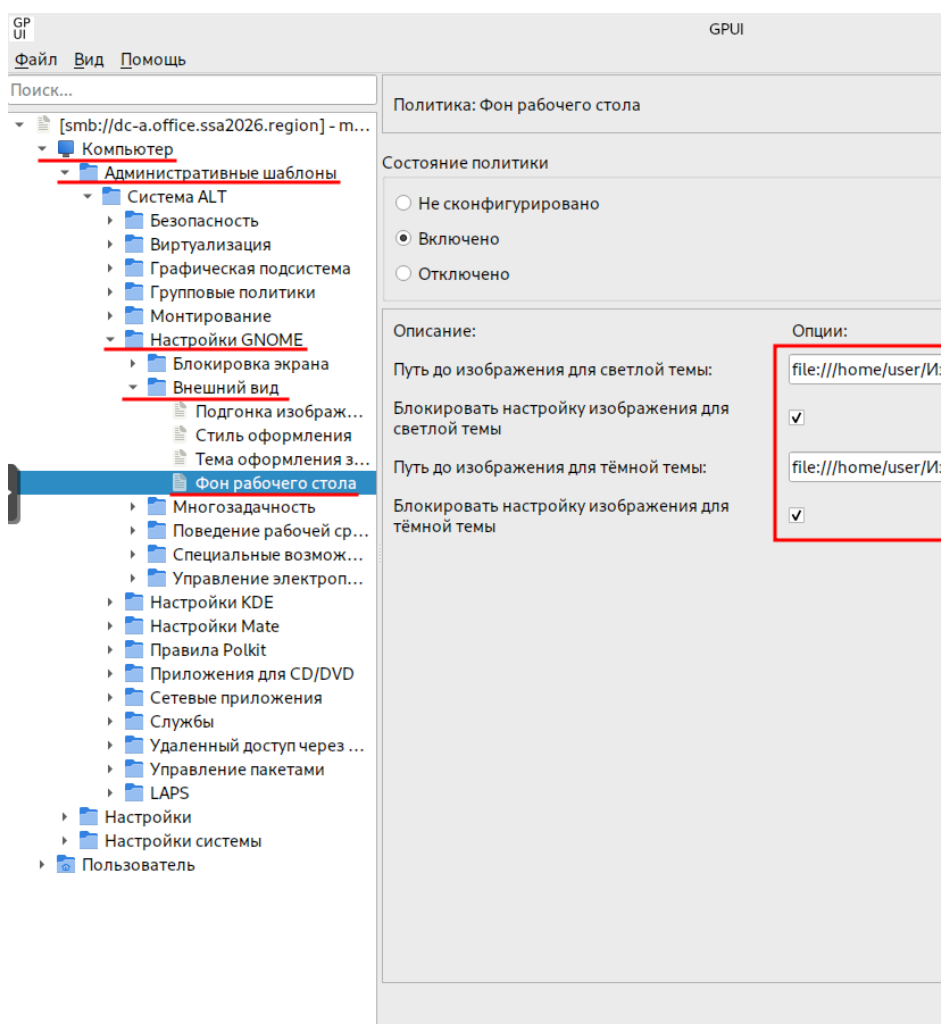
- Задаём имя и нажимаем **ОК**:



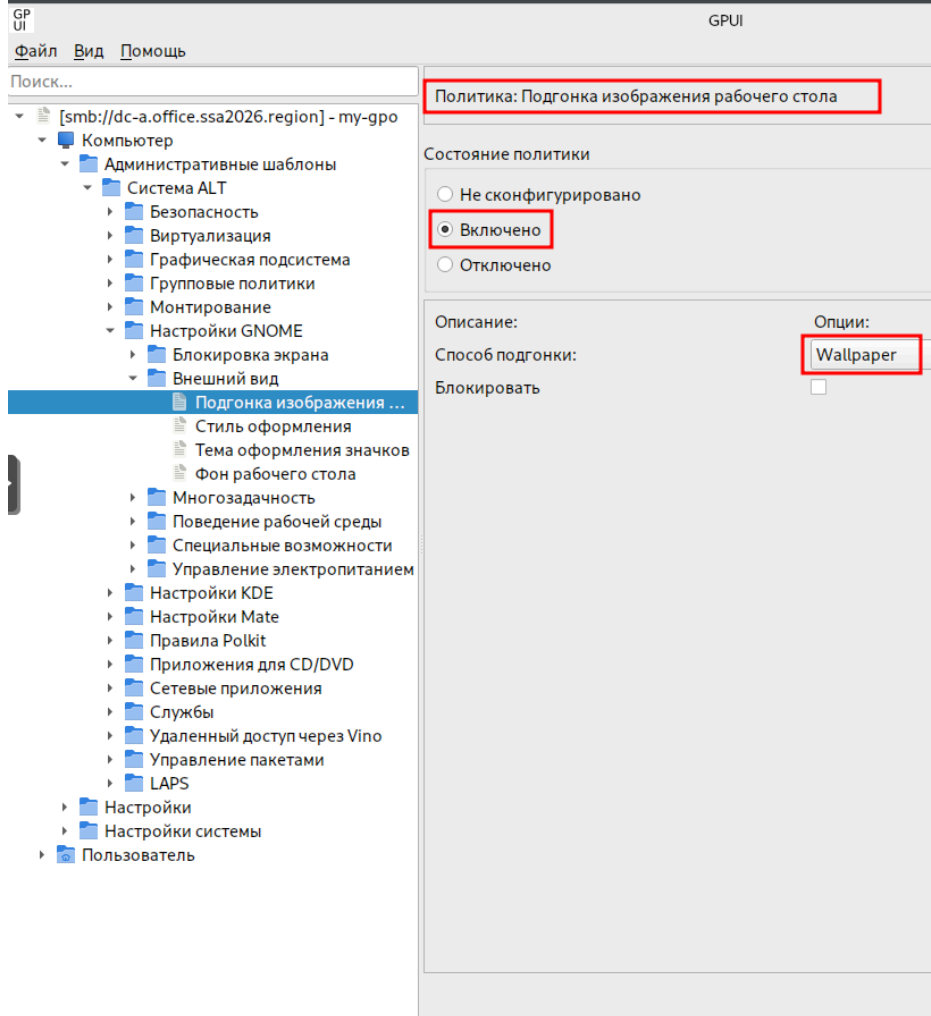
- Выбираем созданную групповую политику **ПКМ** и нажимаем **Изменить**:



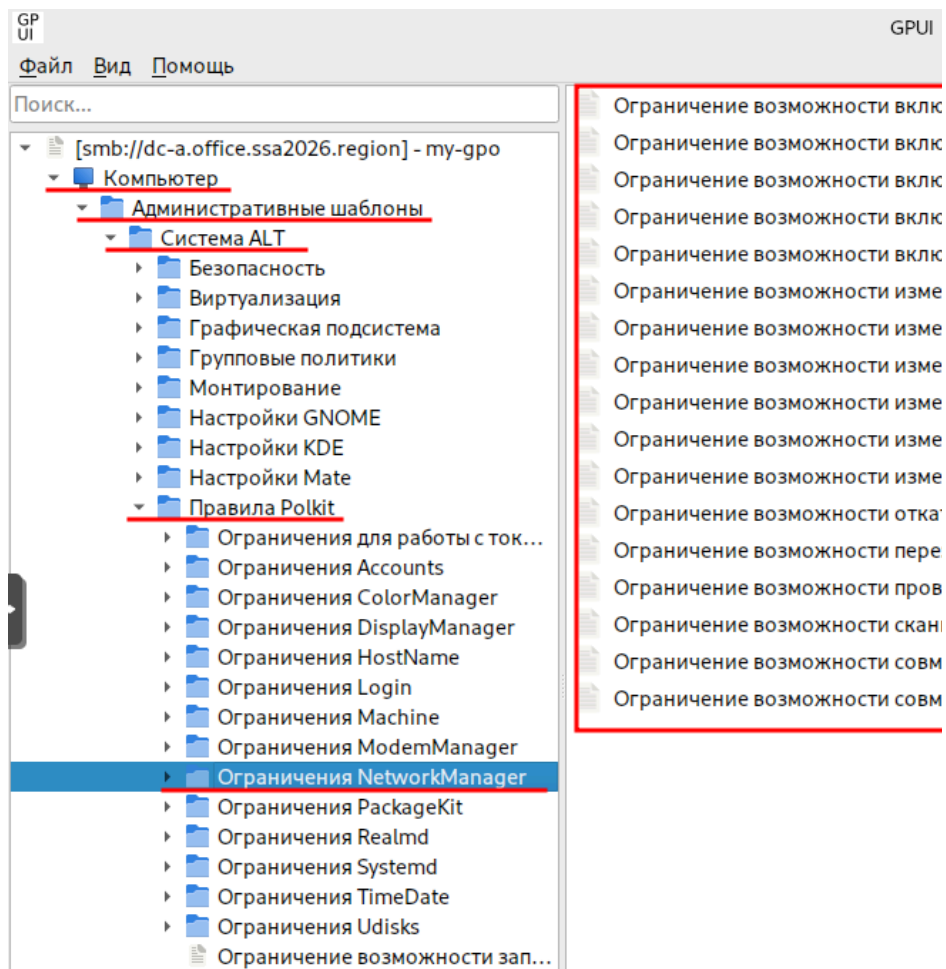
- В соответствии с требованиями задания - реализуем необходимый функционал:
 - Задаём картинку компании для рабочего стола и запрещаем её менять
 - но, чтобы на **cli2-a** (или же из под любого другого пользователя) картинка также была установлена, её необходимо разместить на общем ресурсе (например, создав общую папку)



- опционально (необязательно):

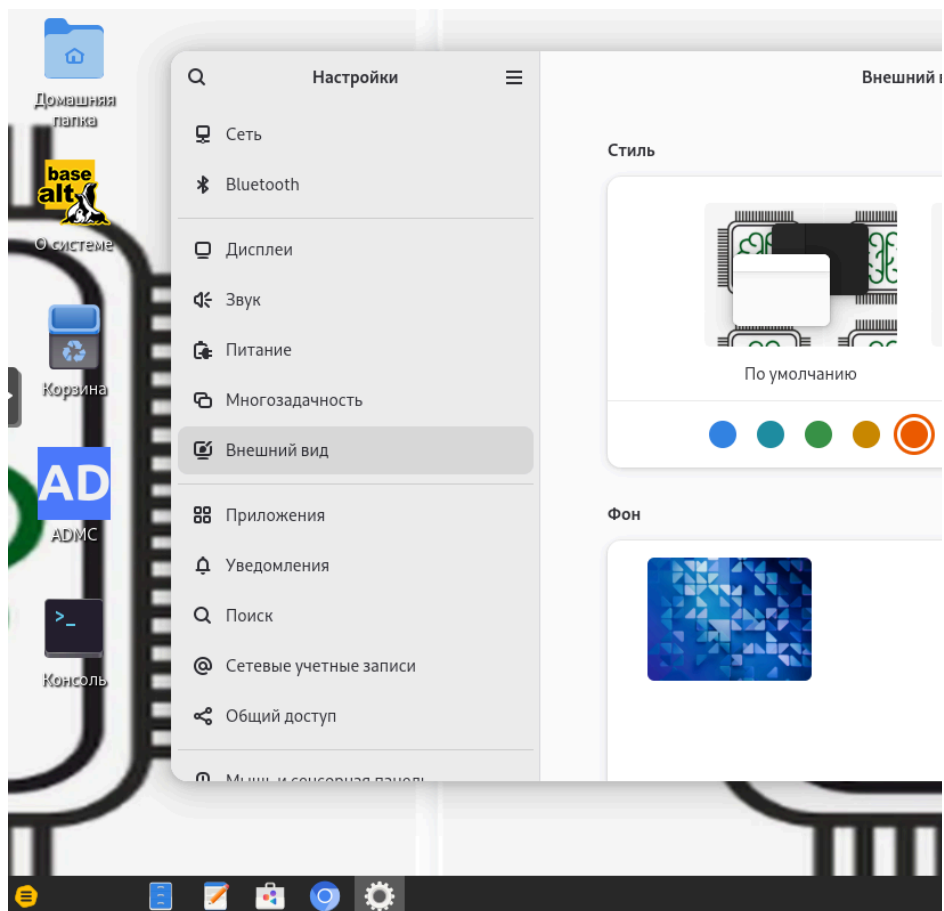


- Запрещаем изменение сетевых настроек:
 - пройтись по всему списку и выставить **Включено**, с вариантом ограничений **No** (выставить чек-бокс **Блокировать**):

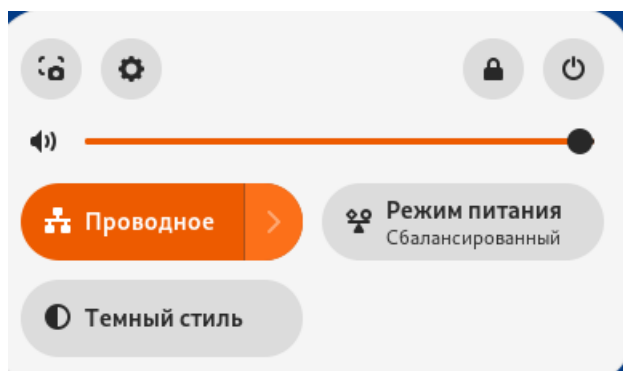


- Проверить, перезагружаем **cli1-a** и **cli2-a**:

- картинка фона рабочего стола должна быть установлена:



- при попытке отключить сетевой интерфейс, данная возможность отсутствует:



Последнее изменение: среда, 19 ноября 2025, 14:16

Предыдущий
элемент курса

[14. Настройка службы
доменных имен в COD-e](#)

Перейти на...

Следующий элемент
курса

[16. Настройка службы
доменных имен в OFFICE](#)

Обратная связь

Подпишитесь

Вы используете гостевой
доступ (Вход)

Сводка хранения данных

Тема оформления сайта разработана

16. Настройка службы доменных имен в OFFICE

[REG26](#)

> 16. Настройка службы доменных имен в OFFICE

Вариант реализации:

dc-a (alt-server):

- Средствами **samba-tool** можно посмотреть список зон DNS:

```
[root@dc-a ~]# samba-tool dns zonelist 127.0.0.1
Password for [administrator@OFFICE.SSA2026.REGION1]:
2 zone(s) found

pszZoneName      : office.ssa2026.region
Flags            : DNS_RPC_ZONE_DSINTEGRATED DNS_RPC_ZONE_U
ZoneType         : DNS_ZONE_TYPE_PRIMARY
Version          : 50
dwDpFlags        : DNS_DP_AUTOCREATED DNS_DP_DOMAIN_DEFAULT
pszDpFqdn        : DomainDnsZones.office.ssa2026.region

pszZoneName      : _msdcs.office.ssa2026.region
Flags            : DNS_RPC_ZONE_DSINTEGRATED DNS_RPC_ZONE_U
ZoneType         : DNS_ZONE_TYPE_PRIMARY
Version          : 50
dwDpFlags        : DNS_DP_AUTOCREATED DNS_DP_FOREST_DEFAULT
pszDpFqdn        : ForestDnsZones.office.ssa2026.region
[root@dc-a ~]#
```

- Средствами **samba-tool** можно посмотреть список DNS-записей в определённой зоне:

```
[root@dc-a ~]# samba-tool dns query 127.0.0.1 office.ssa2026.region @ A
Password for [administrator@OFFICE.SSA2026.REGION1]:
Name=, Records=1, Children=0
  A: 172.20.10.10 (flags=600000f0, serial=38, ttl=900)
Name=_msdcs, Records=0, Children=0
Name=_sites, Records=0, Children=1
Name=_tcp, Records=0, Children=4
Name=_udp, Records=0, Children=2
Name=cli1-a, Records=1, Children=0
  A: 172.20.20.1 (flags=f0, serial=39, ttl=3600)
Name=cli2-a, Records=1, Children=0
  A: 172.20.20.2 (flags=f0, serial=39, ttl=3600)
Name=dc-a, Records=1, Children=0
  A: 172.20.10.10 (flags=f0, serial=1, ttl=900)
Name=DomainDnsZones, Records=0, Children=2
Name=ForestDnsZones, Records=0, Children=2
[root@dc-a ~]#
```

- Средствами **samba-tool** создадим не достающие записи типа **A** в зоне прямого просмотра:

```
samba-tool dns add 127.0.0.1 office.ssa2026.region rtr-a A
172.20.10.254 -U administrator
samba-tool dns add 127.0.0.1 office.ssa2026.region rtr-a A
172.20.20.254 -U administrator
samba-tool dns add 127.0.0.1 office.ssa2026.region rtr-a A
172.20.30.254 -U administrator
samba-tool dns add 127.0.0.1 office.ssa2026.region sw1-a A
172.20.30.1 -U administrator
samba-tool dns add 127.0.0.1 office.ssa2026.region sw2-a A
172.20.30.2 -U administrator
```

- Проверить наличие записей:

```
[root@dc-a ~]# samba-tool dns query 127.0.0.1 office.ssa2026.region @ A
Password for [administrator@OFFICE.SSA2026.REGION]:
Name=, Records=1, Children=0
  A: 172.20.10.10 (flags=600000f0, serial=38, ttl=900)
Name=_msdcs, Records=0, Children=0
Name=_sites, Records=0, Children=1
Name=_tcp, Records=0, Children=4
Name=_udp, Records=0, Children=2
Name=cli1-a, Records=1, Children=0
  A: 172.20.20.1 (flags=f0, serial=39, ttl=3600)
Name=cli2-a, Records=1, Children=0
  A: 172.20.20.2 (flags=f0, serial=39, ttl=3600)
Name=dc-a, Records=1, Children=0
  A: 172.20.10.10 (flags=f0, serial=1, ttl=900)
Name=DomainDnsZones, Records=0, Children=2
Name=ForestDnsZones, Records=0, Children=2
Name=rtr-a, Records=3, Children=0
  A: 172.20.10.254 (flags=f0, serial=40, ttl=900)
  A: 172.20.20.254 (flags=f0, serial=41, ttl=900)
  A: 172.20.30.254 (flags=f0, serial=42, ttl=900)
Name=sw1-a, Records=1, Children=0
  A: 172.20.30.1 (flags=f0, serial=43, ttl=900)
Name=sw2-a, Records=1, Children=0
  A: 172.20.30.2 (flags=f0, serial=44, ttl=900)
[root@dc-a ~]#
```

- Проверить функционально:

```
[root@dc-a ~]# host rtr-a
rtr-a.office.ssa2026.region has address 172.20.20.254
rtr-a.office.ssa2026.region has address 172.20.30.254
rtr-a.office.ssa2026.region has address 172.20.10.254
[root@dc-a ~]# host sw1-a
sw1-a.office.ssa2026.region has address 172.20.30.1
[root@dc-a ~]# host sw2-a
sw2-a.office.ssa2026.region has address 172.20.30.2
[root@dc-a ~]# host dc-a
dc-a.office.ssa2026.region has address 172.20.10.10
[root@dc-a ~]# host cli1-a
cli1-a.office.ssa2026.region has address 172.20.20.1
[root@dc-a ~]# host cli2-a
cli2-a.office.ssa2026.region has address 172.20.20.2
[root@dc-a ~]# _
```

- Добавить в конфигурационный файл `/etc/bind/local.conf` информацию о файле зоны обратного просмотра и о зонах на `srv1-cod`:

```
//include "/etc/bind/rfc1912.conf";

// Consider adding the 1918 zones here,
// if they are not used in your organization.
//   include "/etc/bind/rfc1918.conf";

// Add other zones here
zone "20.172.in-addr.arpa" {
    type master;
    file "20.172.in-addr.arpa";
    allow-transfer { 192.168.10.1; };
};

zone "cod.ssa2026.region" {
    type forward;
    forward only;
    forwarders { 192.168.10.1; };
};

zone "168.192.in-addr.arpa" {
    type forward;
    forward only;
    forwarders { 192.168.10.1; };
};
```

- Скопировать файл шаблона для зоны обратного просмотра:

```
cp /etc/bind/zone/localhost /etc/bind/zone/20.172.in-addr.arpa
```

- Выдать права на файл зоны обратного просмотра:

```
chown root:named /etc/bind/zone/20.172.in-addr.arpa
```

- Привести файл `/etc/bind/zone/20.172.in-addr.arpa` зоны обратного просмотра к следующему виду:

```
$TTL      1D
;
; IN      SOA      office.ssa2026.region. root.office.ssa2026.region.
;                2025110500      ; serial
;                12H              ; refresh
;                1H               ; retry
;                1W               ; expire
;                1H               ; ncache
;
; )
;
; IN      NS       office.ssa2026.region.
254.10    IN      PTR    rtr-a.office.ssa2026.region.
254.20    IN      PTR    rtr-a.office.ssa2026.region.
254.30    IN      PTR    rtr-a.office.ssa2026.region.
1.30      IN      PTR    sw1-a.office.ssa2026.region.
2.30      IN      PTR    sw2-a.office.ssa2026.region.
10.10     IN      PTR    dc-a.office.ssa2026.region.
1.20      IN      PTR    cli1-a.office.ssa2026.region.
2.20      IN      PTR    cli2-a.office.ssa2026.region.
```

- Перезапустить службу `bind`:

```
systemctl restart bind
```

- Проверить записи типа **PTR**:

```
[root@dc-a ~]# host 172.20.10.254
254.10.20.172.in-addr.arpa domain name pointer rtr-a.office.ssa2026.region.
[root@dc-a ~]# host 172.20.20.254
254.20.20.172.in-addr.arpa domain name pointer rtr-a.office.ssa2026.region.
[root@dc-a ~]# host 172.20.30.254
254.30.20.172.in-addr.arpa domain name pointer rtr-a.office.ssa2026.region.
[root@dc-a ~]# host 172.20.30.1
1.30.20.172.in-addr.arpa domain name pointer sw1-a.office.ssa2026.region.
[root@dc-a ~]# host 172.20.30.2
2.30.20.172.in-addr.arpa domain name pointer sw2-a.office.ssa2026.region.
[root@dc-a ~]# host 172.20.20.1
1.20.20.172.in-addr.arpa domain name pointer cli1-a.office.ssa2026.region.
[root@dc-a ~]# host 172.20.20.2
2.20.20.172.in-addr.arpa domain name pointer cli2-a.office.ssa2026.region.
[root@dc-a ~]# host 172.20.10.10
10.10.20.172.in-addr.arpa domain name pointer dc-a.office.ssa2026.region.
[root@dc-a ~]# _
```

- Проверить несколько записей для forward зон:

```
[root@dc-a ~]# host 192.168.10.1
1.10.168.192.in-addr.arpa domain name pointer srv1-cod.cod.ssa2026.region.
[root@dc-a ~]# host 192.168.10.2
2.10.168.192.in-addr.arpa domain name pointer srv2-cod.cod.ssa2026.region.
[root@dc-a ~]# _
```

sw1-a и sw2-a (alt-server):

- Задаём в качестве DNS-сервера **dc-a**:

```
cat <<EOF > /etc/net/ifaces/mgmt/resolv.conf
search office.ssa2026.region
nameserver 172.20.10.10
EOF
```

- Перезагружаем службу **network**:

```
systemctl restart network
```

- Проверить:

```
[root@sw1-a ~]# cat /etc/resolv.conf
# Generated by resolvconf
# Do not edit manually, use
# /etc/net/ifaces/<interface>/resolv.conf instead.
search office.ssa2026.region
nameserver 172.20.10.10
[root@sw1-a ~]# ping -c3 office.ssa2026.region
PING office.ssa2026.region (172.20.10.10) 56(84) bytes of data.
64 bytes from dc-a.office.ssa2026.region (172.20.10.10): icmp_seq=1 ttl=64
64 bytes from dc-a.office.ssa2026.region (172.20.10.10): icmp_seq=2 ttl=64
64 bytes from dc-a.office.ssa2026.region (172.20.10.10): icmp_seq=3 ttl=64

--- office.ssa2026.region ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 73.257/76.143/77.987/2.066 ms
[root@sw1-a ~]# ping -c3 ya.ru
PING ya.ru (77.88.55.242) 56(84) bytes of data.
64 bytes from ya.ru (77.88.55.242): icmp_seq=1 ttl=48 time=97.3 ms
64 bytes from ya.ru (77.88.55.242): icmp_seq=2 ttl=48 time=95.8 ms
64 bytes from ya.ru (77.88.55.242): icmp_seq=3 ttl=48 time=92.7 ms

--- ya.ru ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 92.658/95.265/97.336/1.947 ms
[root@sw1-a ~]# _
```

Последнее изменение: среда, 19 ноября 2025, 15:59

Предыдущий
элемент курса

[15. Развёртывание
контроллера домена на
базе SambaAD](#)

Перейти на...

Следующий элемент
курса

[17. Настройка центра
сертификации](#)

Обратная связь

Подпишитесь

Вы используете гостевой
доступ (Вход)

Сводка хранения данных

Тема оформления сайта разработана

conecti.me

17. Настройка центра сертификации

REG26

> 17. Настройка центра
сертификации

Вариант реализации:

srv1-cod (alt-server):

- Создадим директорию `/var/ca`:

```
mkdir /var/ca
```

- Отредактируем конфигурационный файл `/etc/openssl/openssl.cnf`:

```
#####
[ CA_default ]

dir                = /var/ca                # Where everything is kept
certs              = $dir/certs             # Where the issued certs are kept
crl_dir            = $dir/crl               # Where the issued crl are kept
database           = $dir/index.txt        # database index file.
#unique_subject    = no                    # Set to 'no' to allow creation
#                  # several certs with same subject
new_certs_dir      = $dir/newcerts         # default place for new certs.

certificate        = $dir/cacert.pem       # The CA certificate
serial             = $dir/serial           # The current serial number
crlnumber          = $dir/crlnumber        # the current crl number
# must be commented out to leave
crl                = $dir/crl.pem          # The current CRL
private_key        = $dir/private/cakey.pem # The private key

x509_extensions    = usr_cert              # The extensions to add to the c

# Comment out the following two lines for the "traditional"
# (and highly broken) format.
name_opt           = ca_default            # Subject Name options
cert_opt           = ca_default            # Certificate field options

# Extension copying option: use with caution.
# copy_extensions  = copy

# Extensions to add to a CRL. Note: Netscape communicator chokes on V2 C
# so this is commented out by default to leave a V1 CRL.
# crlnumber must also be commented out to leave a V1 CRL.
# crl_extensions   = crl_ext

default_days       = 365                   # how long to certify for
default_crl_days   = 30                    # how long before next CRL
default_md         = sha256                # which md to use.
preserve           = no                    # keep passed DN ordering

# A few difference way of specifying how similar the request should look
# For type CA, the listed attributes must be the same, and the optional
# and supplied fields are just that :-))
policy             = policy_anything
```

- Перейдём в директорию `/var/ca` и создадим необходимую структуру:

```
cd /var/ca
```

```
mkdir certs
```

```
mkdir crl
```

```
mkdir newcerts
```

```
mkdir private
```

```
touch index.txt
```

```
touch serial
```

```
touch crlnumber
```

```
echo 01 > serial
```

```
echo 01 > crlnumber
```

- Должна получиться следующая структура:

```
[root@srv1-cod cal# pwd
/var/ca
[root@srv1-cod cal# ls
certs  crl  crlnumber  index.txt  newcerts  private  serial
[root@srv1-cod cal# _
```

- Создадим ключ:

```
openssl genrsa -out private/cakey.pem
```

- Создадим запрос:

```
openssl req -new -key private/cakey.pem -out cacert.csr
```

- Указав имя организации и центра сертификации:

```
[root@srv1-cod cal# openssl req -new -key private/cakey.pem -out cacert.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [RU]:
State or Province Name (full name) []:
Locality Name (eg, city) []:
Organization Name (eg, company) []: IRPO
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []: ssa2026
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
[root@srv1-cod cal#
```

- Подписываем запрос и выпускаем сертификат сроком на 5 лет:

```
[root@srv1-cod cal# openssl x509 -req -days 1825 -in cacert.csr -signkey pr
Certificate request self-signature ok
subject=C=RU, O=IRPO, CN=ssa2026
[root@srv1-cod cal# _
```

- Разрешим доступ до **srv1-cod** по **SSH** для пользователя **root**

cli-cod, admin-cod, cli1-a и cli2-a (alt-workstation):

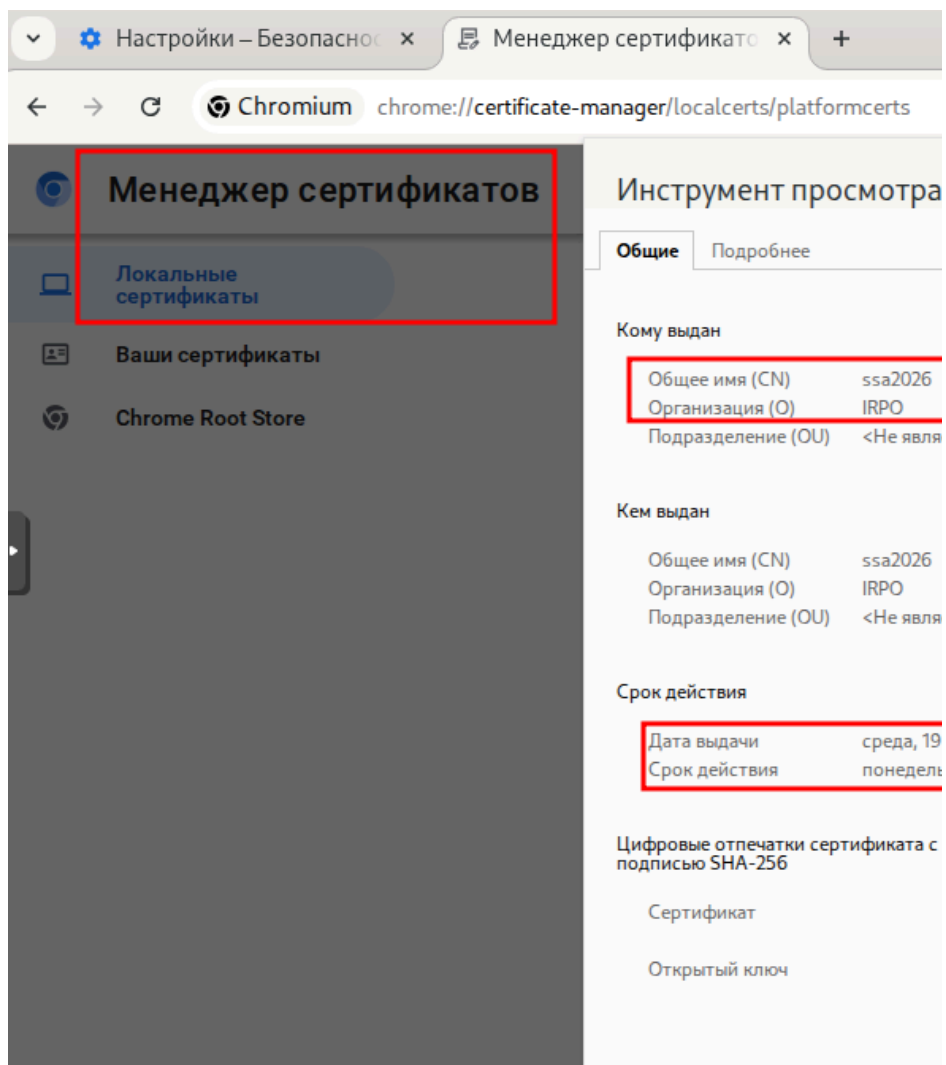
- Забираем через **scp** корневой сертификат и помещаем его в локальное хранилище сертификатов:

```
scp root@srv1-cod.cod.ssa2026.region:/var/ca/cacert.pem  
/etc/pki/ca-trust/source/anchors/ca.crt
```

- Обновляем:

```
update-ca-trust
```

- Проверить в браузере наличие корневого УЦ:



srv1-cod (alt-server):

- Перейдём в директорию **/var/ca**:

```
cd /var/ca
```

- Создаём ключ для сертификата (для системы мониторинга):

```
openssl genrsa -out necerts/monitoring.key
```

- Создаём запрос на подпись (для системы мониторинга):


```
openssl req -new -key newcerts/monitoring.key -out  
newcerts/monitoring.csr
```

◦ Результат:

```
[root@srv1-cod cal# openssl req -new -key newcerts/monitoring.key -out ne  
You are about to be asked to enter information that will be incorporated  
into your certificate request.  
What you are about to enter is what is called a Distinguished Name or a DN.  
There are quite a few fields but you can leave some blank  
For some fields there will be a default value,  
If you enter '.', the field will be left blank.  
-----  
Country Name (2 letter code) [RU]:  
State or Province Name (full name) []:.  
Locality Name (eg, city) []:.  
Organization Name (eg, company) [I:IRPO]  
Organizational Unit Name (eg, section) []:.  
Common Name (e.g. server FQDN or YOUR name) [I:srv1-cod.cod.ssa2026.regio  
Email Address []:  
  
Please enter the following 'extra' attributes  
to be sent with your certificate request  
A challenge password []:  
  optional company name []:  
▶ root@srv1-cod cal# _
```

- Создадим файл с расширениями:

```
cat <<EOF > monitoring.ext  
  authorityKeyIdentifier=keyid,issuer  
  basicConstraints=CA:FALSE  
  subjectAltName=@alt_names  
  [alt_names]  
  DNS.1=srv1-cod.cod.ssa2026.region  
  DNS.2=monitoring.cod.ssa2026.region  
  IP.1=192.168.10.1  
EOF
```

- Подписываем запрос выпуская сертификат (для системы мониторинга):

```
openssl x509 -req -CA cacert.pem -CAkey private/cakey.pem -in  
newcerts/monitoring.csr -out newcerts/monitoring.crt -days 365 -  
CAcreateserial -extfile monitoring.ext
```

◦ Результат:

```
[root@srv1-cod cal# openssl x509 -req -CA cacert.pem -CAkey private/cakey.pem -in newcerts/monitoring.cs  
-extfile monitoring.ext  
Certificate request self-signature ok  
subject=C=RU, O=IRPO, CN=srv1-cod.cod.ssa2026.region  
[root@srv1-cod cal#
```

- Проверить содержимое сертификата:

```
[root@srv1-cod ca]# openssl x509 -noout -text -in newcerts/monitoring.crt
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            4d:ae:0b:45:09:a5:fa:ab:93:47:d4:9b:de:1b:c8:80:54:09:44:f6
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=RU, O=IRPO, CN=ssa2026
        Validity
            Not Before: Nov 19 14:08:01 2025 GMT
            Not After : Nov 19 14:08:01 2026 GMT
        Subject: C=RU, O=IRPO, CN=srv1-cod.cod.ssa2026.region
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            Public-Key: (2048 bit)
            Modulus:
                00:cd:42:d7:f9:cd:0c:7f:51:99:38:08:1a:cd:61:
                f0:4c:b6:cd:c5:fc:a1:42:2c:0f:af:f8:82:24:e7:
                2c:53:78:38:9a:ba:e3:83:70:4f:d6:e8:6a:b7:fc:
                e4:91:ff:f6:e6:ef:1f:da:ca:d4:ef:0a:5e:e5:a6:
                7b:9d:3c:36:bc:0c:ef:27:ff:b3:14:84:9d:ef:04:
                b1:db:13:15:bf:2e:2b:5f:f5:a1:47:24:de:ff:2d:
                0d:53:9b:fb:a1:63:b5:a1:db:1e:f8:27:fb:d8:73:
                8e:17:32:8a:87:26:79:91:96:2f:18:b9:6d:89:35:
                db:37:0a:8c:43:96:f2:f4:67:0b:39:da:33:c1:fc:
                03:de:92:44:39:4b:f6:ae:6a:d2:f6:b9:87:d9:9d:
                88:8d:2e:e5:b9:76:f2:d3:9d:47:24:e7:c5:c4:aa:
                a0:b7:2f:2d:ff:28:c9:1a:66:a3:b1:c9:04:5a:df:
                59:8c:b3:d5:aa:bf:b4:7a:77:fa:61:b5:f8:14:e7:
                80:0e:8a:ac:fe:7e:e4:81:20:ec:91:09:01:df:84:
                cd:ac:ba:5c:af:7a:e4:37:97:bb:1f:30:46:73:03:
                1a:c9:55:4c:c8:d8:0f:ff:c3:4f:75:6a:35:f8:39:
                d3:84:b0:1d:0b:b1:88:19:9c:70:46:f3:d1:ad:3a:
                f8:95
            Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Authority Key Identifier:
                5D:A4:C2:BF:2F:16:57:62:EB:EB:C7:C2:F0:E2:5C:B4:86:E3:46:
            X509v3 Basic Constraints:
                CA:FALSE
            X509v3 Subject Alternative Name:
                DNS:srv1-cod.cod.ssa2026.region, DNS:monitoring.cod.ssa2026
            X509v3 Subject Key Identifier:
                D7:64:25:07:7E:41:6C:E5:4D:4B:62:C3:57:63:4D:15:EB:C7:0B:
        Signature Algorithm: sha256WithRSAEncryption
        Signature Value:
[root@srv1-cod ca]#
```

Последнее изменение: среда, 19 ноября 2025, 17:06

Предыдущий
элемент курса

[16. Настройка службы
доменных имен в OFFICE](#)

Следующий элемент
курса

[18. Настройка сервера
баз данных](#)

Перейти на...

Обратная связь

Вы используете гостевой
доступ (Вход)

Подпишитесь

Сводка хранения данных

Тема оформления сайта разработана

connecti.me

18. Настройка сервера баз данных

[REG26](#)

> 18. Настройка сервера баз данных

Вариант реализации:

srv2-cod (alt-server):

- Устанавливаем PostgreSQL сервер:

```
apt-get update && apt-get install -y postgresql17-server
```

- Создаём системные базы данных:

```
/etc/init.d/postgresql initdb
```

- Результат:

```
[root@srv2-cod ~]# /etc/init.d/postgresql initdb
Creating default database:
Файлы, относящиеся к этой СУБД, будут принадлежать пользователю "postgres".
От его имени также будет запускаться процесс сервера.

Кластер баз данных будет инициализирован с локалью "ru_RU.UTF-8".
Кодировка БД по умолчанию, выбранная в соответствии с настройками: "UTF8".
Выбрана конфигурация текстового поиска по умолчанию "russian".

Контроль целостности страниц данных включён.

Введите новый пароль суперпользователя:
Повторите его:

исправление прав для существующего каталога /var/lib/pgsql/data... ок
создание подкаталогов... ок
выбирается реализация динамической разделяемой памяти... posix
выбирается значение "max_connections" по умолчанию... 100
выбирается значение "shared_buffers" по умолчанию... 128MB
выбирается часовой пояс по умолчанию... Europe/Moscow
создание конфигурационных файлов... ок
▶ выполняется подготовительный скрипт... ок
полняется заключительная инициализация... ок
сохранение данных на диске... ок

Success. You can now start the database server using:

    service postgresql start
or
    pg_ctl -D /var/lib/pgsql/data -l файл_журнала start

[root@srv2-cod ~]# _
```

- Включаем и добавляем в автозагрузку PostgreSQL:

```
systemctl enable --now postgresql
```

- По умолчанию доступ из сети к PostgreSQL отключён:

```
[root@srv2-cod ~]# ss -tlnp | grep "5432"
LISTEN 0      200      127.0.0.1:5432      0.0.0.0:*    users:((("postgr
[root@srv2-cod ~]#
```

- Разрешаем доступ к PostgreSQL из сети отредактировав конфигурационный файл `/var/lib/pgsql/data/postgresql.conf`:

- в конфигурационном файле находим строку "listen_addresses = 'localhost'" и приводим её к следующему виду:

```
#-----
# CONNECTIONS AND AUTHENTICATION
#-----

# - Connection Settings -

listen_addresses = '*'

#port = 5432
#max_connections = 100
```

- Настраиваем парольную аутентификацию для удалённого доступа отредактировав конфигурационный файл `/var/lib/postgresql/data/pg_hba.conf`:
 - добавляем следующую запись:

```
# TYPE      DATABASE    USER        ADDRESS            METHOD
# "local" is for Unix domain socket connections only
local      all             all                                     scram-sha256
# IPv4 local connections:
host       all             all          0.0.0.0/0           md5
host       all             all          127.0.0.1/32        scram-sha256
# IPv6 local connections:
host       all             all          ::1/128             scram-sha256
# Allow replication connections from localhost, by a user with the
# replication privilege.
local      replication  all                                     scram-sha256
host       replication  all          127.0.0.1/32        scram-sha256
```

- Перезапускаем PostgreSQL:

```
systemctl restart postgresql
```

- Проверяем доступ по сети:

```
[root@srv2-cod ~]# ss -tlnp | grep "5432"
LISTEN 0      200      0.0.0.0:5432      0.0.0.0:*      users:((("postgres"))
LISTEN 0      200      [::]:5432       [::]:*         users:((("postgres"))
[root@srv2-cod ~]# _
```

- Создадим суперпользователя **superadmin**:
 - задав пароль **P@ssw0rdSQL**

```
createuser -U postgres --superuser --encrypted --pwprompt
superadmin
```

- Создадим базу данных **superadmin**:

```
su - postgres -s /bin/sh -c 'createdb -O superadmin superadmin'
```

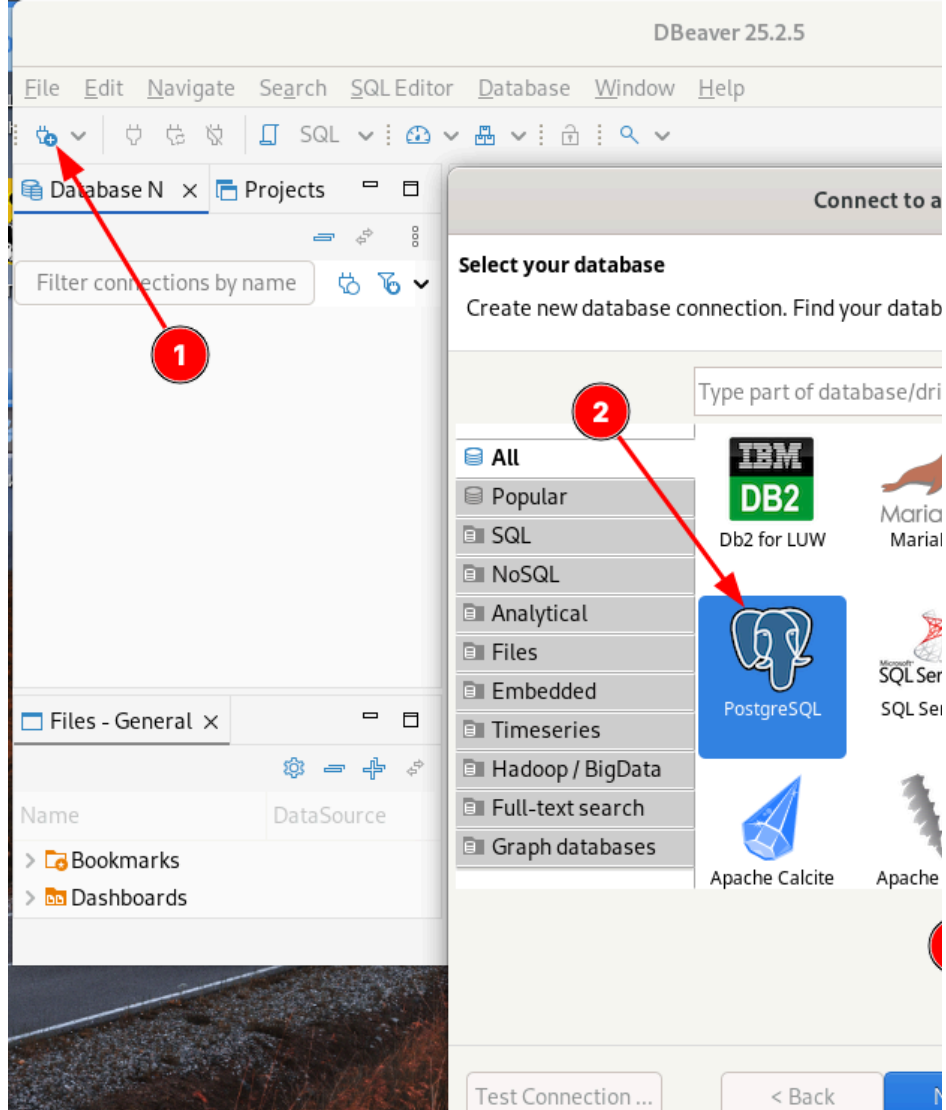
admin-cod (alt-server):

- Устанавливаем **DBeaver**:

```
apt-get update && apt-get install -y eepm
```

```
eepm play -y dbeaver
```

- Открываем **DBeaver** и создаём подключение к БД:



Connect to a database

Connection Settings

PostgreSQL connection settings



Main Advanced Driver properties + SSH, SSL, ... No pro

Server

Connect by: ☒ Host ☐ URL

URL: jdbc:postgresql://srv2-cod.cod.ssa2026.region:5432/superadmin

Host: srv2-cod.cod.ssa2026.region Port: 5432

Database: superadmin ☐ Show all dat

Authentication

Authentication: Database Native

Username: superadmin

Password: ☒ Save password

[Connection variables information](#) [PostgreSQL](#) Connection details (name, ty

Driver name: PostgreSQL Driver Settings Driver

Test Connection ...

< Back

Next >

Cancel

Fin

Driver settings



Download driver files

Download PostgreSQL driver files

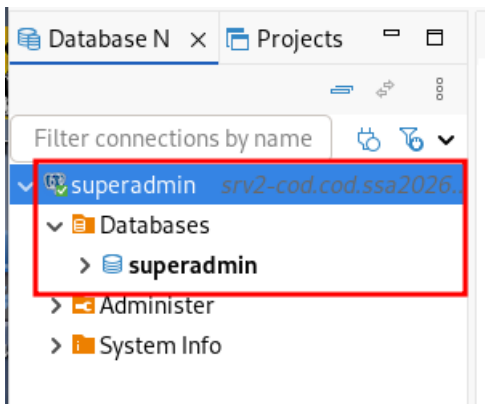
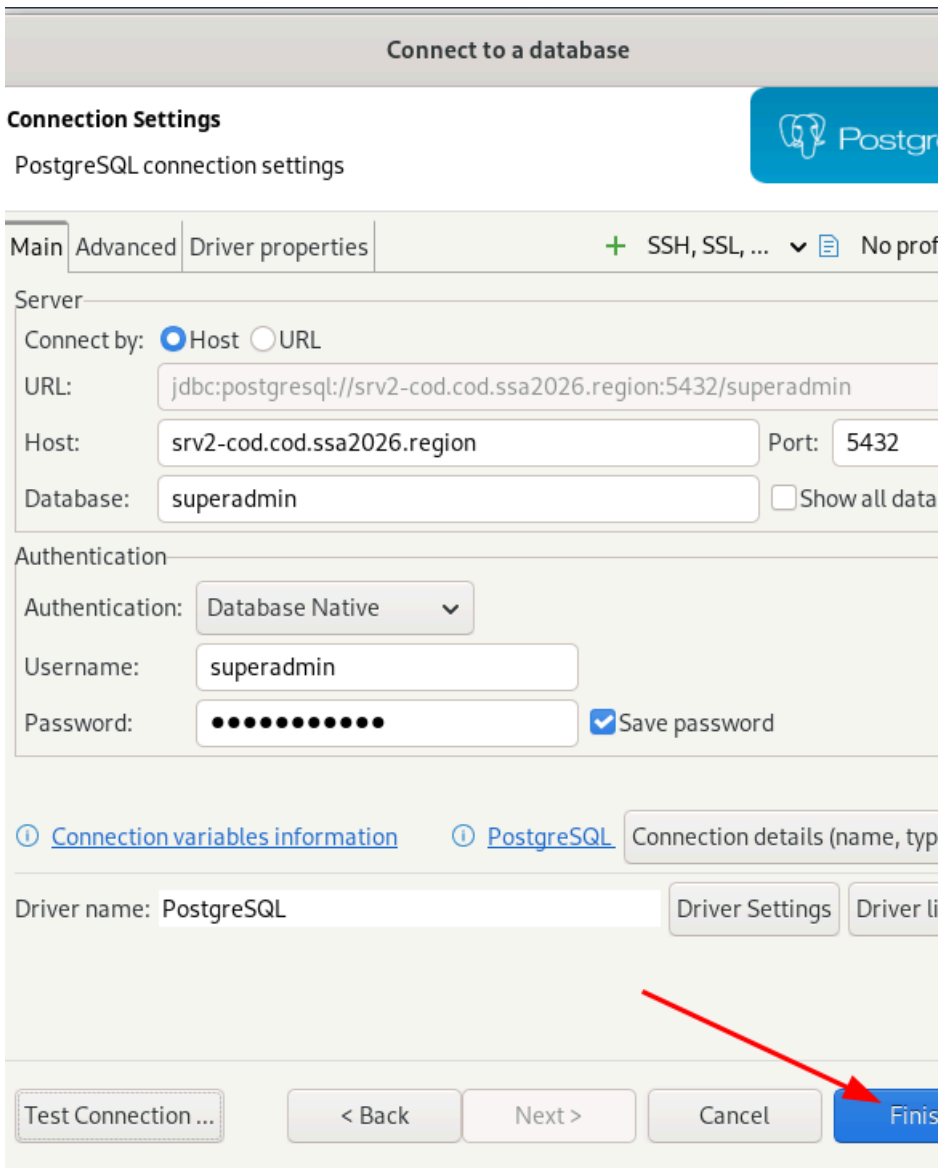
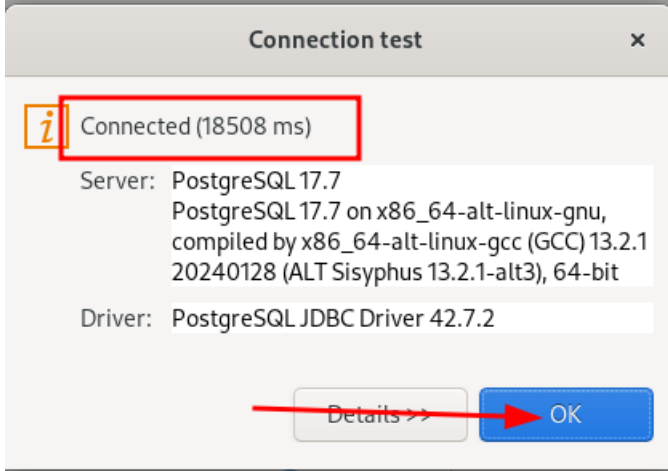
[You are about to download files for PostgreSQL standard driver.](#)

► Show details

Edit Driver

Cancel

Download



Предыдущий
элемент курса

[17. Настройка центра
сертификации](#)

Перейти на...

Следующий элемент
курса

[19. Настройка
устройства хранения
данных \(iSCSI\)](#)

Обратная связь

Подпишитесь

Вы используете гостевой
доступ (Вход)

Сводка хранения данных

Тема оформления сайта разработана

conecti.me

19. Настройка устройства хранения данных (iSCSI)

[REG26](#)

> 19. Настройка устройства хранения данных (iSCSI)

Вариант реализации:

srv2-cod (alt-server):

- Установить пакет **scsitarget-utils**:

```
apt-get update && apt-get install -y scsitarget-utils
```

- Включить и добавить в автозагрузку службу **tgt**:

```
systemctl enable --now tgt
```

- С помощью утилиты **lsblk** посмотреть список блочных устройств и определиться с диском, который будет использоваться:
 - в данном примере это **sda**

```
[root@srv2-cod ~]# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda         8:0    0    5G  0 disk 
sdb         8:16    0    20G  0 disk 
└─sdb1       8:17    0     2G  0 part [SWAP]
└─sdb2       8:18    0    18G  0 part /
sr0        11:0    1 1024M  0 rom
```

- Настроить отдачу нашего диска по iSCSI отредактировав конфигурационный файл **/etc/tgt/targets.conf**:
 - добавив в конец файла следующее содержимое

```
# This is a sample config file for tgt-admin.
#
# The "#" symbol disables the processing of a line.
# Set the driver. If not specified, defaults to "iscsi".
default-driver iscsi

# Set iSNS parameters, if needed
#iSNSServerIP 192.168.111.222
#iSNSServerPort 3205
#iSNSAccessControl On
#iSNS On

# Continue if tgtadm exits with non-zero code (equivalent of
# --ignore-errors command line option)
#ignore-errors yes

<target iqn.2026-11.region.ssa2026.cod:data.target>
    direct-store /dev/sda
</target>
```

- Перезапустить службу **tgt**:

```
systemctl restart tgt
```

- Проверить можно с помощью команды **tgtadm --lld iscsi --op show --mode target**:

```

[root@srv2-cod ~]# tgtadm --lld iscsi --op show --mode target
Target 1: iqn.2026-11.region.ssa2026.cod:data.target
System information:
  Driver: iscsi
  State: ready
I_T nexus information:
LUN information:
  LUN: 0
    Type: controller
    SCSI ID: IET      00010000
    SCSI SN: beaf10
    Size: 0 MB, Block size: 1
    Online: Yes
    Removable media: No
    Prevent removal: No
    Readonly: No
    SWP: No
    Thin-provisioning: No
    Backing store type: null
    Backing store path: None
    Backing store flags:
  LUN: 1
    Type: disk
    SCSI ID: IET      00010001
    SCSI SN: beaf11
    Size: 5369 MB, Block size: 512
    Online: Yes
    Removable media: No
    Prevent removal: No
    Readonly: No
    SWP: No
    Thin-provisioning: No
    Backing store type: rdwr
    Backing store path: /dev/sda
    Backing store flags:
Account information:
ACL information:
  ALL
[root@srv2-cod ~]#

```

- Для корректной работы необходимо указать, чтобы LVM не сканировал наши iSCSI-диски (sd*) в конфигурационном файле `/etc/lvm/lvm.conf`:
 - в блоке **devices**

```

#
# This configuration option has an automatic default value.
# filter = [ "a!*" ]
filter = [ "r!/?dev/sd*" ]
# Configuration option devices/global/filter

```

srv1-cod (alt-server):

- Установим пакет **open-iscsi**:

```
apt-get update && apt-get install -y open-iscsi
```

- Включаем и добавляем в автозагрузку службу **iscsid**:

```
systemctl enable --now iscsid
```

- Посмотреть доступные для подключения target-ы можно с помощью команды:

```
iscsiadm -m discovery -t sendtargets -p 192.168.20.2
```

- Подключить target-ы:

```
iscsiadm -m node --login
```

- В файле `/etc/iscsi/iscsid.conf` внести изменения:
 - закомментировать **node.startup = manual**

- раскомментировать `node.startup = automatic`

```
#####
# Startup settings
#####

# To request that the iscsi service scripts startup a session, use "auto
node.startup = automatic
#
# To manually startup the session, use "manual". The default is manual.
#node.startup = manual

# For "automatic" startup nodes, setting this to "Yes" will try logins
# available iface until one succeeds, and then stop. The default "No"
# logins on all available ifaces simultaneously.
```

- В файле `/var/lib/iscsi/send_targets/<TargetServer>,<Port>/st_config` внести изменения:
 - параметр `discovery.sendtargets.use_discoveryd = No` поменять на `discovery.sendtargets.use_discoveryd = Yes`

```
# BEGIN RECORD 2.1.10
discovery.startup = manual
discovery.type = sendtargets
discovery.sendtargets.address = 192.168.10.2
discovery.sendtargets.port = 3260
discovery.sendtargets.auth.authmethod = None
node.session.auth.chap_algs = MD5
discovery.sendtargets.timeo.login_timeout = 15
discovery.sendtargets.use_discoveryd = Yes
discovery.sendtargets.discoveryd_poll_inval = 30
discovery.sendtargets.reopen_max = 5
discovery.sendtargets.timeo.auth_timeout = 45
discovery.sendtargets.timeo.active_timeout = 30
discovery.sendtargets.iscsi.MaxRecvDataSegmentLength = 32768
# END RECORD
```

- Выполнить перезагрузку устройства:

reboot

- Проверить командой `lsblk` наличие блочного устройства подключённого по сети с `srv2-cod`:
 - у `srv1-cod` всего **1 диск** на **20 ГБ**

Virtual Machine 106 (srv1-cod) on node 'stand0'		alt-server-11
Summary	Add Remove Edit DiskAction Revert	
Console	Memory	2.00 GiB [balloon=0]
Hardware	Processors	2 (1 sockets, 2 cores) [x86-64-v2-AES]
Cloud-Init	BIOS	Default (SeaBIOS)
Options	Display	Default
Task History	Machine	Default (i440fx)
Monitor	SCSI Controller	VirtIO SCSI single
Backup	CD/DVD Drive (sata2)	none,media=cdrom
Replication	Hard Disk (scsi0)	local-lvm:vm-106-disk-0,iosthread=1,size=
Snapshots	Network Device (net0)	virtio=BC:24:11:7A:11:8B,bridge=vmbri1
Firewall	Network Device (net1)	virtio=BC:24:11:9D:FC:4E,bridge=vmbri1
Permissions		

- значит диск в **5 гб** тот самый по iSCSI:

```
Welcome to ALT Server 11.0 (Mendeleevium)!

Hostname: srv1-cod.cod.ssa2026.region
IP: 192.168.10.1
srv1-cod login: root
Password:
Last login: Thu Nov 20 16:12:22 MSK 2025 on tty1
[root@srv1-cod ~]# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda          8:0    0   20G  0 disk
└─sda1       8:1    0    2G  0 part [SWAP]
└─sda2       8:2    0   18G  0 part /
sdb          8:16   0    5G  0 disk
sr0         11:0    1 1024n  0 rom
[root@srv1-cod ~]#
```

Последнее изменение: пятница, 21 ноября 2025, 08:19

Предыдущий
элемент курса

[18. Настройка сервера
баз данных](#)

Перейти на...

Следующий элемент
курса

[20. Настройка
устройства хранения
данных \(LVM\)](#)

Обратная связь

Подпишитесь

Вы используете гостевой
доступ (Вход)

Сводка хранения данных

Тема оформления сайта разработана

conecti.me

20. Настройка устройства хранения данных (LVM)

[REG26](#)

> 20. Настройка устройства хранения данных (LVM)

Вариант реализации:

srv1-cod (alt-server):

- Реализуем LVM-том:
 - при помощи утилиты **lsblk** необходимо определить диск который будет использоваться:
 - в данном случае будет использоваться диск **sdb** (ранее подключённый по iSCSI)

```
[root@srv1-cod ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
sda   8:0    0  20G  0 disk
├─sda1 8:1    0   2G  0 part [SWAP]
├─sda2 8:2    0  18G  0 part /
└─sdb   8:16   0   5G  0 disk
sr0   11:0    1 1024M  0 rom
```

- Помечаем диски, что данный диск будет использоваться для LVM:

```
pvccreate /dev/sdb
```

- результат:

```
[root@srv1-cod ~]# pvccreate /dev/sdb
Physical volume "/dev/sdb" successfully created.
[root@srv1-cod ~]# _
```

- Создаём группу логических томов с именем **VG**:

```
vgcreate VG /dev/sdb
```

- результат:

```
[root@srv1-cod ~]# vgcreate VG /dev/sdb
Volume group "VG" successfully created
[root@srv1-cod ~]#
```

- Создадим логический том с именем **DATA**, используя **100%** дискового пространства:

```
lvcreate -l 100%FREE -n DATA VG
```

- результат:

```
[root@srv1-cod ~]# lvcreate -l 100%FREE -n DATA VG
Logical volume "DATA" created.
[root@srv1-cod ~]# _
```

- Проверить можно с помощью команды **lvdisplay**:

```
[root@srv1-cod ~]# lvsdisplay
--- Logical volume ---
LV Path                /dev/VG/DATA
LV Name                 DATA
VG Name                 VG
LV UUID                 FzuK7j-asKb-hIhP-gNif-AYNw-bzFB-Ab9DQ5
LV Write Access         read/write
LV Creation host, time  srv1-cod.cod.ssa2026.region, 2025-11-21 08:30:1
LV Status               available
# open                  0
LV Size                 <5.00 GiB
Current LE              1279
Segments                1
Allocation              inherit
Read ahead sectors      auto
 - currently set to     256
Block device            253:0

[root@srv1-cod ~]# _
```

- в качестве файловой системы используем **xfs**:

```
mkfs.xfs /dev/VG/DATA
```

- результат:

```
[root@srv1-cod ~]# mkfs.xfs /dev/VG/DATA
meta-data=/dev/VG/DATA             isize=512    agcount=4, agsize=327424 b
        =                           sectsz=512   attr=2, projid32bit=1
        =                           crc=1        finobt=1, sparse=1, rmapbt
        =                           reflink=1     bigtime=1 inobtcount=1 nre
        =                           exchange=0
data      =                           bsize=4096   blocks=1309696, imaxpct=25
        =                           sunit=0       swidth=0 blks
naming    =version 2                bsize=4096   ascii-ci=0, ftype=1, paren
log       =internal log             bsize=4096   blocks=16384, version=2
        =                           sectsz=512    sunit=0 blks, lazy-count=1
realtime  =none                     extsz=4096   blocks=0, rtextents=0

[root@srv1-cod ~]# _
```

- Настраиваем **автоматическое монтирование** логического тома в **/opt/data**:
 - создадим директорию для монтирования:

```
mkdir /opt/data
```

- Добавляем в файл **/etc/fstab** следующую строку для автоматического монтирования логического тома **DATA**:
 - *P.S. или на основе UUID*

```
proc          /proc                      proc      nosuid,nosexec,gid=proc
devpts        /dev/pts                  devpts    nosuid,nosexec,gid=tty,mo
tmpfs         /tmp                      tmpfs     nosuid
UUID=5f52230a-0885-4f1d-ad03-a424e1077d2c /      ext4      relatime
UUID=6bbd2088-67a6-4df6-8814-e898623224d3 swap    swap      defaults
/dev/VG/DATA  /opt/data      xfs       defaults,_netdev 0
```

- Выполняем монтирование:

```
mount -av
```

- результат:

```
[root@srv1-cod ~]# mount -av
/proc          : already mounted
/dev/pts       : already mounted
/tmp           : already mounted
/              : ignored
swap           : ignored
/opt/data      : successfully mounted
[root@srv1-cod ~]#
```

- Проверить можно с помощью команд **lsblk** или **df**:

```
[root@srv1-cod ~]# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda          8:0    0   20G  0 disk
├─sda1       8:1    0    2G  0 part [SWAP]
└─sda2       8:2    0   18G  0 part /
sdb          8:16   0    5G  0 disk
└─VG-DATA 253:0  0    5G  0 lvm  /opt/data
sr0         11:0    1 1024M  0 rom

[root@srv1-cod ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
udevfs          5.0M   4.0K  5.0M   1% /dev
runfs           986M  600K  986M   1% /run
/dev/sda2       18G   2.1G   15G  13% /
tmpfs           986M    0   986M   0% /dev/shm
tmpfs           986M    0   986M   0% /tmp
tmpfs           198M   4.0K  198M   1% /run/user/0
/dev/mapper/VG-DATA 5.0G  130M  4.9G   3% /opt/data
[root@srv1-cod ~]#
```

Последнее изменение: пятница, 21 ноября 2025, 09:31

Предыдущий
элемент курса

[19. Настройка
устройства хранения
данных \(iSCSI\)](#)

Перейти на...

Следующий элемент
курса

[21. Настройка
устройства хранения
данных \(NFS\)](#)

Обратная связь

Подпишитесь

Вы используете гостевой
доступ (Вход)

Сводка хранения данных

Тема оформления сайта разработана

connecti.me

21. Настройка устройства хранения данных (NFS)

[REG26](#)

> 21. Настройка устройства хранения данных (NFS)

Вариант реализации:

srv1-cod (alt-server):

- Устанавливаем пакеты для NFS сервера:

```
apt-get install -y nfs-server nfs-utils
```

- Назначаем права на созданную ранее директорию для общего доступа (полный доступ):

```
chmod 777 /opt/data
```

- Редактируем файл **/etc/exports**:
 - Добавляем туда следующую информацию

```
#/srv/share -rw,insecure,fsid=0,sec=krb5 *  
/opt/data 192.168.30.0/24(rw,no_root_squash)
```

- Экспортируем файловую систему, указанную выше в **/etc/exports**:

```
exportfs -arv
```

- Результат:

```
[root@srv1-cod ~]# exportfs -arv  
exporting 192.168.30.0/24:/opt/data  
[root@srv1-cod ~]# _
```

- Запускаем и добавляем в автозагрузку NFS - сервер:

```
systemctl enable --now nfs-server
```

admin-cod (alt-workstation):

- Выполняем установку пакетов для NFS - клиента:

```
apt-get update && apt-get install -y nfs-utils nfs-clients
```

- Создадим директорию для монтирования общего ресурса:

```
mkdir /mnt/nfs
```

- Задаём права на созданную директорию:

```
chmod 777 /mnt/nfs
```


- Настраиваем автомонтирование общего ресурса через **fstab**:

- Добавляем следующую информацию:

```
proc          /proc          proc      nosuid,nosuid,gid=proc
devpts        /dev/pts        devpts    nosuid,nosuid,gid=tty,mo
tmpfs         /tmp          tmpfs     nosuid
UUID=eb35d215-ef34-4519-b707-8daa5f637662 /      btrfs    relatime
UUID=eb35d215-ef34-4519-b707-8daa5f637662 /home   btrfs    nosuid,t
UUID=477c4adc-ab05-4d7c-a66d-afae7c94e593 swap     swap     defaults
/dev/sr0      /media/ALTlinux udf,iso9660 ro,noauto,user,utf8,nofa
192.168.10.1:/opt/data /mnt/nfs nfs      defaults,_netdev
```

- Выполняем монтирование общего ресурса:

```
mount -av
```

- Результат:

```
[root@admin-cod ~]# mount -av
/dev/pts      : already mounted
/dev/pts      : already mounted
/dev/pts      : already mounted
/             : ignored
/home         : already mounted
swap         : ignored
/media/ALTlinux : ignored
mount.nfs: timeout set for Fri Nov 21 09:03:07 2025
mount.nfs: trying text-based options 'vers=4.2,addr=192.168.10.1,clienta
/mnt/nfs      : successfully mounted
[root@admin-cod ~]#
```

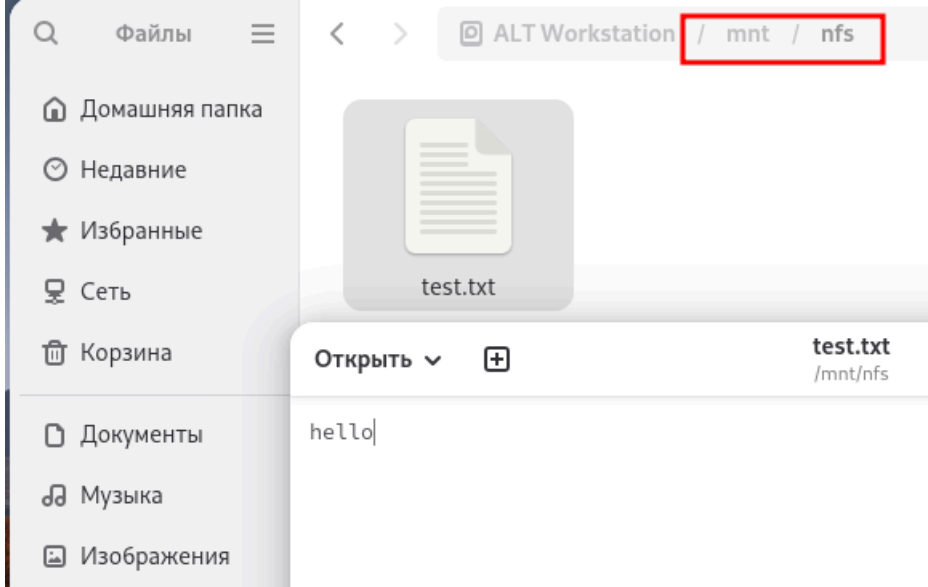
- Проверяем:

```
df -h
```

- Результат:

```
[root@admin-cod ~]# df -h
Файловая система    Размер  Использовано  Дост  Использовано%  Смонтиро
udevfs              5,0М    4,0К    5,0М           1% /dev
runfs               986М    1,3М    985М           1% /run
/dev/sda2            17G     9,0G    7,1G          56% /
tmpfs               986М    0       986М           0% /dev/shm
tmpfs               986М    20K     986М           1% /tmp
/dev/sda2            17G     9,0G    7,1G          56% /home
tmpfs               198М    92K     198М           1% /run/use
192.168.10.1:/opt/data 5,0G    129М    4,9G           3% /mnt/nfs
[root@admin-cod ~]#
```

- Перезгружаем **admin-cod** и проверяем автомонтирование с правами на запись:



- Результат на сервере **srv1-cod**:

```
[root@srv1-cod ~]# ls /opt/data/  
test.txt  
[root@srv1-cod ~]# cat /opt/data/test.txt  
hello  
[root@srv1-cod ~]#
```

Последнее изменение: пятница, 21 ноября 2025, 08:58

Предыдущий
элемент курса

[20. Настройка
устройства хранения
данных \(LVM\)](#)

Перейти на...

Следующий элемент
курса

[22. Настройка системы
мониторинга](#)

Обратная связь

Подпишитесь

Вы используете гостевой
доступ (Вход)

Сводка хранения данных

Тема оформления сайта разработана

conecti.me

22. Настройка системы мониторинга

[REG26](#)

> 22. Настройка системы мониторинга

Вариант реализации:

srv2-cod (alt-server):

- Устанавливаем пакет **zabbix-server-pgsql**:

```
apt-get update && apt-get install -y zabbix-server-pgsql
```

- Создаём пользователя **zabbix_user** в базе данных **PostgreSQL**:

```
su - postgres -s /bin/sh -c 'createuser --no-superuser --no-createdb --no-createrole --encrypted --pwprompt zabbix_user'
```

- После запуска данной команды - задаём в качестве пароля для пользователя **zabbix_user** - пароль **P@ssw0rdZabbix** и подтверждаем его:

```
root@srv2-cod ~1# su - postgres -s /bin/sh -c 'createuser --no-superuser --no-createdb --no-createrole --encrypted --pwprompt zabbix_user'
Enter password for new role:
Enter it again:
Password:
root@srv2-cod ~1#
```

- Создаём базу данных с именем **zabbix**:

```
su - postgres -s /bin/sh -c 'createdb -O zabbix_user zabbix'
```

- Выполняем перезагрузку службы **postgresql**:

```
systemctl restart postgresql
```

- Добавляем в базу данные для веб-интерфейса:

```
su - postgres -s /bin/sh -c 'psql -U zabbix_user -f /usr/share/doc/zabbix-common-database-pgsql-*/schema.sql zabbix'
```

```
su - postgres -s /bin/sh -c 'psql -U zabbix_user -f /usr/share/doc/zabbix-common-database-pgsql-*/images.sql zabbix'
```

```
su - postgres -s /bin/sh -c 'psql -U zabbix_user -f /usr/share/doc/zabbix-common-database-pgsql-*/data.sql zabbix'
```

srv1-cod (alt-server):

- Устанавливаем пакеты для веб-сервера **apache2**:

```
apt-get install -y apache2 apache2-mod_{php8.2,ssl}
```

- Включаем и добавляем в автозагрузку службу отвечающую за веб-сервер **apache2**:

```
systemctl enable --now httpd2
```

- Установим **PHP** и необходимые модули для корректной работы:

```
apt-get install -y php8.2 php8.2-  
{mbstring,sockets,gd,xmlreader,pgsql,ldap,openssl}
```

- Меняем некоторые опции **php** в файле **/etc/php/8.2/apache2-mod_php/php.ini**:
 - Находим следующие параметры и приводим их к следующему виду:

```
;;;;;;;;;;  
; Maximum execution time of each script, in seconds  
; http://php.net/max-execution-time  
; Note: This directive is hardcoded to 0 for the CLI SAPI  
max_execution_time = 600  
; Maximum amount of time each script may spend parsing request  
; idea to limit this time on production servers in order to  
; long running scripts.  
; Note: This directive is hardcoded to -1 for the CLI SAPI  
; Default Value: -1 (Unlimited)  
; Development Value: 60 (60 seconds)  
; Production Value: 60 (60 seconds)  
; http://php.net/max-input-time  
max_input_time = 600  
; Maximum input variable nesting level  
; http://php.net/max-input-nesting-level  
;max_input_nesting_level = 64  
; How many GET/POST/COOKIE input variables may be accepted  
;max_input_vars = 1000  
; Maximum amount of memory a script may consume (128MB)  
; http://php.net/memory-limit  
memory_limit = 256M
```

```
; Maximum size of POST data that PHP will accept.  
; Its value may be 0 to disable the limit. It is ignored if  
; is disabled through enable_post_data_reading.  
; http://php.net/post-max-size  
post_max_size = 32M
```

```
[Date]  
; Defines the default timezone used by the date functions  
; http://php.net/date.timezone  
date.timezone = Europe/Moscow
```

```
; Always populate the $HTTP_RAW_POST_DATA variable. PHP's de  
; to disable this feature and it will be removed in a future  
; If post reading is disabled through enable_post_data_readi  
; $HTTP_RAW_POST_DATA is *NOT* populated.  
; http://php.net/always-populate-raw-post-data  
always_populate_raw_post_data = -1
```

- Перезапускаем службу отвечающую за веб-сервер **apache2**:

```
systemctl restart httpd2
```

- Устанавливаем **zabbix-server-pgsql**:

```
apt-get install -y zabbix-server-pgsql
```

- Вносим изменения в конфигурационный файл **/etc/zabbix/zabbix_server.conf**:

- Добавляем следующие изменения:

```
# Mandatory: no
# Default:
DBHost=192.168.10.2

### Option: DBName
# Database name.
# If the Net Service Name connection method is used
# the tnsnames.ora file or set to empty string; also
# empty string.
#
# Mandatory: yes
# Default:
# DBName=

DBName=zabbix

# Option: DBSchema
# Schema name. Used for PostgreSQL.
#
# Mandatory: no
# Default:
# DBSchema=

### Option: DBUser
# Database user.
#
# Mandatory: no
# Default:
# DBUser=

DBUser=zabbix_user

### Option: DBPassword
# Database password.
# Comment this line if no password is used.
#
# Mandatory: no
# Default:
DBPassword=P@ssw0rdZabbix

### Option: DBSocket
# Path to MySQL socket.
```

- Добавим Zabbix-сервер в автозапуск и запустить его:

```
systemctl enable --now zabbix_pgsql
```

- Установка веб-интерфейса Zabbix:

```
apt-get install zabbix-phpfrontend-{apache2,php8.2} -y
```

- Создаём файл конфигурации Virtualhost для Zabbix /etc/httpd/conf/sites-available/zabbix.conf:
 - Содержимое:

```
<IfModule mod_ssl.c>
    <VirtualHost *:443>
        DocumentRoot /var/www/webapps/zabbix/ui
        ServerName srv1-cod.cod.ssa2026.region
        ServerAlias monitoring.cod.ssa2026.region 192.168.10.1
        <Directory /var/www/webapps/zabbix/ui/>
            Options Indexes FollowSymlinks
            AllowOverride All
            Require all granted
        </Directory>
        SSLCertificateFile /var/lib/ssl/certs/monitoring.crt
        SSLCertificateKeyFile /var/lib/ssl/private/monitoring.key
    </VirtualHost>

    <VirtualHost *:80>
        ServerName srv1-cod.cod.ssa2026.region
        ServerAlias monitoring.cod.ssa2026.region 192.168.10.1
        Redirect permanent / https://srv1-cod.cod.ssa2026.region/
    </VirtualHost>
```

- Копируем ранее созданные сертификаты:

```
cp /var/ca/newcerts/monitoring.crt /var/lib/ssl/certs/
```

```
cp /var/ca/newcerts/monitoring.key /var/lib/ssl/private/
```

- Добавляем символическую ссылку на файл **/etc/httpd/conf/sites-available/zabbix.conf**:

```
ln -s /etc/httpd/conf/sites-available/zabbix.conf  
/etc/httpd/conf/sites-enabled/
```

- Включаем модуль **ssl** и **rewrite**, а также порт **https**:

```
a2enmod ssl; a2enmod rewrite; a2enport https
```

- Проверяем синтаксис конфигурационного файла **/etc/httpd/conf/sites-available/zabbix.conf**:

```
apachectl -t
```

- Результат:

```
[root@srv1-cod ~]# apachectl -t  
Syntax OK  
[root@srv1-cod ~]#
```

- Назначаем владельца и группу владельцев для файлов проекта **zabbix**:

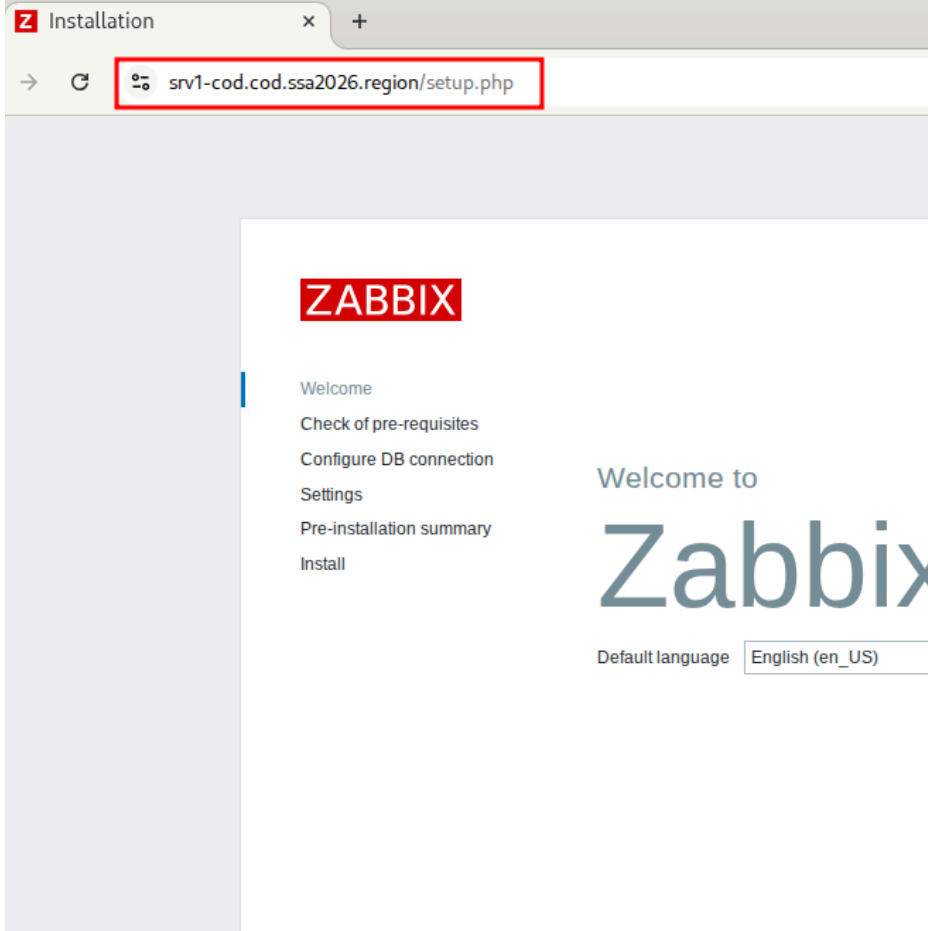
```
chown -R apache2:apache2 /var/www/webapps/zabbix
```

- Перезагружаем службу **httpd2**:

```
systemctl restart httpd2
```

admin-cod (alt-workstation):

- Продолжаем установку **zabbix** - через веб-интерфейс:
 - переходим на <https://srv1-cod.cod.ssa2026.region> и нажимаем **Next step**:



- нажимаем **Next step**:

ZABBIX

Welcome

Check of pre-requisites

Configure DB connection

Settings

Pre-installation summary

Install

Check of pre-requisites

	Current
PHP version	8.2.21
PHP option "memory_limit"	256M
PHP option "post_max_size"	32M
PHP option "upload_max_filesize"	20M
PHP option "max_execution_time"	600
PHP option "max_input_time"	600
PHP databases support	MySQL PostgreSQL
PHP bcmath	on
PHP mbstring	on
PHP option "mbstring.func_overload"	off

- В качестве базы данных выбираем - **PostgreSQL**;
- В качестве сервера базы данных указываем **IP** - адрес или имя **srv2-cod.cod.ssa2026.region**;
- Указываем имя созданной базы данных **zabbix**;
- Указываем имя созданного пользователя **zabbix_user**;
- Указываем пароль для пользователя zabbix **P@ssw0rdZabbix**;
- Нажимаем **Next step**



Configure DB connection

Please create database manually, and set the configuration p database. Press "Next step" button when done.

Welcome

Check of pre-requisites

Configure DB connection

Settings

Pre-installation summary

Install

Database type PostgreSQL

Database host srv2-cod.cod.ssa2026.re

Database port 0 0 - us

Database name zabbix

Database schema

Store credentials in Plain text HashiCorp Vault

User zabbix_user

Password

Database TLS encryption ☒

- при необходимости задаём имя серверу и нажимаем **Next step**



Settings

Zabbix server name monitoring.cod.ssa2026.r

Default time zone System: (UTC+03:00) Europe/Moscow

Default theme Blue

Welcome

Check of pre-requisites

Configure DB connection

Settings

Pre-installation summary

Install

- проверяем введённые ранее параметры и нажимаем **Next step**



Welcome
Check of pre-requisites
Configure DB connection
Settings
Pre-installation summary
Install

Pre-installation summary

Please check configuration parameters. If all is correct, press
change configuration parameters.

Database type	PostgreSQL
Database server	srv2-cod.cod.ssa2026.region
Database port	default
Database name	zabbix
Database user	zabbix_user
Database password	*****
Database schema	
Database TLS encryption	true
Zabbix server name	monitoring.cod.ssa2026.region

-
- нажимаем **Finish**



Welcome
Check of pre-requisites
Configure DB connection
Settings
Pre-installation summary
Install


Install

**Congratulations! You have success
frontend.**

Configuration file "conf/zabbix.conf.php" created.

-
- выполняем вход из под пользователя по умолчанию: **Admin** с
паролем: **zabbix**

srv1-cod.cod.ssa2026.region/index.php



Username

Password


☒ Remember me for 30 days

Sign in

- Результат:

monitoring.cod.ssa2026. x +

srv1-cod.cod.ssa2026.region/zabbix.php?action=dashboard.view&dashboardid=1



Dashboards
Monitoring
Services
Inventory
Reports
Data collection
Alerts
Users
Administration
Support
Integrations
Help
User settings
Sign out

Global view


All dashboards / Global view

Top hosts by CPU utilization

Host name	Utilization	1m avg	5m avg	15m avg	Processes
Zabbix server					

1.10↓

Zabbix server Values per sec...



System info

Parameter
Zabbix server
Zabbix server
Zabbix frontend
Number of hosts
Number of templates
Number of items supported

Host availability

0 Available	1 Not available	0 Mixed	0 Unknown	1 Tot
-------------	-----------------	---------	-----------	-------

Problems by severity

0 Disaster	0 High	1 Average	0 Warning
------------	--------	-----------	-----------

Current problems

Time	Info	Host	Problem • Severity	Duration	Update	Actions	Tag
10:37:26 AM		Zabbix server	Linux: Zabbix agent is not available (for 3m)	59m 47s	Update		cla scc

- В качестве пароля для пользователя **Admin** - необходимо установить **P@ssw0rd**:
 - переходим в настройки аутентификации и снимаем галочку, которая запрещает использование слабых паролей

monitoring.cod.ssa2026. x +

← → ↻ ⚙️ srv1-cod.cod.ssa2026.region/zabbix.php?action=authentication.edit

ZABBIX << >>
monitoring.cod.ssa2026.r...

Dashboards

Monitoring

Services

Inventory

Reports

Data collection

Alerts

Users

User groups

User roles

Users

API tokens

Authentication

Authentication

Authentication HTTP settings LDAP settings SAML settings

Default authentication Internal LDAP

Deprovisioned users group ?

Password policy

Minimum password length

Password must contain ?
☐ an uppercase and a lowercase
☐ a digit
☐ a special character

Avoid easy-to-guess passwords ? ☐

Update

Authentication settings updated

- Задаём новый пароль **P@ssw0rd** - для пользователя **Admin**

monitoring.cod.ssa2026. x +

← → ↻ 🌐 srv1-cod.cod.ssa2026.region/zabbix.php

ZABBIX << 🔍
monitoring.cod.ssa2026.r...

Dashboards
Monitoring
Services
Inventory
Reports
Data collection
Alerts
Users
User groups
User roles
Users
API tokens
Authentication
Administration

Users

User Media Permissions

* Username Admin

Name Zabbix

Last name Administrator

Groups Internal Zabbix administrators
type here to search

* Current password

* Password ?

* Password (once again)

Password is not mandatory for non int

Language System default

Time zone System default: (UTC+03:00) Europe

Theme System default

Auto-login ☒

Auto-logout ☐ 15m

* Refresh 30s

* Rows per page 50

srv1-cod, srv2-cod, dc-a, sw1-cod, sw2-cod, sw1-a, sw2-a
(alt-server):

- Устанавливаем необходимый пакет **zabbix-agent**:

```
apt-get install zabbix-agent -y
```

- Редактируем конфигурационный файл **/etc/zabbix/zabbix_agentd.conf**:
 - вносим следующие изменения:

```
# Mandatory: yes, if StartAgents is not explicitly set to 0
# Default:
# Server=
Server=192.168.10.1
### Option: ListenPort
# Agent will listen on this port for connections from the s
#
```

- в поле **Hostname** необходимо указывать полное доменное имя соответствующего хоста

```

ServerActive=192.168.10.1

# Option: Hostname
#   List of comma delimited unique, case sensitive hostnames.
#   Required for active checks and must match hostnames as configured.
#   Value is acquired from HostnameItem if undefined.
#
# Mandatory: no
# Default:
# Hostname=

Hostname=srv1-cod.cod.ssa2026.region

```

- Включаем и добавляем в автозагрузку службу **zabbix_agentd**:

```
systemctl enable --now zabbix_agentd.service
```

admin-cod (alt-workstation):

- Переходим в веб-интерфейс управления **fw-cod**:

The screenshot shows the web interface of Idec NGFW Novum. The browser address bar shows the URL `https://fw-cod.cod.ssa2026.region:8443/#/monitor/zabbix-agent`. The left sidebar has a menu with items like 'Панель мониторинга', 'Пользователи', 'Мониторинг' (selected), 'Сессии администраторов', 'Сессии пользователей', 'Сессии ЛК', 'Графики загрузки', 'Монитор трафика', 'Проверка IP/домена', 'Telegram-бот', 'SNMP', 'Zabbix-агент' (selected), 'Правила трафика', 'Профили безопасности', 'Сервисы', and 'Маршрутизация'. The main content area is titled 'Zabbix-агент' and shows the status 'Остановлен'. There are several checkboxes: 'Zabbix-агент' (checked), 'Отправка данных к Zabbix (активный)' (unchecked), and 'Приём запросов от Zabbix (пассивный)' (checked). Below these are fields for 'Порт для подключения' (10050 selected) and 'Адрес сервера' (192.168.10.1). At the bottom is a 'Сохранить' button.

- Результат:

✓ Модуль "zabbix-agent" работает

rtr-cod и rtr-a (ecorouter):

- Запуск сервиса SNMP:

```
rtr-cod(config)#snmp-server enable snmp
rtr-cod(config)#
```

- Настроить SNMP community:

```
rtr-cod(config)#snmp-server community public ro
rtr-cod(config)#write memory
Building configuration...

rtr-cod(config)#
```

admin-cod (alt-workstation):

- Переходим на <https://srv1-cod.cod.ssa2026.region>
- Каждый хост необходимо зарегистрировать на сервере Zabbix, сделать это можно, используя веб-интерфейс
 - переходим **Monitoring -> Hosts -> Create host**:

The screenshot shows the Zabbix web interface. The left sidebar has a menu with 'Monitoring' and 'Hosts' highlighted. Red circles and arrows labeled '1' and '2' point to these items. The main area shows a table of hosts. The table has columns: Interface, Availability, Tags, and Status. A host with IP 127.0.0.1:10050 is shown with status 'Enabled'.

Interface	Availability	Tags	Status
127.0.0.1:10050	ZBX	class: os class: software target: linux ***	Enabled

- Добавляем хосты в соответствии с требованиями задания:

Host

Host

IPMI

Tags

Macros

Inventory

Encryption

Value mapping

* Host name

srv1-cod.cod.ssa2026.region

Visible name

srv1-cod.cod.ssa2026.region

Templates

Linux by Zabbix agent X

type here to search

* Host groups

Linux servers X

type here to search

Interfaces

Type

IP address

DNS name

Agent

192.168.10.1

srv1-cod.cod.ssa2026.regior

Add

Description

- Аналогично все остальные хосты:

Name ▲	Interface	Availability	Tags
dc-a.office.ssa2026.region	172.20.10.10:10050	ZBX	class: os target: linux
rtr-a.office.ssa2026.region	10.10.10.2:161	SNMP	class: network target: cisco
rtr-cod.cod.ssa2026.region	172.16.1.1:161	SNMP	class: network target: cisco
srv1-cod.cod.ssa2026.region	192.168.10.1:10050	ZBX	class: os target: linux
srv2-cod.cod.ssa2026.region	192.168.10.2:10050	ZBX	class: os target: linux
sw1-a.office.ssa2026.region	172.20.30.1:10050	ZBX	class: os target: linux
sw1-cod.cod.ssa2026.region	192.168.30.1:10050	ZBX	class: os target: linux
sw2-a.office.ssa2026.region	172.20.30.2:10050	ZBX	class: os target: linux
sw2-cod.cod.ssa2026.region	192.168.30.2:10050	ZBX	class: os target: linux

Последнее изменение: пятница, 21 ноября 2025, 12:23

Предыдущий
элемент курса

21. Настройка
устройства хранения
данных (NFS)

Перейти на...

Следующий элемент
курса

23. Настройка IP
телефонии

Обратная связь

Подпишитесь

Вы используете гостевой
доступ (Вход)

Сводка хранения данных

23. Настройка IP телефонии

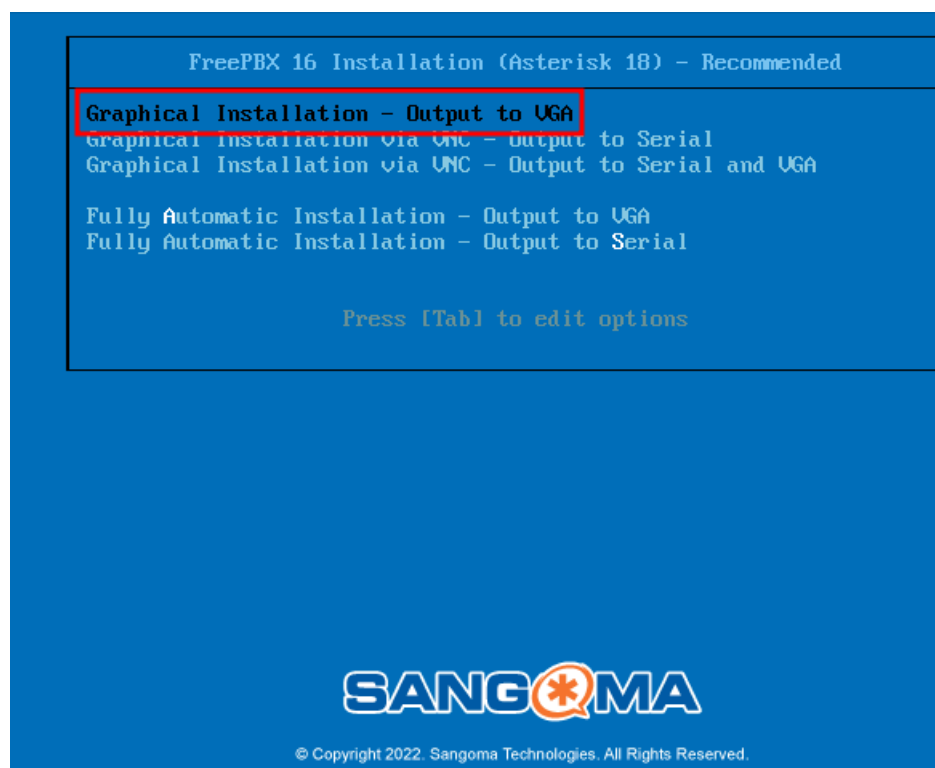
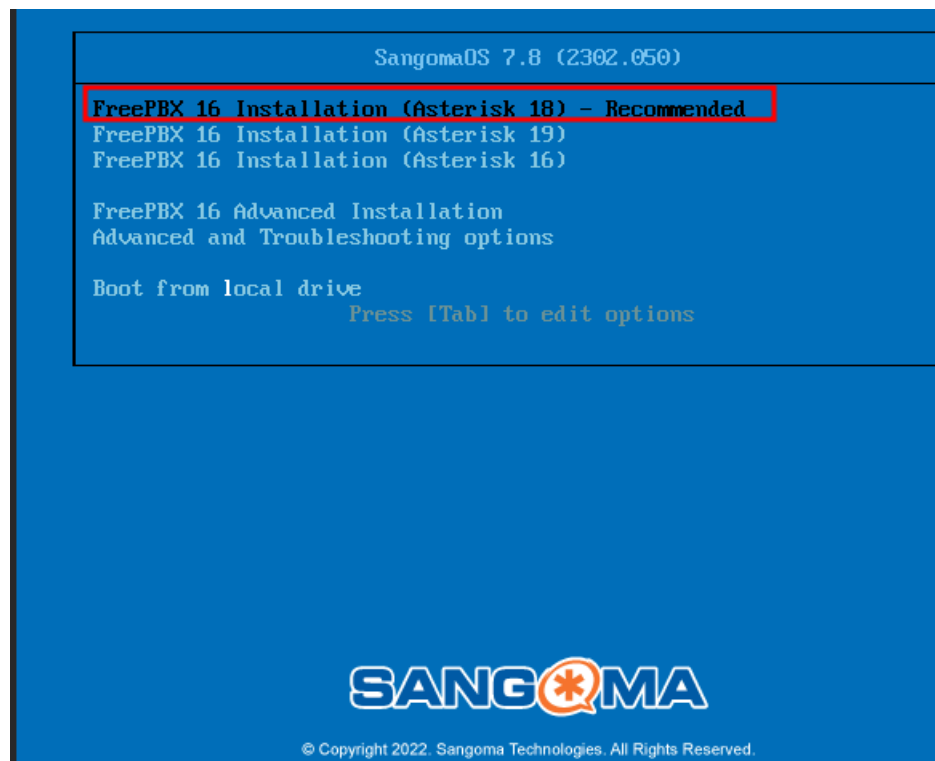
[REG26](#)

> 23. Настройка IP телефонии

Вариант реализации:

sip-cod (SNG7-PBX16):

- Устанавливаем ОС:



FreePBX Standard

Press [Tab] to edit options

Automatic install of Asterisk 18 and FreePBX 16
Note that if more than two identically sized HDDs are detected,
a RAID will be created and they will assigned to it
This install deletes all existing data on this machine



© Copyright 2022. Sangoma Technologies. All Rights Reserved.



CONFIGURATION

SANGOMAOS 7.8 2302-1 INST

us

USER SETTINGS



ROOT PASSWORD
Root password is not set



USER CREATION
User asterisk will be created

Starting package installation process



Choose the Sangoma IP Phone
That Best Fits Your Needs



ROOT PASSWORD

SANGOMAOS 7.8 2302-1 INS

Done

us

The root account is used for administering the system. Enter a password for the root user.

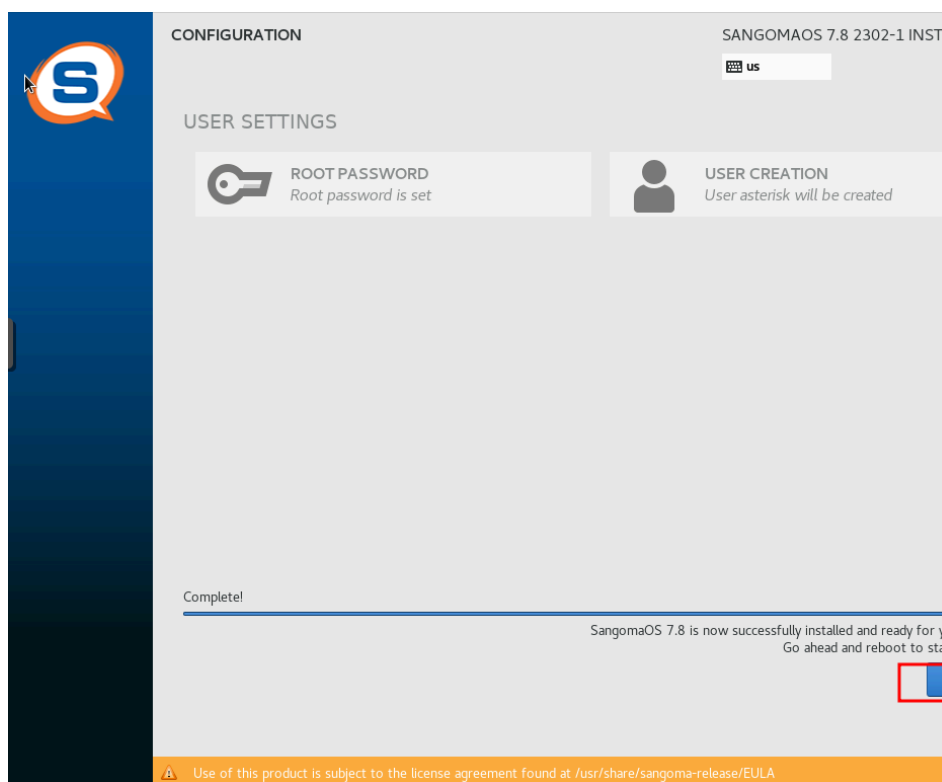
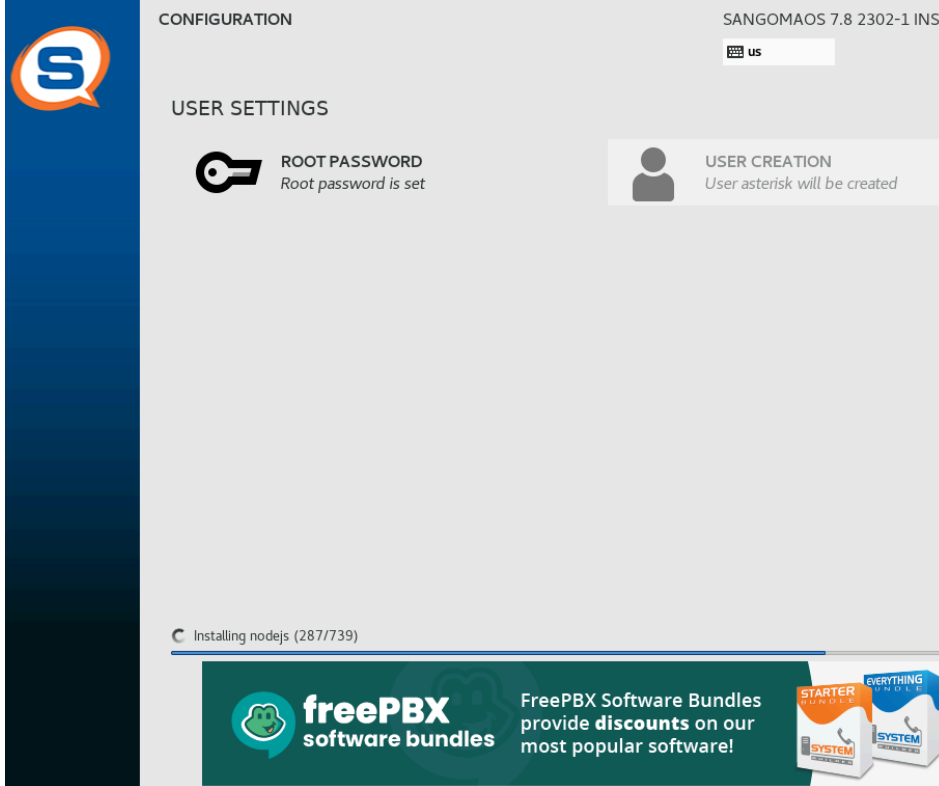
Root Password:

....

Too short

Confirm:

....



- Логинимся из-под суперпользователя **root** с паролем указанным на этапе установки ОС
- Редактируем конфигурационный файл **/etc/sysconfig/network-scripts/ifcfg-eth0**:

```

TYPE="Ethernet"
PROXY_METHOD="none"
BROWSER_ONLY="no"
BOOTPROTO="static"
IPADDR="192.168.50.50"
NETMASK="255.255.255.0"
GATEWAY="192.168.50.254"
DNS1="192.168.10.1"
DEFROUTE="yes"
IPV4_FAILURE_FATAL="no"
IPV6INIT="yes"
IPV6_AUTOCONF="yes"
IPV6_DEFROUTE="yes"
IPV6_FAILURE_FATAL="no"
IPV6_ADDR_GEN_MODE="stable-privacy"
NAME="eth0"
UUID="6080706e-7d6a-4baf-ae4a-3d6860361673"
DEVICE="eth0"
ONBOOT="yes"

```

- Перезагружаем **sip-cod**:

```
reboot
```

- Проверяем заданные сетевые параметры:

```

[root@freepbx ~]# ip -c -br -4 a
lo                UNKNOWN    127.0.0.1/8
eth0              UP          192.168.50.50/24
[root@freepbx ~]# ip -c r
default via 192.168.50.254 dev eth0
169.254.0.0/16 dev eth0 scope link metric 1002
192.168.50.0/24 dev eth0 proto kernel scope link src 192.168.50.50
[root@freepbx ~]# cat /etc/resolv.conf

nameserver 192.168.10.1

```

admin-cod (alt-workstation):

- Переходим в браузере на **192.168.50.50**:
 - Создаём учётную запись Администратора

Welcome to FreePBX Administration!

Initial Setup

Please provide the core settings that will be used to administer and update your system

Administrator User

Username: admin

Password:

Really Weak

Confirm Password:

System Notifications Email

Notifications Email address: admin@cod.ssa2026.region

System Identification

System Identifier: VoIP Server

System Updates

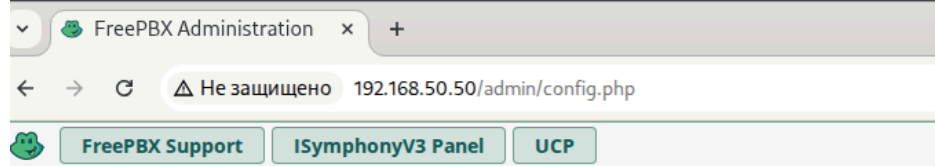
Automatic Module Updates:

Automatic Module Security Updates:

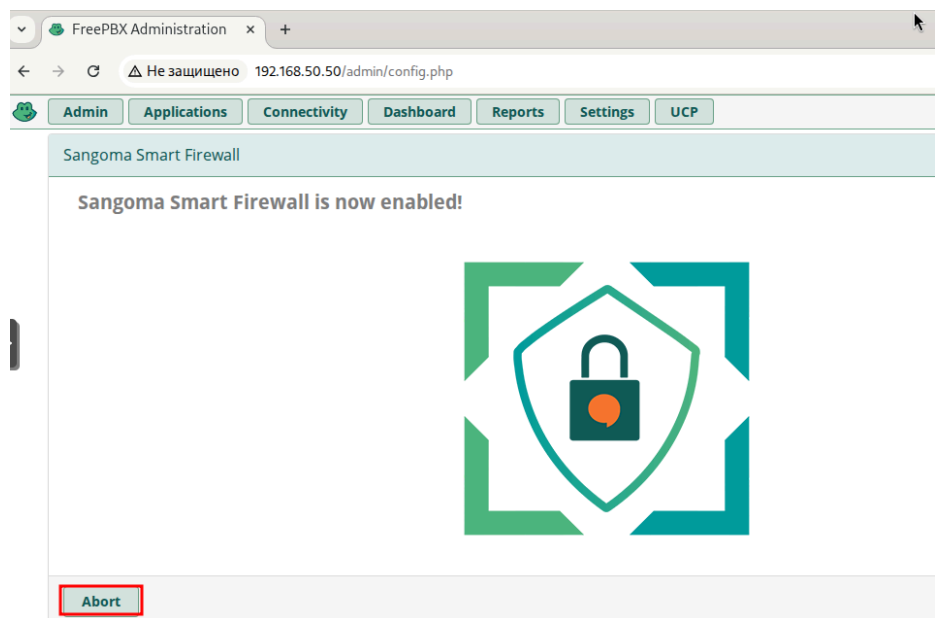
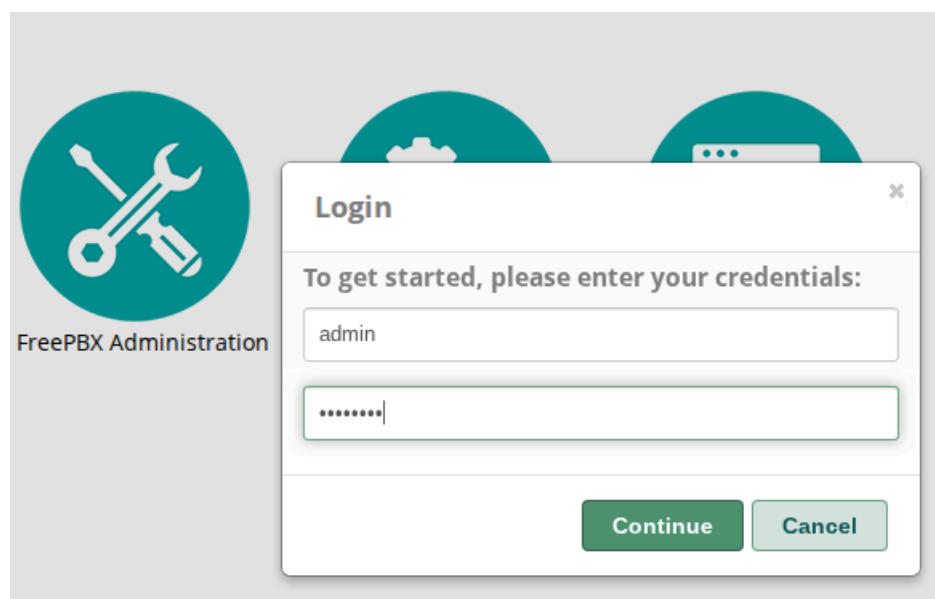
Send Security Emails For Unsigned Modules:

Check for Updates every:

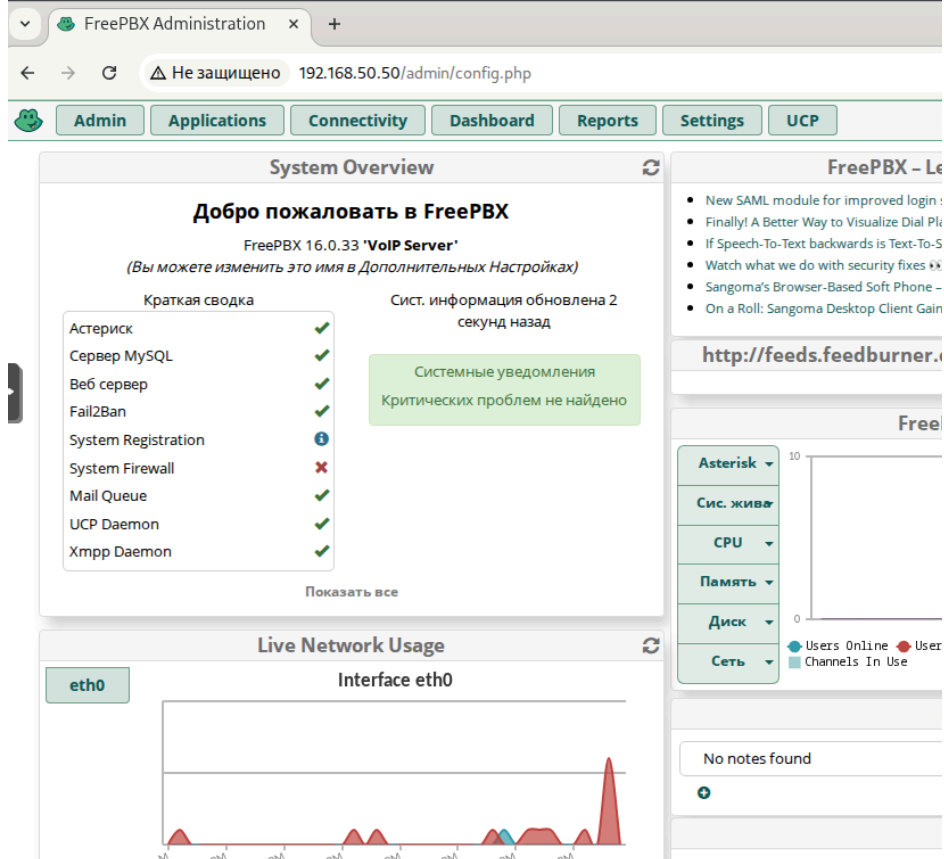
- Выбираем **FreePBX Administration**:



- Выполняем вход из-под созданной учётной записи:



- Результат успешного входа:



- Для реализации внутренних звонков необходимо использовать CHAN_SIP:
 - переходим в **Settings** -> **Advanced Settings**:
 - ищем параметр **SIP Channel Driver** и выбираем **chan_sip** и нажимаем **Submit**

SIP Channel Driver ? ↻

- переходим в **Settings** -> **Asterisk SIP Settings**:
- ищем параметр **Bind Port** и меняем с 5160 на **5060** и нажимаем **Submit**

Bind Port ?

5060

- Нажимаем **Apply Config**, после чего перезагружаем **sip-cod** целиком:

- Переходим в **Applications** -> **Extensions**:

All ExtensionsCustom ExtensionsDAHDi ExtensionsIAX2 Extensions

+ Add Extension ▾

⚡ Quick Create Extension

✕ Delete

+ Add New Custom Extension

+ Add New DAHDi Extension

+ Add New IAX2 Extension

+ Add New SIP (Legacy) [chan_sip] Extension

+ Add New Virtual Extension

CW

DND

FM

No matching

- Создаём внутренний номер и нажимаем **Submit**:

Add SIP Extension1001

GeneralVoicemailFind Me/Follow MeAdvancedPin Sets

— Add Extension

This device uses **CHAN_SIP** technology listening on Port 5060 (UDP)

User Extension ?

1001

Display Name ?

1001

Outbound CID ?

1001

Emergency CID ?

1001

Secret ?

P@ssw0rd

Really Weak

- Результат создания внутреннего номера:

All ExtensionsCustom ExtensionsDAHDi ExtensionsIAX2 ExtensionsSIP (Legacy) [chan_sip]

+ Add Extension ▾

⚡ Quick Create Extension

✕ Delete

	Extension	Name	CW	DND	FM/FM
<input type="checkbox"/>	1001	1001	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Showing 1 to 1 of 1 rows

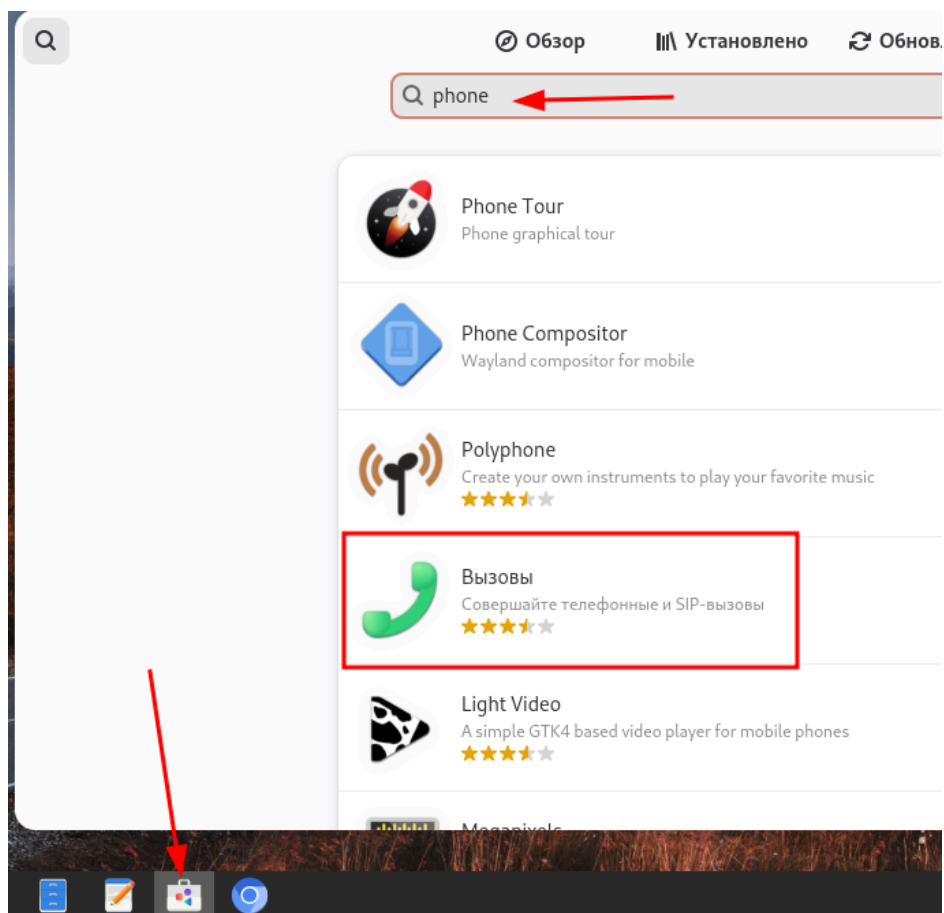
- Аналогичным образом создаём все необходимые по заданию номера:

	Extension	Name	CW	DND	FM/FM	CF
	1001	1001	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1002	1002	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	2001	2001	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	2002	2002	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

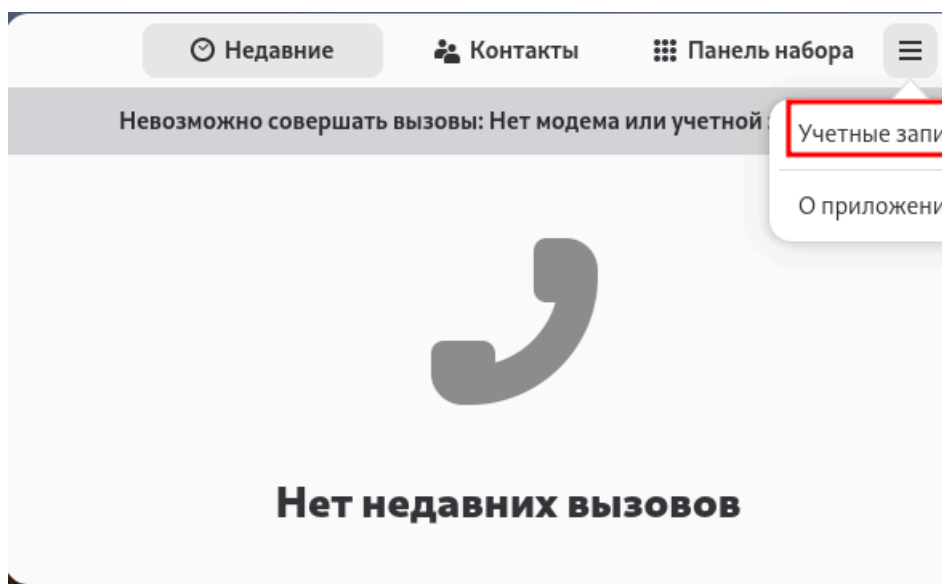
- Нажимаем **Apply Config**, после чего перезагружаем **sip-cod** целиком.

admin-cod, cli-cod, cli1-а и cli-а (alt-workstation):

- Устанавливаем СофтФон:



- Подключаемся к sip-cod:
 - открываем установленный СофтФон





Добавление учетных записей VoIP

Здесь вы можете добавить учетную запись VoIP. Это позволит вам совершать и принимать VoIP-вызовы с использованием протокола SIP. Эта функция все еще относительно новая и еще не завершена (т.е. нет зашифрованных носителей).

Добавить учетную запись

- в зависимости от ВМ указываем соответствующие номера:

Добавить новую учетную ...

Войти

x

Сервер
192.168.50.50



Отображаемое имя (необязательно)
admin-cod



ID пользователя
1001



Пароль
P@ssw0rd



Порт
5060



Транспорт

UDP ▾

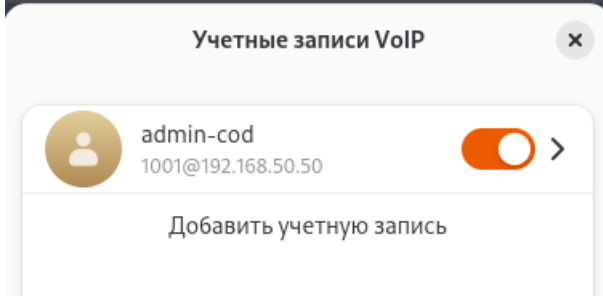
Шифрование информации Без шифрова... ▾

Использовать для телефонных вызовов

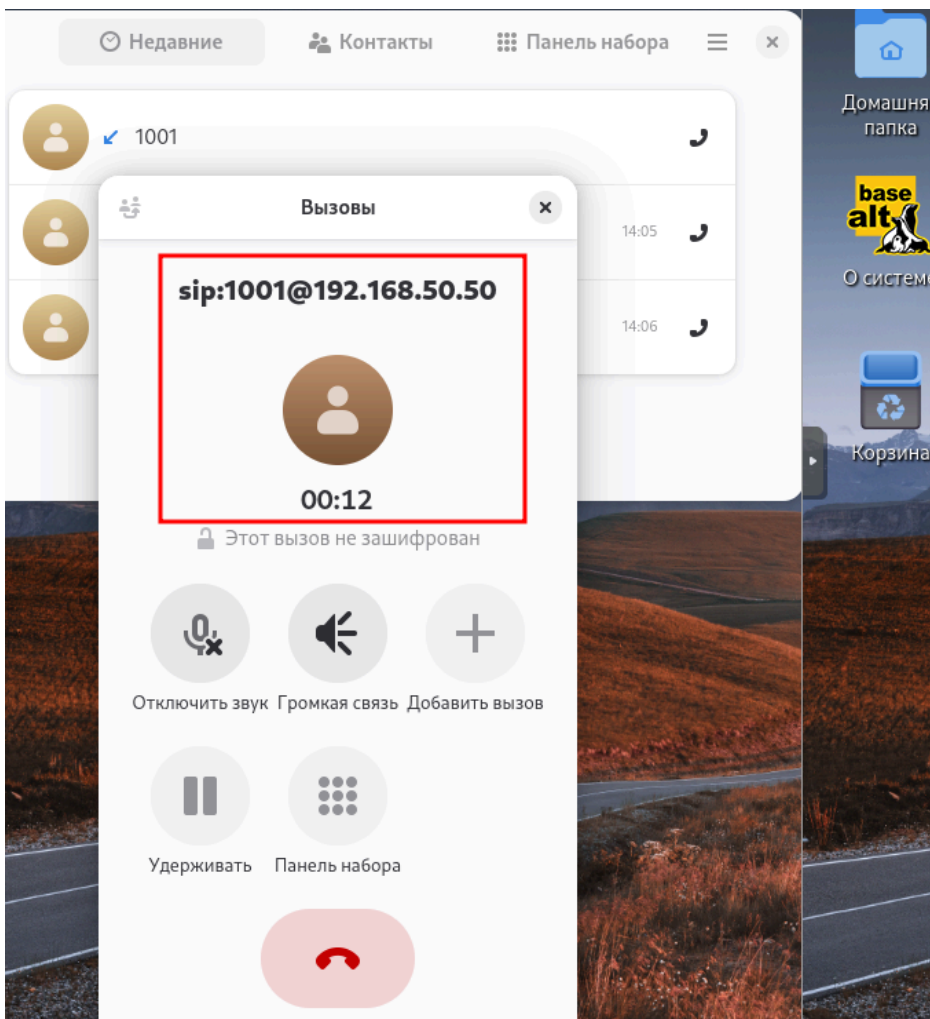
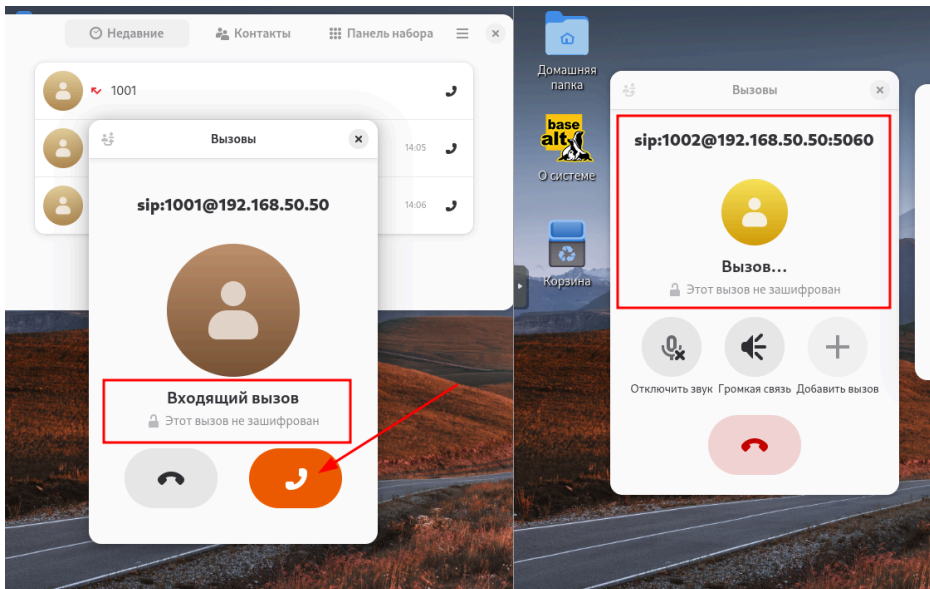


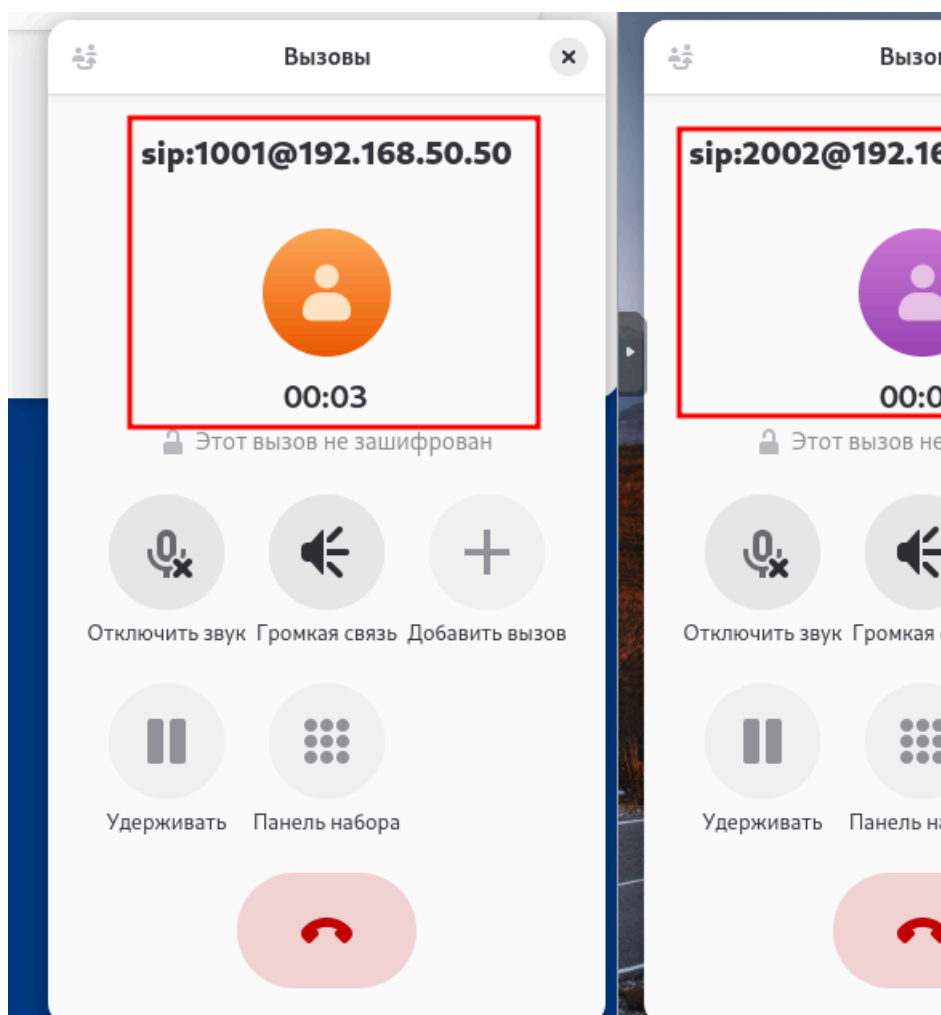
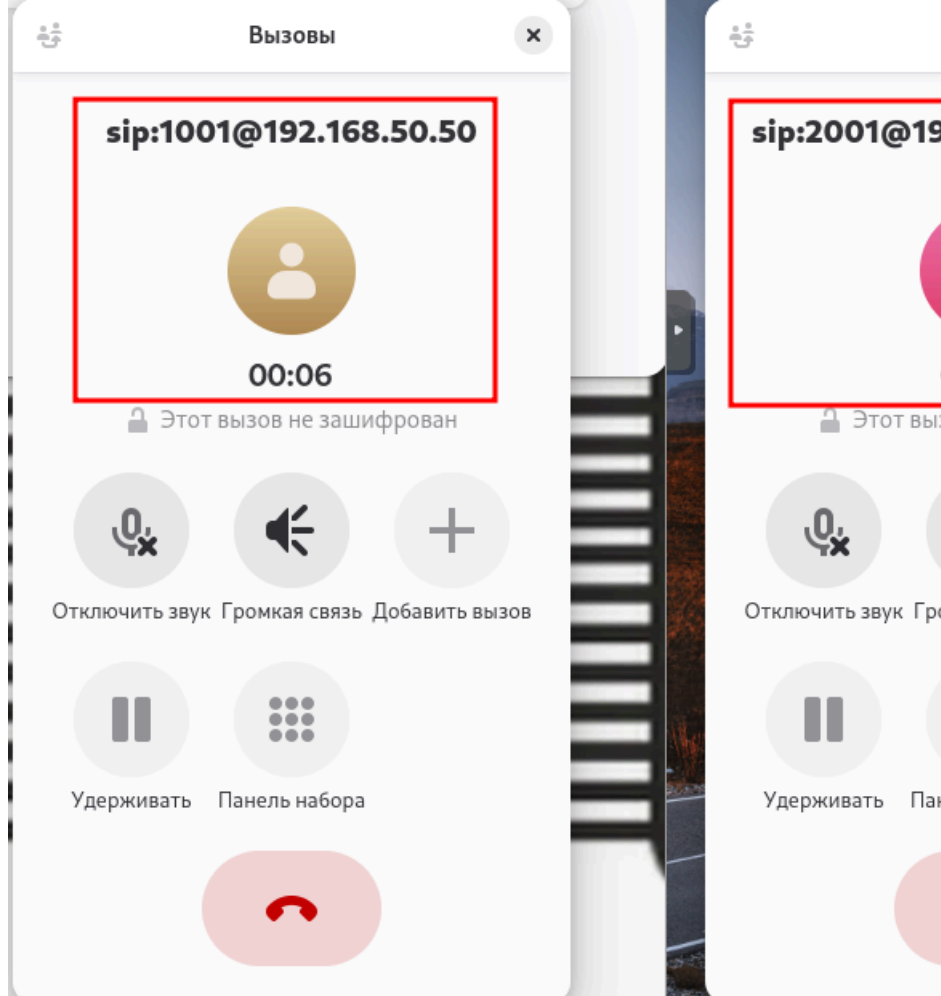
Автоматическое подключение














- Проверяем звонки:





- Итог:

<div>🕒 Недавние</div> <div>👥 Контакты</div> <div>☰ Панель набора</div>		
	 2002	14:23 
	 2001	14:20 
	 1002	14:07 

Последнее изменение: пятница, 21 ноября 2025, 14:17

Предыдущий
элемент курса

22. Настройка системы
мониторинга

Перейти на...

Следующий элемент
курса

1. Подготовка машины
Cloud-ADM

Обратная связь

Подпишитесь

Вы используете гостевой
доступ (Вход)

Сводка хранения данных

Тема оформления сайта разработана

conecti.me

```
ecorouter#show ntp date  
Wed Jul 13 12:08:23 UTC 2016
```



18 PTP

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, EcoRouter поддерживает только вторую, т. е. PTPv2. Протокол PTP работает по принципу master-slave, т. е. в одной схеме синхронизации должен присутствовать источник (master) и приёмник синхронизации (slave). Устройства, которые не являются источником или приёмником 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 передаёт 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.

Перед настройкой необходимо включить поддержку PTP на устройстве. Для этого необходимо произвести следующие действия.

- Выполнить в конфигурационном режиме команду `enable ptp`.
- Сохранить конфигурацию.
- Перезагрузить устройство.

```
ecorouter(config)#enable ptp
Changes will be applied after reboot. Please save config and reload.
ecorouter(config)#enable ptp
Changes will be applied after reboot. Please save config and reload.
ecorouter(config)#ptp mode transparent-e2e udp
% PTP is not enabled yet: reload required. Please save config and reload.
ecorouter(config)#write
Building configuration...
ecorouter(config)#exit
ecorouter#reload
reboot system? (y/n): y
...reboot...
ecorouter login: admin
Password:
User Access Verification
EcoBNGOS version 3.2.5 EcoRouter 07/02/19 13:48:51
ecorouter>show running-config
...
hw mgmt ip 192.168.255.1/24
!
enable ptp
!
ip vrf management
...
ecorouter>enable
ecorouter#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#enable ptp
PTP has already been enabled.
```

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 43 — Параметры команды `ptp mode`

Параметр	Описание
<code>transparent boundary</code>	Тип часов. transparent — transparent clock; boundary — boundary clock
<code>e2e p2p</code>	Режим работы протокола PTPv2. e2e — End-to-End режим; p2p — Peer-to-Peer режим
<code>ethernet udp</code>	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 порте PTPv2 имеет вид:

```
ptp {transparent|slave|master|bmca}
```

В результате выполнения этой команды на соответствующем порте будет включён протокол PTPv2 в режиме **transparent**, **slave**, **master** или будет включён 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>
```


18.1 Команды просмотра

Таблица 44 — Команды просмотра PTP

Команда и результат ее выполнения	Комментарий
<code>show ptp status</code>	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
<code>show ptp boundary-clock</code>	Показать подробную информацию PTP (только для типа boundary)
ge3: State: slave Assigned by: static Grandmaster ID: 1c8776ffffe4005a1 Priority: N/A Offset: 456 ns Path Delay: 783 ns	Порт, информация о котором показана Режим порта Способ задания режима порта (static/bmc) ID Grandmaster часов Приоритет часов. Используется для BMC (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

19 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.

Таблица 4.5 — Полосы пропускания CoPP, заданные по умолчанию

Протокол	Количество пакетов в секунду
Входящий ARP	128
Входящий BGP	512
Входящий DHCP-Discovery	1024
Входящий DHCP-Other	1024
Входящий ICMP	1024
Входящий IS-IS	512
Входящий LDP	512
Входящий Multicast-IGMP	128
Входящий Multicast-Other	4096
Входящий Multicast-PIM	512
Входящий non-IP	256
Входящий OSPF	512
Входящий Other	8192
Входящий SNMP	128
Входящий SSH	512
Исходящий ICMP	1024
Исходящий Other	1024

В CLI пользователь может ограничить полосу пропускания трафика для протоколов, перечисленных в таблице, в CP маршрутизатора. Настройки защиты от DoS и DDoS атак доступны на интерфейсах и портах, а также и глобально на CP 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 46 — Команды ограничения полосы пропускания

Команда	Режимы	Описание
<code>rate-limit dhcp-discovery <0-262144></code>	(config-cp), (config-port), (config-port-channel), (config-int)	Общее ограничение полосы пропускания сообщений DHCP Discovery от всех клиентов
<code>rate-limit dhcp-other <0-4096></code>	(config-cp)	Общее ограничение входной полосы пропускания всех сообщений DHCP от всех клиентов
<code>rate-limit dhcp-discovery per-interface <0-262144></code>	(config-int)	Общее ограничение полосы пропускания сообщений DHCP Discovery на интерфейсе от всех клиентов
<code>rate-limit dhcp-discovery per-subscriber <0-15></code>	(config-int)	Ограничение полосы пропускания сообщений DHCP Discovery от одного клиента
<code>rate-limit arp <0-524288></code>	(config-cp), (config-port), (config-port-channel), (config-int)	Общее ограничение полосы пропускания сообщений ARP Request от всех клиентов
<code>rate-limit arp per-interface <0-524288></code>	(config-int)	Общее ограничение полосы пропускания сообщений ARP Request на интерфейсе от всех клиентов
<code>rate-limit arp per-subscriber <0-524288></code>	(config-int)	Ограничение полосы пропускания сообщений ARP Request от одного клиента
<code>rate-limit bgp <0-4096></code>	(config-cp)	Общее ограничение входной полосы пропускания BGP трафика
<code>rate-limit icmp <0-2048> (in out)</code>	(config-cp)	Общее ограничение полосы пропускания для ICMP трафика в различных направлениях
<code>rate-limit isis <0-4096></code>	(config-cp)	Общее ограничение входной полосы пропускания IS-IS трафика
<code>rate-limit ldp <0-4096></code>	(config-cp)	Общее ограничение входной полосы пропускания LDP трафика

Команда	Режимы	Описание
<code>rate-limit multicast-igmp <0-262144></code>	(config-cp)	Общее ограничение входной полосы пропускания IGMP трафика
<code>rate-limit multicast-other <0-262144></code>	(config-cp)	Общее ограничение входной полосы пропускания мультикастного трафика
<code>rate-limit multicast-pim <0-262144></code>	(config-cp)	Общее ограничение входной полосы пропускания PIM трафика
<code>rate-limit non-ip <0-4096></code>	(config-cp)	Общее ограничение входной полосы пропускания для любого не IP трафика от всех клиентов
<code>rate-limit ospf <0-4096></code>	(config-cp)	Общее ограничение входной полосы пропускания OSPF трафика
<code>rate-limit other <0-524288> (in out)</code>	(config-cp)	Общее ограничение полосы пропускания для юникастового трафика в различных направлениях
<code>rate-limit snmp <0-512></code>	(config-cp)	Общее ограничение входной полосы пропускания SNMP трафика
<code>rate-limit ssh <0-2048></code>	(config-cp)	Общее ограничение входной полосы пропускания SSH трафика

В случае превышения rate-limit по ARP или DHCP с одного MAC-адреса, подозрительный трафик от абонента блокируется на 30 секунд.****

19.1 Команды просмотра

Для просмотра текущего состояния счётчиков политики уровня управления необходимо в режиме администрирования выполнить команду `show counters copp`.

```
ecorouter#show counters copp
```

Received

	rate limit	packets	bytes	dropped
OSPF	512	182483	12718584	0
ISIS	512	0	0	0
LDP	512	42	2058	0
ARP	2048	2	92	0
IGMP	128	689758	31887634	0

PIM	512	45491	2638478	0
SNMP	128	45326	3550662	0
SSH	4096	213469	46415291	849
ICMP	1024	25399	5731432	0
BGP	512	81	4046	0
DHCP	1024	3399	1165613	0
DHCP DISC	1024	322	110891	0
MCAST	4096	3693916	946661169	0
L2	256	109178	5022188	0
Other	8192	705552	36033915	0

Transmitted

	rate limit	packets	bytes	dropped
ICMP	1024	34622545	1938862520	29433
Other	8192	2864904	125315112	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 счётчиков необходимо выполнить команду `clear counters copp` в режиме конфигурирования.

```
ecorouter(config)#clear counters copp
```



20 Маршрутизация Unicast

20.1 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 счёт 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 EcoRouterOS доступно максимум восемь ECMP маршрутов. Если количество ECMP маршрутов превышает восемь, то в FIB устанавливаются первые 8 nexthop, остальные присутствуют только в RIB таблице.

Данный сценарий отображается в выводе команды `show ip route database`.

```
ecorouter#show ip route database
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
...
> - selected route, * - FIB route, p - stale info, b - BMI route
IP Route Table for VRF "default"
S *> 1.1.1.1/32 [1/0] via 10.1.1.2, e1
  *>           [1/0] via 10.1.1.3, e1
  *>           [1/0] via 10.1.1.4, e1
  *>           [1/0] via 10.1.1.5, e1
  *>           [1/0] via 10.1.1.6, e1
  *>           [1/0] via 10.1.1.7, e1
  *>           [1/0] via 10.1.1.8, e1
  *>           [1/0] via 10.1.1.9, e1
  >           [1/0] via 10.1.1.10, e1
  >           [1/0] via 10.1.1.11, e1
```

Глубина рекурсии в EcoRouterOS равна трём. После трёх лукапов маршрут должен быть доступен из непосредственно подключённой сети (directly connected).

Маршрут неудовлетворяющий этим правилам будет отброшен.

Пример:

```
ip route 1.1.1.1/32 10.1.1.2
ip route 1.1.1.1/32 10.1.1.3
```



```
ip route 1.1.1.1/32 10.1.1.4
ip route 1.1.1.1/32 10.1.1.5
ip route 1.1.1.1/32 10.1.1.6
ip route 1.1.1.1/32 10.1.1.7
ip route 1.1.1.1/32 10.1.1.8
ip route 1.1.1.1/32 10.1.1.9
ip route 1.1.1.1/32 10.1.1.10
ip route 1.1.1.1/32 10.1.1.11
ip route 4.4.4.4/32 1.1.1.1
ip route 4.4.4.4/32 10.1.1.100
ip route 4.4.4.4/32 10.1.1.101
ip route 5.5.5.5/32 4.4.4.4
```

Маршрут 5.5.5.5 будет доступен только через 10.1.1.100 и 10.1.1.101.

Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP

...

> - selected route, * - FIB route, p - stale info, b - BMI route

IP Route Table for VRF "default"

```
S *> 1.1.1.1/32 [1/0] via 10.1.1.2, e1
    *>          [1/0] via 10.1.1.3, e1
    *>          [1/0] via 10.1.1.4, e1
    *>          [1/0] via 10.1.1.5, e1
    *>          [1/0] via 10.1.1.6, e1
    *>          [1/0] via 10.1.1.7, e1
    *>          [1/0] via 10.1.1.8, e1
    *>          [1/0] via 10.1.1.9, e1
    >          [1/0] via 10.1.1.10, e1
    >          [1/0] via 10.1.1.11, e1
S *> 4.4.4.4/32 [1/0] via 1.1.1.1 (recursive *via 10.1.1.2
                                                *via 10.1.1.3
                                                *via 10.1.1.4
                                                *via 10.1.1.5
                                                *via 10.1.1.6
                                                *via 10.1.1.7
                                                *via 10.1.1.8
                                                *via 10.1.1.9
                                                via 10.1.1.10
                                                via 10.1.1.11)
    >          [1/0] via 10.1.1.100, e1
    >          [1/0] via 10.1.1.101, e1
S *> 5.5.5.5/32 [1/0] via 4.4.4.4 (recursive *via 10.1.1.100
                                                *via 10.1.1.101)
```

В документации можно найти подробные инструкции по настройке для каждого протокола.

Таблица 47 — Значения по умолчанию административных дистанций

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

20.2 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.

Обратите внимание, что статический маршрут может присутствовать в RIB/FIB, когда физический порт, через который доступен следующий узел, отключён или неработоспособен. Это актуально, когда логический интерфейс является BMI (BRAS) или BDI (Bridge) интерфейсом.

20.2.1 Базовая настройка статических маршрутов

Настройка происходит в режиме конфигурации.

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

Где: 192.168.1.0 — адрес целевой подсети, 255.255.255.0 — её маска, 172.16.10.1 — адрес интерфейса следующего транзитного узла (маршрутизатора), за которым расположена целевая подсеть.

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.

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

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

20.2.2 Административная дистанция статических маршрутов

Административная дистанция — это степень надёжности источника маршрутной информации. Административная дистанция используется маршрутизаторами для определения приоритетного маршрута при наличии двух и более различных маршрутов до одной цели по различным протоколам маршрутизации.

По умолчанию, статический маршрут имеет административную дистанцию равную единице, что даёт 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.

Пример использования:

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

Если есть динамические маршруты с административной дистанцией 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.

20.3 Протокол RIP

Routing Information Protocol (RIP) — это дистанционно-векторный протокол динамической маршрутизации, используемый для динамического обновления таблиц маршрутизации. RIP определяет оптимальные пути передачи данных между узлами на основе метрики, которая измеряется количеством переходов (хопов) до целевой сети. Протокол работает на прикладном уровне модели OSI и использует UDP-порт 520 для обмена информацией между маршрутизаторами.

В EcoRouterOS поддерживается RIP версии 2.

20.3.1 Метрика RIP

Основной метрикой протокола RIP является количество переходов между узлами сети. Для вычисления кратчайшего маршрута до сети назначения использует алгоритм Беллмана-Форда. Данный алгоритм не учитывает загруженность канала и пропускную способность интерфейсов устройств. Лучшим маршрутом, который будет помещён в таблицу маршрутизации, считается маршрут с минимальным возможным значением метрики.

Административная дистанция протокола по умолчанию равна 120.

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

20.3.2 Таймеры RIP

По умолчанию маршрутизатор под управлением протокола RIP рассылает пакеты с обновлением маршрутной информации каждые 30 секунд (update timer) с небольшим временным отклонением.

Маршрут считается недостижимым если в течение 6 интервалов по 30 секунд (invalid timer) маршрутизатор не получил обновление маршрутной информации. Такой маршрут помечается максимальным допустимым в протоколе RIP количеством переходов — 16 (invalid, метрика 16), что считается недостижимым расстоянием.

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 секундам.

Таблица 48 — Допустимые значения и значения по умолчанию для таймеров

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.

20.3.3 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.

20.3.4 Функция ручной суммаризации маршрутов

Суммаризация или агрегация (объединение) маршрутов — это метод объявления общей сети, которая включает в себя более конкретные сети или подсети. Суммирование применяется для минимизации размера таблицы маршрутизации и нагрузки на маршрутизатор, а также для уменьшения объёмов объявлений маршрутной информации, то есть экономит пропускную способность сети.

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.

20.3.5 Команды настройки

Команды настройки протокола RIP:

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

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

20.3.6 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 присоединённых 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 присоединённых 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
```

20.3.7 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
```

20.3.8 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)
```

20.4 Протокол OSPF

OSPF (Open Shortest Path First) — это протокол динамической маршрутизации, отслеживающий состояние каналов (Link-State) и использующий для нахождения оптимального пути передачи данных алгоритм Дейкстры.

Для повышения эффективности и масштабируемости OSPF поддерживает иерархическую структуру сети, разделяя её на зоны (areas), что позволяет уменьшить нагрузку на устройства и упростить управление крупными сетями. Главная зона, называемая Area 0 (или backbone, магистральная зона), связывает все остальные зоны и обеспечивает их взаимодействие.

OSPF наряду с такими протоколами как RIP, IS-IS и EIGRP является IGP (Interior Gateway Protocol) протоколом, т.е. обеспечивает маршрутизацию внутри одной автономной системы (autonomous system, AS) или сети с единой политикой маршрутизации.

Маршрутизаторы OSPF обмениваются информацией о состоянии соединений, "стоимости" передачи данных по тому или иному соединению и другой информацией со своими соседями. Обмен информацией позволяет всем участвующим маршрутизаторам выстроить карту топологии сети. Каждый маршрутизатор применяет к полученной карте алгоритм кратчайшего пути Дейкстры (SPF) для расчёта оптимального пути к каждому пункту назначения в сети.

Основные операции выполняемые протоколом OSPF:

- Маршрутизатор с настроенным OSPF отправляет приветственные (Hello) сообщения через каждый интерфейс, на котором активирован OSPF. Если маршрутизатор получает корректное приветственное сообщение на интерфейсе с OSPF, он устанавливает отношения смежности (adjacency) с другими маршрутизаторами OSPF в этой сети.
- Если интерфейс подключён к широковещательной сети, такой как Ethernet, маршрутизаторы используют пакеты Hello для выбора назначенного маршрутизатора (Designated Router, DR) и резервного назначенного маршрутизатора (Backup Designated Router, BDR) в этой сети.
- Маршрутизаторы обмениваются пакетами описания базы данных (Database Description packets). Эти пакеты содержат индекс всех объявлений о состоянии соединений (Link-State Advertisements, LSAs), которые есть в топологической базе данных маршрутизатора.

- На основе содержимого пакетов с описанием базы данных каждый маршрутизатор запрашивает LSAs, которые необходимы для обновления его топологической базы данных. Запрос выполняется с помощью пакета запроса состояния канала (Link-State Request, LSR).
- По запросу маршрутизатор отправляет обновление состояния соединения (Link-State Update, LSU), содержащее LSAs, запрошенные соседом. В ответ на каждое LSU маршрутизатор отправляет пакет подтверждения состояния соединения (Link-State Acknowledgment).
- После завершения обмена LSU маршрутизаторы становятся полностью смежными. Они продолжают обмениваться периодическими приветственными сообщениями для поддержания смежности.
- Если происходит изменение топологии, затронутые изменением маршрутизаторы передают обновлённое LSA, отражающее это изменение. Каждый маршрутизатор OSPF обновляет свою базу данных состояния каналов (Link-State Database, LSDB), распространяет новый LSA своим соседям и запускает алгоритм кратчайшего пути (Shortest Path First, SPF) для пересчёта своей таблицы маршрутизации.
- LSAs устаревают в LSDB и считаются устаревшими через 3,600 секунд (1 час). Маршрутизатор, который создал LSA, повторно распространяет его после достижения возраста ~1,800 секунд, чтобы обновить его в LSDB.

20.4.1 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 ко 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.

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

Тип сети задаётся 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 весовые коэффициенты интерфейсов, если нужно повлиять на выбор оптимального маршрута: укажите значение в режиме конфигурирования интерфейсов командой `ip ospf cost <значение>`. Для изменения множителя в формуле расчёта 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 определённой зоне в режиме конфигурирования протокола маршрутизации с помощью команды `area authentication`.

20.4.2 Пример настройки

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

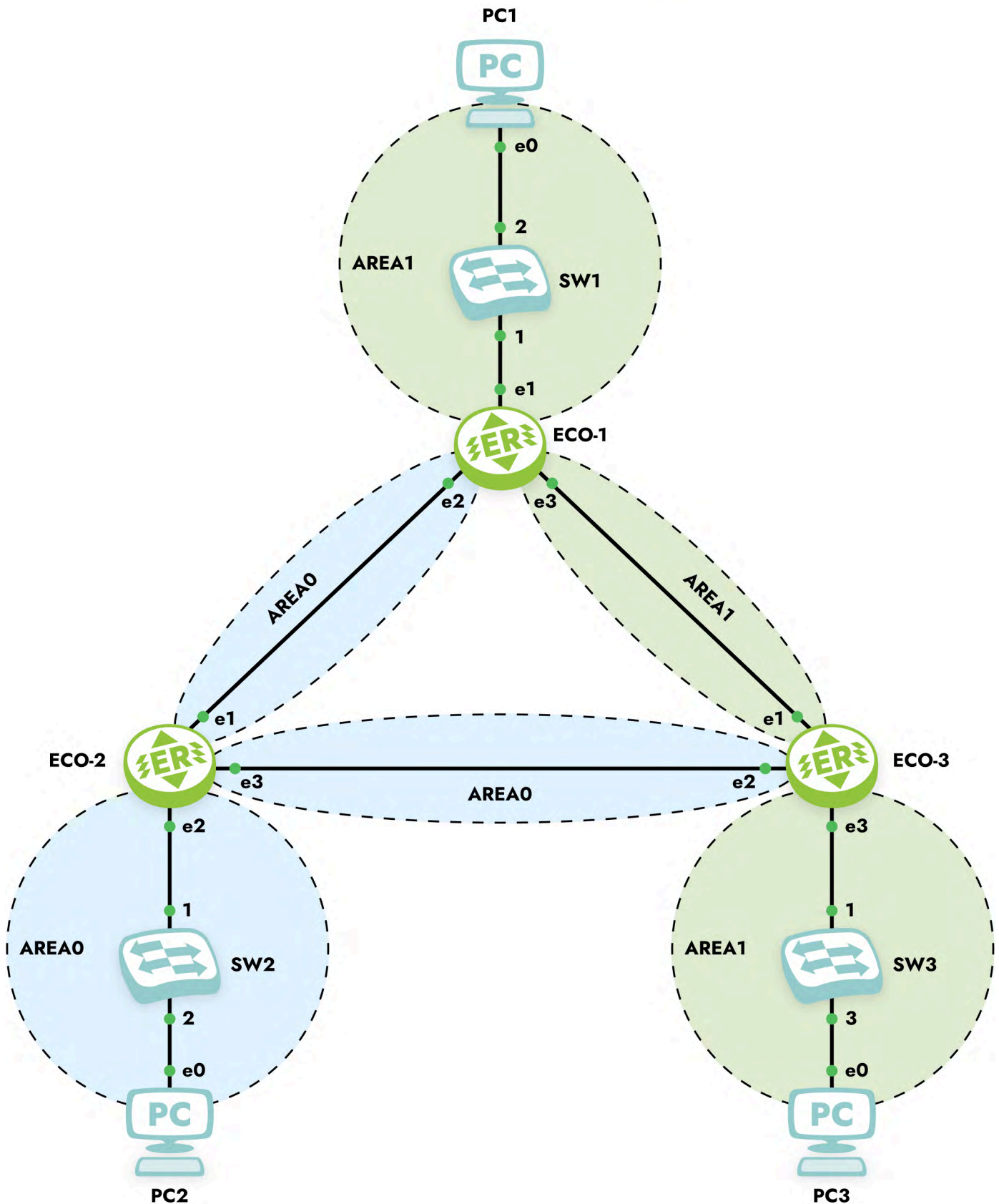


Рисунок 17

Пример конфигураций маршрутизаторов

ECO-1

Шаг 1. Задание имени устройства.



```
hostname ECO-1
```

Шаг 2. Настройка портов, интерфейсов и сервисных интерфейсов.

```
enable
configure

interface e1
  ip address 10.10.0.1/16

interface e2
  ip address 10.12.0.1/16

interface e3
  ip address 10.13.0.1/16

port ge1
  service-instance ge1/e1
  encapsulation untagged
  connect ip interface e1

port ge2
  service-instance ge2/e2
  encapsulation untagged
  connect ip interface e2

port ge3
  service-instance ge3/e3
  encapsulation untagged
  connect ip interface e3
```

Шаг 3. Включение маршрутизации и объявление присоединённых 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 ge1
service-instance ge1/e1
encapsulation untagged
connect ip interface e1
port ge2
service-instance ge2/e2
encapsulation untagged
connect ip interface e2
port ge3
service-instance ge3/e3
encapsulation untagged
connect ip interface e3
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 ge1
service-instance ge1/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
```

20.4.3 Аутентификация

В общедоступной сети неавторизованному устройству легко эмулировать маршрутизатор OSPF и потенциально нарушить работу сети, предоставив ложную информацию. В OSPFv2 предусмотрена возможность настройки аутентификации между маршрутизаторами-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 зоны (area). Также при создании authentication-key необходимо выбрать, в каком виде ключ будет передаваться между соседями: в открытом виде или с использованием хэшированного ключа MD5.

Таблица 49 — Команды конфигурирования

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 md5	(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 приведённой 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 применён 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)#router ospf 1
(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.

20.4.4 Фильтрация 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:

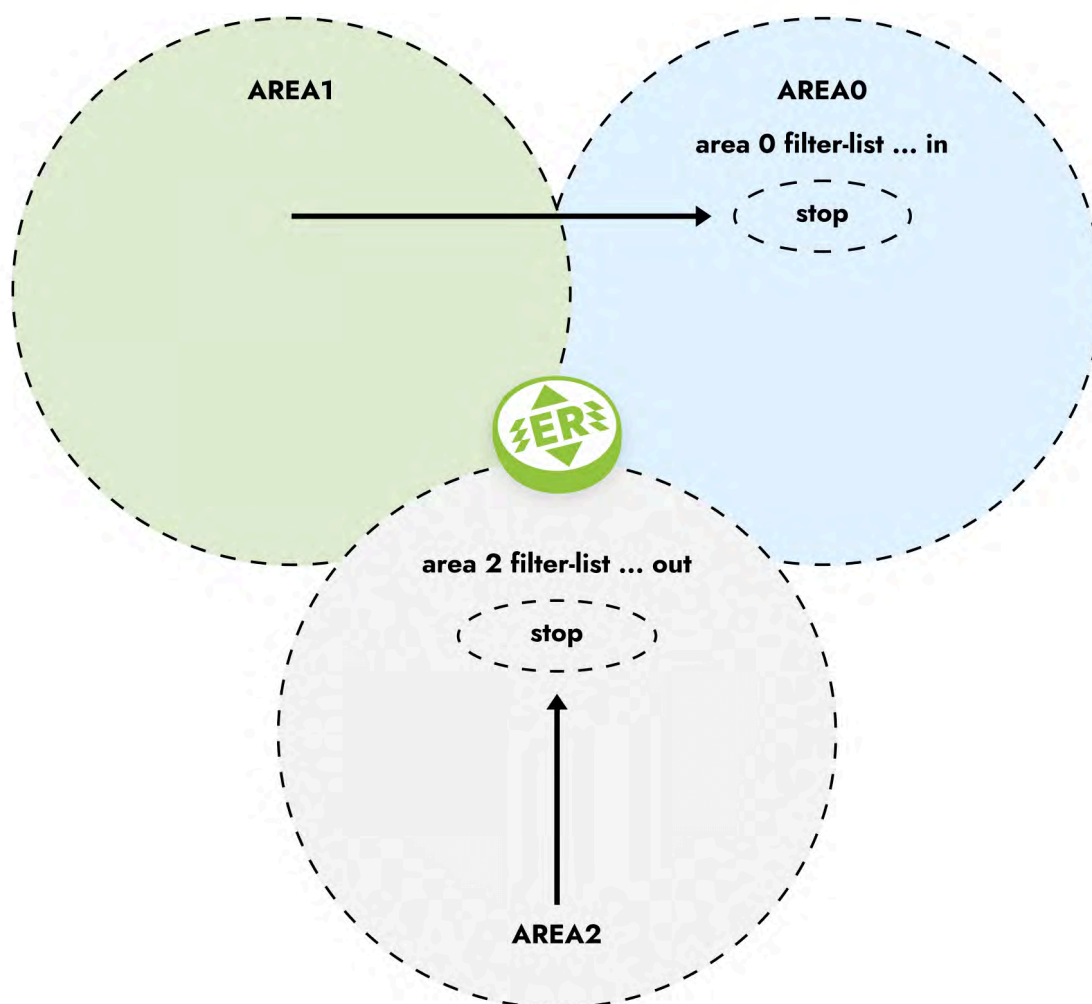


Рисунок 18

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 определённым 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`.

20.4.5 Суммирование маршрутов OSPF

В идеале топология сети и план адресации должны быть спроектированы так, чтобы в объявлении ABR одна запись могла описывать целый диапазон адресов. Route Summarization, или суммирование маршрутов, позволяет объединять несколько подсетей в одну общую запись LSA (Link State Advertisement), чтобы уменьшить размер таблиц маршрутизации и снизить объём передаваемых служебных данных.

Суммирование настраивается на ABR маршрутизаторе расположенном в той зоне, которую необходимо суммировать. При суммировании идентификатор состояния соединения (link-state ID) в объявлении LSA принимает значение суммированного маршрута зоны.

Суммирование происходит на ABR и ASBR маршрутизаторах. Внутри зоны суммирование невозможно. Для рассылки суммарного маршрута используются Summary LSA (Тип 3 — межзональное суммирование) и External LSA (Тип 5 — суммирование маршрутов из других автономных систем).

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 суммировании 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 метрика.

Если производится суммирование маршрутов, то OSPF предполагает, что все маршруты подпадающие под префикс суммированного маршрута как минимум существуют. Но некоторых подсетей может просто не существовать за суммированным адресом. Тогда данные отправленные в эти подсети будут перенаправлены на маршрут по умолчанию. Чтобы этого избежать, при создании суммированного маршрута в таблицу маршрутизации автоматически добавляется нулевой (Null)-маршрут. Это сделано для того, чтобы данные направленные в несуществующие подсети и на сам суммированный маршрут сразу же отбрасывались. При этом трафик в более точно заданные и существующие сети отброшен не будет, поскольку на ABR-маршрутизаторе есть маршруты до этих сетей.

20.4.6 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** задаёт 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.

20.4.7 Зоны OSPF

В OSPF есть разные типы зон: обычные, stub, totally stubby, NSSA. Stub зоны блокируют внешние маршруты (LSA типа 5), но разрешают маршруты внутри автономной системы. Totally stubby блокируют и внешние, и межзональные маршруты, оставляя только маршрут по умолчанию. NSSA (Not-So-Stubby Area) — это особая зона, которая позволяет импортировать внешние маршруты через LSA типа 7, которые затем конвертируются в LSA типа 5 на ABR.

При правильном дизайне OSPF сети для уменьшения размера базы данных топологии может потребоваться использование тупиковых зон OSPF. EcoRouterOS поддерживает эту функциональность.

Таблица 50 — Тупиковые зоны OSPF

Тип области	ABR передаёт LSA type 5 в область?	ABR передаёт LSA type 3 в область?	Позволена редистрибуция в тупиковую зону?	Команда конфигурирования
Stubby	Нет	Да	Нет	<code>area <номер> stub</code>
Totally stubby	Нет	Нет	Нет	<code>area <номер> stub no-summary</code>
NSSA	Нет	Да	Да	<code>area <номер> nssa</code>
Totally NSSA	Нет	Нет	Да	<code>area <номер> nssa no-summary</code>

20.4.8 Редистрибуция OSPF

Редистрибуция (перераспределение маршрутов) в OSPF — это процесс импорта маршрутов из других источников (например, статических маршрутов, RIP, EIGRP, BGP) в OSPF-домен. Это позволяет OSPF распространять информацию о сетях, которые изначально не были частью его маршрутной базы. Редистрибуция выполняется на ASBR (Autonomous System Boundary Router) — маршрутизаторе, который подключён к другой автономной системе или протоколу.

Если сеть использует OSPF как основной протокол, но есть участки с другим протоколом, редистрибуция позволяет объединить их в общую маршрутную таблицу. Важно отметить, что редистрибуция требует настройки на маршрутизаторе, который является граничным между разными протоколами (ASBR в терминах OSPF).

При редистрибуции метрики маршрутов могут рассчитываться двумя способами: E1 (External Type 1):

- Метрика рассчитывается как внутренняя стоимость OSPF до ASBR + внешняя метрика.
- Используется, когда важно учитывать «расстояние» до ASBR.

E2 (External Type 2):

- Метрика равна только внешней метрике (по умолчанию).
- Используется, если внутренние метрики OSPF не должны влиять на выбор пути.

Редистрибуция из различных протоколов маршрутизации, статических и непосредственно подключённых маршрутов в OSPF может быть настроена в режиме конфигурирования маршрутизатора 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** задаёт значение метрики, параметр **metric-type** указывает тип метрики OSPF, параметр **route-map** ссылается на условия в карте маршрутов, **tag** — тегировать редистрибутированные сети. С помощью команды `default-metric` можно задать значение для всех редистрибутированных маршрутов. Команда `distance` задаёт значение административной дистанции для протокола OSPF.

20.4.9 Виртуальные соединения и Multi-Area соседства

Магистральная зона в автономной системе OSPF должна быть непрерывной, а все остальные зоны должны быть подключены к магистральной. Иногда это непрактично или неоправданно затратно в реализации. Виртуальные соединения могут использоваться для подключения зоны к магистральной через немагистральную зону.

Транзитная зона должна иметь полную информацию о маршрутизации и, следовательно, не может быть тупиковой областью или неполностью тупиковой (NSSA).

Виртуальные соединения считаются частью магистральной и ведут себя так, как если бы они были ненумерованными сетями точка-точка между двумя маршрутизаторами. Когда маршрутизатор полностью смежный со своим соседом по виртуальному каналу, он устанавливает бит V в своём Router LSA.

Виртуальное соединение настраивается на обеих конечных точках. Настроенное на одной точке виртуальное соединение идентифицируется посредством RID (идентификатора маршрутизатора) другой конечной точки. Оба конечных маршрутизатора должны быть подключены к общей транзитной области. Аналогичная конфигурация должна быть произведена с другой стороны виртуального соединения.

Виртуальное соединение 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 соединения используйте в режиме конфигурирования маршрутизатора команду `area <номер> virtual-link <ip-адрес>`, где **номер area** — это зона, через которую создаётся виртуальное соединение, **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.

20.4.10 Команды просмотра OSPF

Таблица 51 — Команды просмотра OSPF

Команда	Описание
<code>show ip route ospf</code>	Просмотр маршрутов в таблице маршрутизации полученных через OSPF



Команда	Описание
<code>show ip ospf neighbor</code>	Просмотр сведений о соседских отношениях между OSPF маршрутизаторами
<code>show ip ospf interface</code>	Просмотр данных о состоянии и сконфигурированных настройках на интерфейсах, участвующих в OSPF процессе
<code>show ip protocols</code>	Просмотр информации о запущенных процессах маршрутизации
<code>show ip ospf database</code>	Просмотр базы данных OSPF топологии
<code>show ip ospf virtual-links</code>	Просмотр информации о OSPF виртуальной линке
<code>show ip ospf border-routers</code>	Просмотр информации о пограничных маршрутизаторах
<code>show ip ospf multi-area-adjacencies</code>	Просмотр информации о multi-area соседях
<code>show ip ospf</code>	Просмотр сведений о OSPF процессах запущенных на маршрутизаторе

20.4.11 Дополнительные команды конфигурирования OSPF

Таблица 52 — Дополнительные команды конфигурирования OSPF

Команда	Режим	Описание
<code>capability restart graceful</code>	(config)#	Включение функционала мягкого перезапуска (graceful restart)
<code>max-concurrent-dd <1-65535></code>	(config)#	Количество одновременно обработанных дескрипторов БД (DD)
<code>maximum-area <1-4294967294></code>	(config)#	Максимально возможное количество областей
<code>ospf flood-reduction</code>	(config)#	Уменьшение сигнальной нагрузки путём установки DNA бита
<code>overflow database</code>	(config)#	Уменьшение максимального количества объявлений о состоянии канала (LSA), которые могут быть обработаны
<code>timers lsa arrival <0-600000></code>	(config)#	Установка минимального интервала приёма того же LSA от соседа

Команда	Режим	Описание
<code>ip ospf database-filter all out</code>	(config-int)#	Отключение рассылки LSA через интерфейс
<code>ip ospf disable all</code>	(config-int)#	Отключение OSPF функционала
<code>ip ospf flood-reduction</code>	(config-int)#	Уменьшение сигнальной нагрузки путём установки DNA бита
<code>ip ospf mtu <576-65535></code>	(config-int)#	Установка MTU для OSPF пакетов
<code>ip ospf mtu-ignore</code>	(config-int)#	Отключение проверки MTU в DD сообщениях
<code>ip ospf priority <0-255></code>	(config-int)#	Установка OSPF приоритета
<code>ip ospf retransmit-interval <1-65535></code>	(config-int)#	Установка временного интервала для рассылки LSA соседям
<code>ip ospf transmit-delay <1-3600></code>	(config-int)#	Установка приблизительного времени передачи LSU через интерфейс
<code>ip ospf <N> area <K></code>	(config-int)#	<p>Включение процесса OSPF под L3 интерфейсом. Где N — номер процесса, а K — номер области.</p> <p>ВАЖНО!</p> <p>При отсутствии в конфигурации команды (router ospf ...), описываемая команда включит:</p> <ul style="list-style-type: none"> - процесс OSPF на всём устройстве, - приём/передачу OSPF сообщений на интерфейсе, - подсеть, сконфигурированную на интерфейсе, в анонс маршрутной информации. <p>Таким образом команды router ospf и network добавятся автоматически.</p> <p>При удалении команды из под интерфейса, процесс запущенный глобально на всём устройстве выключен не будет, произойдёт лишь автоматическое удаление команды network, со всеми вытекающими последствиями</p>

20.4.12 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.

20.4.13 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 создаётся новая таблица с резервными, надёжными маршрутами для быстрого переключения маршрутов (fast-reroute). Под надёжностью маршрута здесь понимается защищённость его от петель.

Если маршрутизатор детектирует падение локального линка, по которому строился основной маршрут, то в FIB моментально отправляется заранее выбранный резервный маршрут.

Пересчёт 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:

- Выигрывает маршрут с наименьшей метрикой.
- 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.

20.5 Протокол 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 на 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>`. Адрес является ISO NSAP (Network Service Access Point) адресом регламентируемым стандартом ISO 8348. Типичный вид NSAP-адреса в ISIS: 49.0001.0100.1001.0003.00, где 49 (AFI) означает частный характер сети, следующая группа цифр это номер зоны IS-IS, далее три группы по 4 цифры — это уникальный, в рамках сети, идентификатор (System-ID), последние две цифры — n-селектор (SEL), всегда должны быть равны нулю.

По умолчанию на маршрутизаторе можно задать три NET-адреса в различных областях, но идентификатор System-ID в каждом адресе всегда будет одинаков (система



принудительно преобразует идентификатор в заданный изначально). Установить допустимое количество задаваемых NET-адресов можно, командой: `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 установлен 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 (широковещательный или точка-точка) в режиме конфигурирования интерфейса с помощью команды `isis network {broadcast | point-to-point}`.

Этап 5.

Задайте значения таймеров в режиме конфигурирования интерфейсов с помощью команд `isis hello-interval <1-65535>` или с помощью задания множителя для расчёта hold-timer с помощью команды `isis hello-multiplier <2-100>`.

Этап 6.

Настройте вручную стоимости интерфейсов, если необходимо повлиять на выбор оптимального маршрута. Для этого в режиме конфигурирования интерфейсов укажите значение командой `isis metric <1-63>`.

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 задаётся через отдельный режим конфигурирования цепочек ключей с помощью команды `key chain <название цепочки>`, в этом режиме позволено указывать несколько ключей и паролей.

20.5.1 Пример настройки

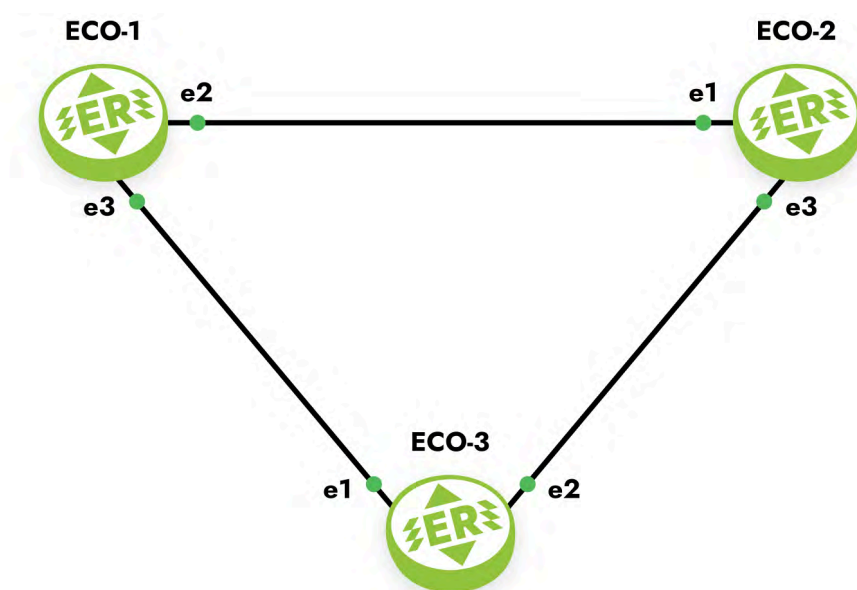


Рисунок 19

Шаг 1. Задание имени устройства.

```
hostname ECO-1
```

Шаг 2. Настройка портов, интерфейсов и сервисных интерфейсов.

```
interface e2
  ip address 10.12.0.1/16

interface e3
  ip address 10.13.0.1/16

port ge2
  service-instance ge2/e2
  encapsulation untagged
  connect ip interface e2

port ge3
  service-instance ge3/e3
  encapsulation untagged
  connect ip interface e3
```

Шаг 3. Включение маршрутизации.

```
router isis
  net 49.0001.0000.0000.0001.00
```



```
exit

interface e2
  ip router isis

interface e3
  ip router isis
exit
```

Шаг 4. Включение аутентификации между соседями.

```
key chain test
  key 1
    key-string ecorouter
  exit
exit

interface e2
  isis authentication mode md5
  isis authentication key-chain test

interface e3
  isis authentication mode md5
  isis authentication key-chain test
```

Конфигурация оставшихся маршрутизаторов будет аналогичной.

```
hostname ECO-2
key chain test2
  key 2
    key-string 0x8de456332b943f870ef377482f699e4c
  exit
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
```

20.5.2 Редистрибуция, фильтрация и суммирование 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> | <0-4261412864>] [metric-type <internal/external>] [level-1/level-2/level-1-2] [route-map <название>] .`

Для суммирования маршрутов используется команда: `summary-address <адрес/маска> [level-1/level-2/level-1-2] [metric <0-63> | <0-4261412864>] .`

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.

20.5.3 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 маршрут по умолчанию, 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 флудингом в NBMA соединениях EcoRouterOS позволяет добавлять интерфейсы в разные mesh-группы, тем самым накладывая определённые 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 передаётся дальше обычным путём. Если LSP был принят на интерфейс, который принадлежит mesh-группе, то он передаётся во все интерфейсы, кроме тех, которые принадлежат той же группе, или указаны с параметром `blocked`.

20.5.4 Дополнительные команды конфигурирования

Таблица 53 — Дополнительные команды конфигурирования протокола IS-IS

Команда	Режим	Описание
<code>ignore-lsp-errors</code>	(config-router)#	Игнорирование LSP с ошибками в контрольной сумме
<code>ispf</code>	(config-router)#	Включение инкрементального SPF
<code>lsp-gen-interval</code>	(config-router)#	Установка временного интервала регенерации LSP
<code>lsp-mtu</code>	(config-router)#	Размер MTU для LSP
<code>lsp-refresh-interval</code>	(config-router)#	Интервал обновления LSP

Команда	Режим	Описание
<code>max-lsp-lifetime</code>	(config-router)#	Время жизни LSP
<code>passive-interface</code>	(config-router)#	Задание пассивного интерфейса
<code>prc-interval-exp</code>	(config-router)#	Установка интервалов для PRC
<code>restart-timer</code>	(config-router)#	Установка сброса IS-IS таймера
<code>set-overload-bit</code>	(config-router)#	Установка overload бита
<code>spf-interval-exp</code>	(config-router)#	Установка интервалов для SPF
<code>isis csnp-interval</code>	(config-int)#	Установка CSNP интервала
<code>isis hello padding</code>	(config-int)#	Уменьшение размера Hello сообщений
<code>isis lsp-interval</code>	(config-int)#	Установка LSP интервала
<code>isis priority <0-127></code>	(config-int)#	Установка приоритета
<code>isis retransmit-interval</code>	(config-int)#	Установка временного интервала регенерации LSP
<code>clear isis process</code>	#	Сброс процесса маршрутизации

20.5.5 Команды просмотра

В таблице ниже приведены команды просмотра информации, относящиеся к протоколу. Как и другие команды **show**, они поддерживают использование модификаторов.

Таблица 54 — Команды просмотра протокола IS-IS

Команда	Описание
<code>show isis counter</code>	Выводит количественную информацию о IS-IS сообщениях
<code>show isis database</code>	Выводит краткую информацию о содержимом в базе данных
<code>show isis database detail</code>	Выводит полную информацию о содержимом в базе данных

Команда	Описание
<code>show isis interface</code>	Выводит информацию о параметрах сконфигурированных на интерфейсах, включённых в процесс маршрутизации
<code>show isis topology</code>	Выводит информацию о содержимом в базе данных топологии
<code>show clns neighbors</code>	Выводит информацию о соседских отношениях
<code>show clns protocol</code>	Выводит общую информацию о протоколе

20.6 Протокол 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.

Приведём несколько основных концепций протокола и сравним их с логикой работы Internal Gateway Protocol (IGP) маршрутизации, в качестве примера будет выступать OSPF.

Таблица 55 — сравнение OSPF логики с BGP

OSPF	BGP
Для отправки маршрутной информации должны сформироваться соседские отношения между маршрутизаторами	Используется подобная же логика
Соседи обнаруживаются при помощи мультикастовых сообщений в непосредственно подключённой подсети	Соседи указываются путём статической конфигурации и могут быть в разных подсетях
Не используют TCP	Используется TCP соединение между соседями (порт 179)
Рекламирует prefix/length	Рекламирует prefix/length (Network Layer Reachability Information)

OSPF	BGP
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

20.6.1 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 маршрутизируемых IP-адресов, процесс присвоения номеров регулируется ассоциацией IANA. При определённых 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 одной или различным автономным системам BGP определяет два класса соседств между маршрутизаторами: internal BGP (iBGP) и external BGP (eBGP) соответственно. Реализация протокола в оборудовании EcoRouter даёт 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 соседство 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 удалённой 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.

20.6.2 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 56 — Атрибуты BGP

Атрибут	Описание	Направление трафика
Weight	Числовое значение в диапазоне от 0 до 216-1, 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 232-1, 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 /



Атрибут	Описание	Направление трафика
		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 232-1, 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 предпочитаемых.

Таблица 57 — Приоритеты атрибутов BGP

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

Приоритет	Атрибут/свойство	Что лучше?
12	IP адрес соседа	Минимальное значение

Приведём 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`.

20.6.3 Команды конфигурирования атрибутов через route-map

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
vpn4            VPNv4 information
weight          BGP weight for routing table
```

Таблица 58 — Доступные для конфигурирования атрибуты

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	<p>Атрибут используется при агрегировании маршрутов. Команда для суммирования маршрутов:</p> <pre>aggregate-address <address> [summary-only] [as-set]</pre> <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 ?</pre> <p><1-65535> community number</p> <p>AA:NN community number in aa:nn format</p> <p>additive Add to the existing community</p> <p>internet Internet (well-known community)</p> <p>local-AS Do not send outside local AS (well-known community)</p> <p>no-advertise Do not advertise to any peer (well-known community)</p> <p>no-export Do not export to next AS (well-known community)</p> <p>none No community attribute</p> <p>Для дальнейшей рекламы маршрутов с атрибутом Community указывается команда: 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>, где numberAS — номер AS, которая прорекламировала маршрут</pre> <pre>route-map community permit 100</pre> <pre>match community 1</pre> <pre>set metric 777</pre>

Атрибут	Описание
	Для дальнейшей рекламы маршрутов с атрибутом Community указывается команда: <code>neighbor <address> send-community</code>
Dampening	Дополнительная функциональность протокола BGP для защиты от нестабильности соединений (route flapping). Команда <code>set dampening <1-45></code> , где 1-45 устанавливает значение Reachability Half-life time в минутах (время с момента успешного возобновления соединения до снятия штрафных очков (penalty))
Extcommunity / extcommunity-list	Атрибут для использования регулярных выражений
Local Preference	Атрибут указывает на выбор маршрутизатора, который будет использован для выхода из данной автономной системы. Команда <code>set local-preference <0-4294967295></code>
Metric	Атрибут Multiexit_Discriminator (MED) представляется аналогом метрики маршрута, устанавливается командой <code>set metric <1-4294967295></code> , по умолчанию MED равен нулю.
Origin	Атрибут указывает на то, каким образом был получен маршрут в обновлении. Значение меняется командой <code>set origin</code>
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 передаётся далее соседям. Значение устанавливается командой <code>set originator-id</code> . Возможные значения атрибута: <ul style="list-style-type: none"> - 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 <code>set vpnv4 next-hop <address></code> , 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 <code>set weight</code>

Для одновременной конфигурации большого количества соседств удобнее использовать группы соседей и применять политики ко всей группе. Конфигурация

потребуется несколько команд:

- `neighbor <name> peer-group` , где name — это имя группы;
- `neighbor <address> peer-group <name>` — привязка соседа к группе.

20.6.4 Пример настройки BGP

Рассмотрим пример настройки топологии:

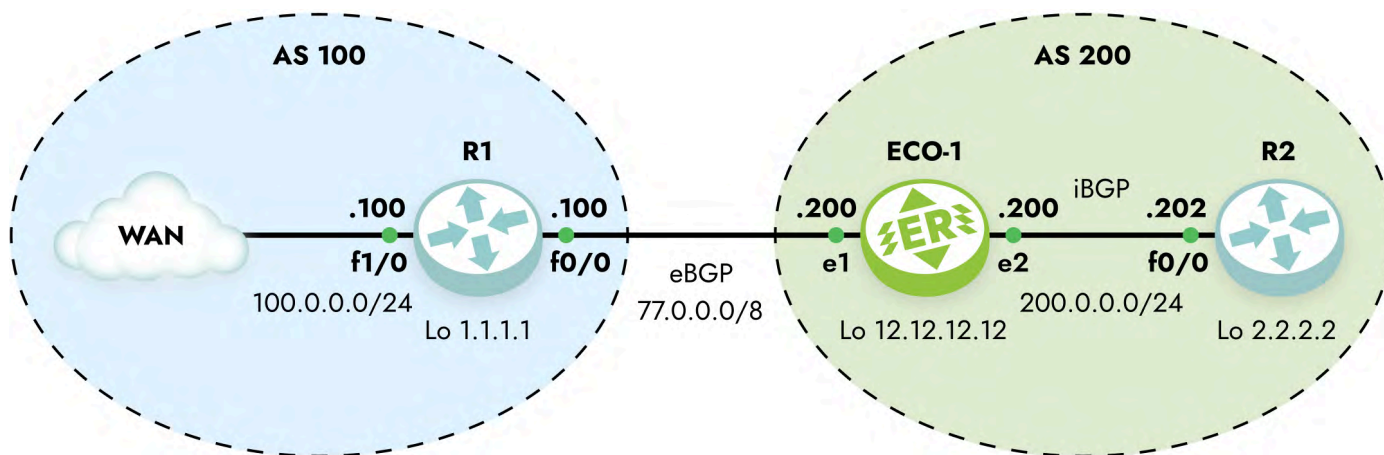


Рисунок 20

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
```

```
EC01(config-port)#service-instance ge2/e2
EC01(config-service-instance)#encapsulation untagged
EC01(config-service-instance)#connect ip interface e2
EC01(config-service-instance)#exit
EC01(config-port)#exit
```

Шаг 3. Настройка списков фильтрации

```
EC01(config)#policy-filter-list 1 permit 33.0.0.0 0.0.0.7
EC01(config)#policy-filter-list 2 permit 33.0.0.8 0.0.0.7
```

Шаг 4. Привязка списков фильтрации и назначение метрики для сетей

```
EC01(config)#route-map bgp permit 1
EC01(config-route-map)#match ip address 1
EC01(config-route-map)#set metric 1000
EC01(config-route-map)#route-map bgp permit 2
EC01(config-route-map)#match ip address 2
EC01(config-route-map)#set metric 500
```

Шаг 5. Создание пустого списка фильтрации для всех остальных маршрутов с метрикой по умолчанию

```
EC01(config-route-map)#route-map bgp permit 3
EC01(config-route-map)#exit
```

Шаг 6. Создание и описание групп соседей

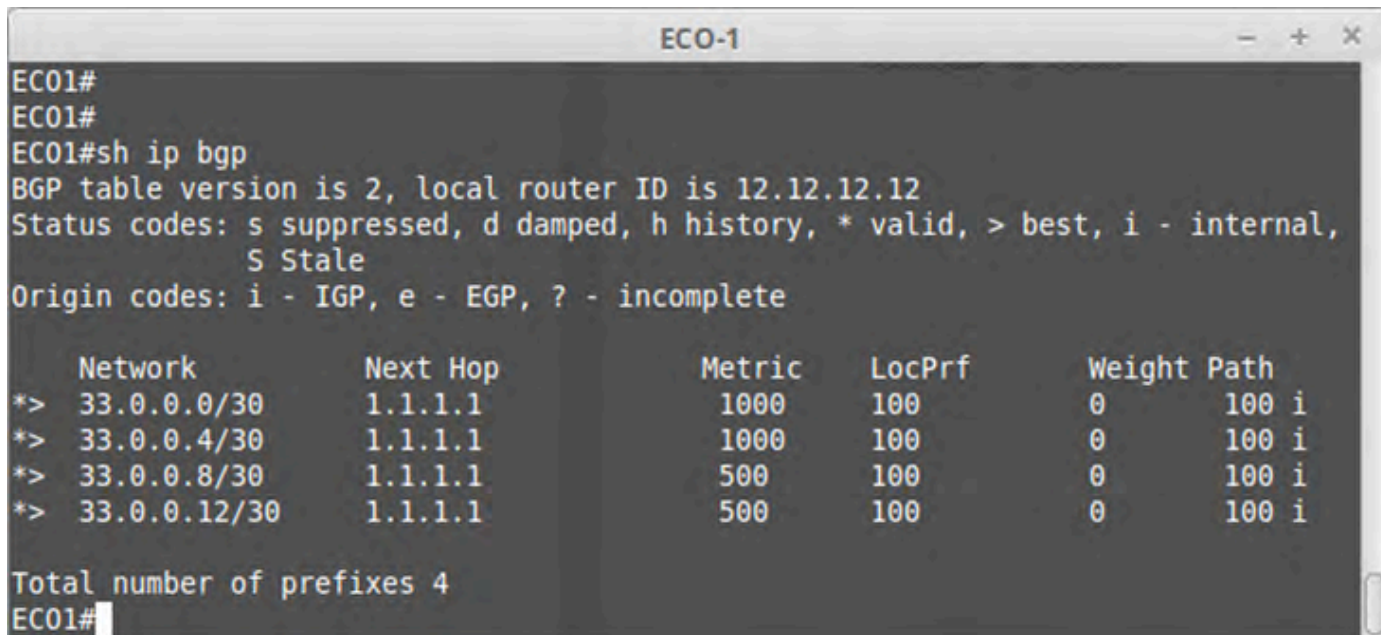
```
EC01(config)#router bgp 200
EC01(config-router)#neighbor eBGP peer-group
EC01(config-router)#neighbor eBGP remote-as 100
EC01(config-router)#neighbor eBGP ebgp-multihop 2
EC01(config-router)#neighbor eBGP update-source loopback.0
EC01(config-router)#neighbor eBGP route-map bgp in
EC01(config-router)#neighbor iBGP peer-group
EC01(config-router)#neighbor iBGP remote-as 200
EC01(config-router)#neighbor iBGP update-source loopback.0
EC01(config-router)#neighbor iBGP next-hop-self
EC01(config-router)#neighbor 1.1.1.1 peer-group eBGP
EC01(config-router)# neighbor 2.2.2.2 peer-group iBGP
EC01(config-router)#exit
```

Шаг 7. Создание статических маршрутов

```
EC01(config)#ip route 1.1.1.1/32 77.0.0.100
EC01(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
*> 33.0.0.0/30    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 21

Для помещения маршрутов в BGP и дальнейшего анонсирования следует воспользоваться командой `network` либо сделать редистрибуцию из Interior Gateway Protocols (далее IGP) командой `redistribute`.

Таблица 59 — Параметры редистрибуции маршрутов

connected	Включить в редистрибуцию маршруты к присоединённым 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.

20.6.5 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 .

Таблица 60 — Команды для фильтрации маршрутов в BGP

Команда	Список, на который ссылается команда
<code>neighbor <A.B.C.D> allowas-in <1-10></code>	
<code>neighbor <A.B.C.D> soft-reconfiguration inbound</code>	
<code>neighbor <A.B.C.D> capability dynamic</code>	
<code>neighbor <A.B.C.D> capability orf prefix-list {both receive send}</code>	
<code>neighbor <A.B.C.D> capability route-refresh</code>	
<code>neighbor <A.B.C.D> distribute-list {<1-199> <1300-2699> WORD} {in out}</code>	policy-filter-list
<code>neighbor <A.B.C.D> prefix-list <WORD> {in out}</code>	ip prefix-list
<code>neighbor <A.B.C.D> filter-list <WORD> {in out}</code>	ip as-path access-list
<code>neighbor <A.B.C.D> route-map <WORD> {in out}</code>	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>**.

20.6.6 BGP Additional Paths

BGP Additional Paths — это расширение возможностей протокола BGP (Border Gateway Protocol), позволяющее маршрутизаторам передавать и принимать не только лучший путь к подсети, но и дополнительные альтернативные пути. Это улучшает отказоустойчивость, балансировку нагрузки и оптимизацию передачи данных в сложных сетях.

В норме маршрутизаторы распространяют только единственный лучший путь для каждой подсети. Дополнительные пути оказываются скрыты, и это не всегда является

оптимальным поведением. Соккрытие путей может помешать эффективному использованию сети.

При плановом обслуживании и переключении на резервный путь BGP должен заново обработать все маршруты, что может вызвать задержки в работе сети.

Аналогичная ситуация может случиться при сбое на основном пути, BGP должен будет дожидаться обновлений от соседей, что занимает секунды. Но с Additional Paths резервные пути известны, и переключение происходит почти мгновенно.

Для настройки BGP Additional Paths достаточно дать команду в контекстном режиме настройки BGP роутера (`router bgp <AS_NUM>`): `neighbor <1.1.1.1> capability add-path {both|receive|send}`, где **AS_NUM** — номер автономной системы, **1.1.1.1** — IP-адрес соседнего маршрутизатора, **both | receive | send** — то что мы "предлагаем" сделать соседу: принимать, отправлять дополнительные пути или и то и другое вместе.

Также можно выбрать какие пути принимать во внимание все или только лучшие. Делается это командой `neighbor <1.1.1.1> advertise add-path {all|best}`, где **all** — все доступные маршруты, а **best** — маршруты не хуже лучшего по совокупности параметров. По умолчанию считается применённой команда с параметром **all**.

20.6.7 Обновление партнёрских BGP отношений

Таблица 61 — Команды для обновления партнёрских BGP отношений

Команда	Тип обновления	Количество соседей, направление
<code>clear ip bgp</code>	Жёсткий	Все, входящие/исходящие
<code>clear ip bgp neighbor-id</code>	Жёсткий	Один, входящие/исходящие
<code>clear ip bgp neighbor-id in/out</code>	Мягкий	Один, входящие/исходящие
<code>clear ip bgp neighbor-id soft in/out</code>	Мягкий	Один, входящие/исходящие
<code>clear ip bgp soft</code>	Мягкий	Все, входящие/исходящие
<code>clear ip bgp neighbor-id soft</code>	Мягкий	Один, входящие/исходящие

Под "жёстким" 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-соседами вызывают всплеск нагрузки на центральный процессор маршрутизатора. Чтобы уменьшить этот эффект и сделать работу с BGP-соседами 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 1.1.1.1 advertisement-interval <VALUE>`, где **VALUE** указывается в секундах. Выключить подобное поведение можно командой `neighbor 10.0.0.2 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 команд сброса `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 добавив 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-refre**sh**** 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
. . . . .
```

Фраза в выводе **«advertised and received»** говорит о включённой 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
. . . . .
```

20.6.8 Регулярные выражения

В реализации EcoRouterOS представлен следующий набор регулярных выражений (см. таблицу ниже):

Таблица 62 — Регулярные выражения в EcoRouterOS

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 подключённой (соседней) AS.

20.6.9 Рефлекторы маршрутов и конфедерации

Рефлектором называется маршрутизатор, который выполняет функцию отражения маршрутов. Рефлектор получает маршрут от одного соседа и рассылает 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 рефлятора 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.

20.6.10 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 63 — Команды конфигурирования BGP

Команда	Режим	Описание
<code>router bgp <номер AS></code>	конфигурационный	Переход в режим конфигурирования протокола BGP
<code>address-family ipv4 {unicast multicast}</code>	контекстный	Переход в режим конфигурирования address-family
<code>aggregate-address <адрес></code>	контекстный	Создание суммарного маршрута
<code>auto-summary</code>	контекстный	Включение автосуммаризации
<code>bgp always-compare-med</code>	контекстный	Сравнение атрибутов MED для маршрута, полученного из разных AS определяет лучший путь

Команда	Режим	Описание
<code>bgp as-local-count <2-64></code>	контекстный	Определяет количество значений собственной AS в атрибуте AS-path
<code>bgp bestpath ...</code>	контекстный	Изменение алгоритма выбора лучшего пути
<code>bgp client-to-client reflection</code>	контекстный	Включение роли рефлектора
<code>bgp cluster-id <1-4294967295></code>	контекстный	Указание номера кластера
<code>bgp confederation identifier <1-65535></code>	контекстный	Указание номера конфедерации
<code>bgp confederation peers <1-65535></code>	контекстный	Указание соседей в конфедерации
<code>bgp config-type {standard ecorouteros}</code>	конфигурационный	Определение типа конфигурации, по умолчанию включена ecorouteros , для передачи атрибута community используется тип standard
<code>bgp dampening ...</code>	контекстный	Настройка подавления нестабильных маршрутов
<code>bgp default local-preference <0-4294967295></code>	контекстный	Указание значения атрибута local preference
<code>bgp deterministic-med</code>	контекстный	Сравнение атрибутов MED для маршрута, полученного из одной AS; атрибуты weight , local preference , AS-path и origin должны быть равны
<code>bgp enforce-first-as</code>	контекстный	Update сообщение, которое пришло не от соседней сконфигурированной AS, будет отброшено
<code>bgp fast-external-failover</code>	контекстный	Моментальный сброс BGP сессии при падении интерфейса
<code>bgp nexthop-trigger delay <1-100></code>	конфигурационный	Установка задержки на изменения параметров в BGP таблице при каких-либо изменениях параметров у соседнего маршрутизатора

Команда	Режим	Описание
<code>bgp nexthop-trigger enable</code>	конфигурационный	Включение специального мониторинга адреса соседа
<code>bgp rfc1771-path-select</code>	конфигурационный	Включение алгоритма выбора лучшего пути согласно RFC 1771
<code>bgp rfc1771-strict</code>	конфигурационный	Установка атрибута origin согласно RFC 1771
<code>bgp router-id <адрес></code>	контекстный	Указание BGP идентификатора маршрутизатора
<code>bgp scan-time <0-60></code>	контекстный	Значение интервала сканирования доступности маршрутов в таблице маршрутизации (по умолчанию 60 сек)
<code>distance bgp <1-255> <1-255> <1-255></code>	контекстный	Установка административных расстояний для external, internal, local маршрутов
<code>max-paths {ebgp ibgp} <2-64></code>	контекстный	Максимальное количество возможных маршрутов, которые считаются равными
<code>mpls-resolution</code>	контекстный	Автоматическое создание FTN записи для префиксов, полученных от BGP соседей
<code>neighbor <адрес> activate</code>	контекстный	Активация соседских отношений при конфигурировании address-family
<code>neighbor <адрес> advertisement-interval <0-65535></code>	контекстный	Указание минимального интервала для отправки Update сообщений
<code>neighbor <адрес> allowas-in <1-10></code>	контекстный	Указание на рекламу префиксов (маршрутов) даже если их источник в той же AS
<code>neighbor <адрес> as-origination-interval <1-65535></code>	контекстный	Указание минимального интервала для отправки AS-origination Update сообщений
<code>neighbor <адрес> attribute-unchanged [as-path next-hop med]</code>	контекстный	При изменении значений атрибутов отправлять значения по умолчанию

Команда	Режим	Описание
<code>neighbor <адрес> capability dynamic</code>	контекстный	Включение динамических возможностей для определённого узла
<code>neighbor <адрес> capability orf prefix-list</code>	контекстный	Включение фильтрации ORF и анонсирования ORF соседям
<code>neighbor <адрес> capability route-refresh</code>	контекстный	Включение анонсирования поддержки возможностей обновления маршрутов
<code>neighbor <адрес> connection-retry-time <1-65535></code>	контекстный	Установка таймера повторного соединения с соседом (по умолчанию 120 сек.)
<code>neighbor <адрес> default-originate</code>	контекстный	Отправка соседу маршрута по умолчанию
<code>neighbor <адрес> description</code>	контекстный	Описание для соседнего маршрутизатора (максимум 80 символов)
<code>neighbor <адрес> disable-infinite-holdtime</code>	контекстный	Невозможность задать бесконечный hold-time
<code>neighbor <адрес> disable-capability-negotiate</code>	контекстный	Отключение проверки на совместимость версий протокола (отключена по умолчанию)
<code>neighbor <адрес> ebgp-multihop <1-255></code>	контекстный	Установка значения TTL в BGP пакетах при eBGP сессии
<code>neighbor <адрес> enforce-multihop</code>	контекстный	Включение принудительного непрямого соседства
<code>neighbor <адрес> local-as <1-4294967295></code>	контекстный	Указание номера локальной AS
<code>neighbor <адрес> maximum-prefix <1-4294967295></code>	контекстный	Контроль количества принимаемых маршрутов от соседа
<code>neighbor <адрес> next-hop-self</code>	контекстный	Отправка информации о Next-Hop соседям iBGP
<code>neighbor <адрес> passive</code>	контекстный	Включение пассивного режима
<code>neighbor <адрес> password</code>	контекстный	Задание MD5 пароля для аутентификации (максимум 80 символов)

Команда	Режим	Описание
<code>neighbor {имя адрес} peer-group <имя></code>	контекстный	Создание группы соседей/добавление в группу
<code>neighbor <адрес> port <0-65535></code>	контекстный	Указание BGP порта для соседа
<code>neighbor <адрес> remote-as</code>	контекстный	Задание номера AS соседа
<code>neighbor <адрес> remove-private-AS</code>	контекстный	Удаление AS из приватного диапазона при отправке обновлений
<code>neighbor <адрес> route-reflector-client</code>	контекстный	Включение роли рефлেকтора и указание соседа в роли клиента
<code>neighbor <адрес> route-server-client</code>	контекстный	Указание соседа в роли сервер-клиента
<code>neighbor <адрес> send-community <both/extended/standard></code>	контекстный	Отправка атрибута community
<code>neighbor <адрес> shutdown</code>	контекстный	Административное выключение BGP отношений
<code>neighbor <адрес> soft-reconfiguration inbound</code>	контекстный	Включение локальной базы данных для принятых маршрутов
<code>neighbor <адрес> timers <0-65535> <0-65535> [connect <1-65535>]</code>	контекстный	Задание keepalive, hold и connect таймеров
<code>neighbor <адрес> transparent-as</code>	контекстный	Включение режима прозрачной AS, не включать значение собственной AS в атрибут AS-path
<code>neighbor <адрес> transparent-nexthop</code>	контекстный	Включение режима прозрачной AS, не указывать себя в качестве Next-Hop к маршруту
<code>neighbor <адрес> unsuppress-map <имя группы></code>	контекстный	Реклама более специфических маршрутов для соседа при созданном суммарном маршруте
<code>neighbor <адрес> update-source <адрес></code>	контекстный	Указание интерфейса для создания TCP-сессии
<code>neighbor <адрес> weight <0-65535></code>	контекстный	Задание атрибута weight
<code>network <адрес></code>	контекстный	Указание подсетей для рекламы

Команда	Режим	Описание
<code>redistribute {connected isis rip static}</code>	контекстный	Редистрибуция в BGP
<code>synchronization</code>	контекстный	Включение режима синхронизации
<code>timers bgp <0-65535> <0-65535></code>	контекстный	Задание keepalive и hold таймеров

20.6.11 BGP Команды просмотра

Команды просмотра информации о настройках и статистике протокола BGP представлены в таблице ниже.

Таблица 64 — Команды просмотра BGP

Команда	Описание
<code>show bgp statistics</code>	Вывод статистических данных
<code>show ip bgp</code>	Выводит BGP таблицу
<code>show ip bgp <адрес сети></code>	Выводит информацию о конкретном маршруте
<code>show ip bgp attribute-info</code>	Отображает информацию обо всех внутренних атрибутах
<code>show ip bgp community</code>	Выводит список маршрутов, принадлежащих к конкретному сообществу
<code>show ip bgp community-info</code>	Выводит информацию о сообществах
<code>show ip bgp dampening {dampened-paths flap-statistics parameters} vrf {<vrf-name> all default}</code>	Выводит информацию о подавлении нестабильных маршрутов
<code>show ip bgp filter-list</code>	Выводит список маршрутов, соответствующий AS-path списку
<code>show ip bgp ipv4 <unicast/multicast> ...</code>	Вывод информации для address-family
<code>show ip bgp neighbors</code>	Выводит информацию обо всех сконфигурированных соседях
<code>show ip bgp neighbors <address>advertised-routes</code>	Выводит информацию о рекламируемых маршрутах, которые прошли исходящую фильтрацию

Команда	Описание
<code>show ip bgp neighbors <address> routes</code>	Выводит информацию о получаемых маршрутах, которые прошли входящую фильтрацию
<code>show ip bgp neighbors <address>received-routes*</code>	Выводит информацию о получаемых маршрутах до каких-либо входных фильтров
<code>show ip bgp paths</code>	Отображает информацию о маршрутах для локального маршрутизатора
<code>show ip bgp prefix-list</code>	Выводит список маршрутов, соответствующий списку префиксов
<code>show ip bgp regexp</code>	Выводит список маршрутов, соответствующий регулярному выражению
<code>show ip bgp route-map</code>	Выводит список маршрутов, соответствующий карте маршрутов
<code>show ip bgp summary</code>	Отображает состояние всех соединений BGP

20.6.12 Dampening

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 превышает 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-REACHIBILITY-HALF-LIFE-TIME>}`. Команда также позволяет в явном виде указать карту маршрутов для подавления.

Таблица 65 — Параметры команды `bgp dampening`

Параметр	Описание
<code><ROUTE-MAP-NAME></code>	Имя карты маршрутов для подавления
<code><REACHIBILITY-HALF-LIFE-TIME></code>	Время доступности в минутах, за которое штраф уменьшается вдвое (default reachability half-life time). Допустимый диапазон 1-45. Значение по умолчанию — 15.
<code><REUSE-VALUE></code>	Значение предела для повторного использования маршрута. Когда значение штрафа опускается ниже этого значения, маршрут перестает подавляться. Допустимый диапазон 1–20000. Значение по умолчанию 750.
<code><SUPPRESS-VALUE></code>	Значение предела для подавления маршрута. Когда значение штрафа превышает это значение, маршрут подавляется. Допустимый диапазон 1–20000. Значение по умолчанию 2000.
<code><MAX-SUPPRESS-VALUE></code>	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.
<code><UN-REACHIBILITY-HALF-LIFE-TIME></code>	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
```

20.6.13 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.

20.6.14 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 66 — Параметры команды `clear ip bgp dampening`

Параметр	Описание
<code><ADDRESS>/<MASK></code>	Очистить информацию для подсети с указанным IP и маской, например, 35.0.0.0/8
<code>vrf {<VRF-NAME> default all}</code>	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 67 — Параметры команды `clear bgp ipv4`

Параметр	Описание
<code><ADDRESS>/<MASK></code>	IP-адрес подсети и маска, например, 35.0.0.0/8
<code>multicast unicast</code>	Выбрать режим multicast или unicast
<code><AS-number></code>	Номер автономной системы в диапазоне от 1 до 4294967295
<code>flap-statistics</code>	Очистить статистику по переключающимся маршрутам выбранной по адресу и маске (ADDRESS/MASK) или имени (VRF-NAME) сущности, всех сущностей (all) или сущности по умолчанию (default)

Примеры:

```
#clear bgp statistics
#clear bgp ipv4 unicast flap-statistics all
```

20.6.15 BGP Black-Hole

В качестве одного из методов защиты от DDoS атак в EcoRouterOS предусмотрен функционал отбрасывания трафика через Null-интерфейс путём 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.

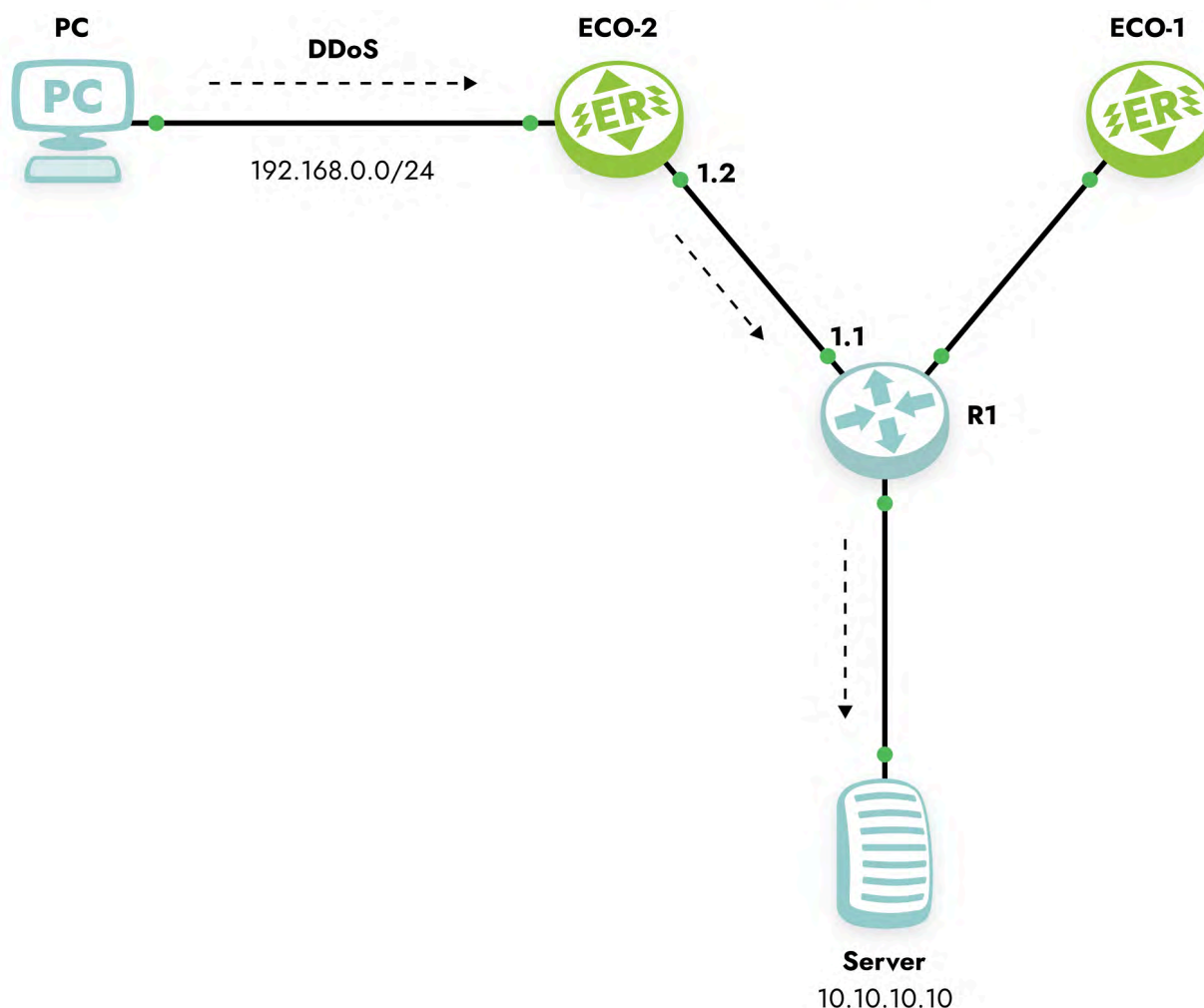


Рисунок 22

Допустим, что злоумышленник PC из сети 192.168.0.0/24 подаёт 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 определённым 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 te1
 lacp-priority 32767
 mtu 9728
 service-instance 1
  encapsulation untagged
!
interface 1
 ip mtu 1500
 connect port te1 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-multihop <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).

20.7 Функция защиты BGP сессий

Функция представляет собой не требовательный к ресурсам маршрутизатора механизм безопасности, предназначенный для защиты внешних сессий BGP (eBGP) от атак, основанных на использовании поддельных IP-пакетов с целью загрузки процессора маршрутизатора. Включение этой функции предотвращает попытки захвата сессии eBGP со стороны хоста в сетевом сегменте, который не является частью ни одной из BGP-сетей, или со стороны хоста в сетевом сегменте, который не находится между eBGP-пирами.

Эта функция активируется путём настройки значения Time To Live (TTL) для входящих IP-пакетов, полученных от определённого eBGP-пира. Когда функция включена,



BGP устанавливает и поддерживает сессию только в том случае, если значение TTL в заголовке IP-пакета равно или превышает значение TTL, которое вычисляется по формуле: **TTL = 255 - $H(SecurityValue)$** .

Если значение TTL IP-пакета меньше вычисленного, пакет отбрасывается, и сообщение протокола ICMP (Internet Control Message Protocol) не генерируется. Эта функция эффективна и проста в развёртывании.

20.7.1 Предварительные требования для поддержки BGP проверки безопасности TTL

- BGP должен быть настроен в вашей сети, и сессии eBGP-пиринга должны быть установлены.
- Функция должна быть настроена на обоих маршрутизаторах eBGP.

20.7.2 Ограничения функции BGP TTL-security

- Функция предназначена для защиты только сессий eBGP-пиринга и не поддерживается для внутренних BGP-пиров (iBGP) и групп iBGP-пиров.
- При настройке функции BGP TTL-security для существующей многоскачковой сессии eBGP пиринга необходимо сначала отключить команду конфигурации маршрутизатора `neighbor ebgp-multihop`, введя команду `no neighbor <ip address> ebgp-multihop`, перед настройкой этой функции с помощью команды `neighbor <ip address> ttl-security`. Эти команды являются взаимоисключающими, и для установления multihop сессии пиринга требуется только одна из них.
- Функцию следует настроить на обоих участниках eBGP сессии. Чтобы максимизировать эффективность этой функции, аргумент hop-count должен быть строго настроен в соответствии с количеством прыжков между локальной и внешней сетью. Однако при настройке этой функции для многоскачковой сессии пиринга также следует учитывать возможные изменения пути.
- Эффективность этой функции снижается в многоскачковых сессиях с большим диаметром. В случае атаки, направленной на загрузку процессора маршрутизатора BGP, настроенного для многоскачкового пиринга с большим диаметром, вам, возможно, все равно придется отключать затронутые сессии пиринга для устранения атаки.

- Функция неэффективна против атак со стороны пира, который был скомпрометирован внутри вашей сети. Это ограничение также включает BGP-пиры, которые не являются частью локальной или внешней BGP-сети, но подключены к сетевому сегменту между BGP-пирами (например, коммутатор или хаб, используемый для соединения локальной и внешней BGP-сетей).
- Функция не защищает целостность данных, передаваемых между eBGP-пирами, и не проверяет подлинность eBGP-пиров с использованием каких-либо методов аутентификации. Эта функция только проверяет локально настроенное значение TTL в сравнении с полем TTL в заголовке IP-пакета.

20.7.3 Пример настройки функции BGP TTL-security

Конфигурация EcoRouter для включения функции BGP TTL-security:

```
configure terminal
router bgp 65200
  bgp router-id 20.20.20.2
  network 20.1.2.0/27
  network 20.20.20.2/32
  neighbor 10.2.4.2 remote-as 65002
  neighbor 10.2.4.2 ttl-security hops 1
  neighbor 10.2.4.2 update-source 10.2.4.4
```

В данном случае конфигурация подразумевает значение TTL установленное в 255 для исходящих пакетов и 254 для входящих пакетов.

Проверка установленной BGP сессии на EcoRouter:

```
br2#show ip bgp neighbors 10.2.4.2
BGP neighbor is 10.2.4.2, remote AS 65002, local AS 65200, external link
  BGP version 4, remote router ID 10.10.10.2
  BGP state = Established, up for 00:37:45
  Last read 00:37:45, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (new)
    4-Octet ASN Capability: advertised and received
    Address family IPv4 Unicast: advertised and received
  Received 755 messages, 0 notifications, 0 in queue
  Sent 2639 messages, 607 notifications, 0 in queue
```



```
Route refresh request: received 0, sent 0
Minimum time between advertisement runs is 30 seconds
Maximum number of update messages in a burst 50
Auto-refresh enabled
Update source is 10.2.4.4
For address family: IPv4 Unicast
BGP table version 9, neighbor version 9
Index 1, Offset 0, Mask 0x2
Community attribute sent to this neighbor (both)
7 accepted prefixes
6 announced prefixes

Connections established 4; dropped 4
      External BGP neighbor must be up to 1 hops away.
# Результат работы BGP TTL-security
Local host: 10.2.4.4, Local port: 39713
Foreign host: 10.2.4.2, Foreign port: 179
Nexthop: 10.2.4.4
Last Reset: 00:37:51, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)

br2#
```

20.8 Настройка uRPF

Технология Unicast Reverse Path Forwarding (uRPF) обеспечивает надёжную и безопасную маршрутизацию в различных сегментах сетевой инфраструктуры на основе проверки IP-адресов отправителей (anti-spoofing) на третьем уровне модели OSI.

EcoRouterOS реализует строгий режим uRPF (Strict Mode), проверяет IP адрес отправителя, и если интерфейс на который пришёл пакет не соответствует записи в FIB (например в действительности такой отправитель находится за другим интерфейсом), то пакет будет отброшен. В противном случае, он будет перенаправлен в соответствии с записью в FIB.

Функция uRPF может быть включена и выключена для каждого интерфейса в отдельности:

```
ecorouter(config)#interface te0
ecorouter(config-if)#ip urpf ?
  disable  Disable strict URPF
  enable   Enable strict URPF
```



```
ecorouter(config-if)#ip urpf enable  
ecorouter(config-if)#ip urpf disable
```

Также есть возможность включить функцию uRPF для всех имеющихся интерфейсов с помощью макрокоманды:

```
ecorouter(config)#ip urpf ?  
    enable    Enable strict URPF  
  
ecorouter(config)#ip urpf enable ?  
    all-interfaces  Enable urpf on all interfaces  
  
ecorouter(config)#ip urpf enable all-interfaces
```

ВНИМАНИЕ! Для вновь создаваемых интерфейсов функция uRPF будет отключена. Для её включения потребуется повторно ввести команду включения для конкретного интерфейса или для всех имеющихся интерфейсов.

21 Списки доступа

Списки доступа представляют собой набор текстовых выражений-инструкций, которые позволяют "заглянуть" внутрь кадра/пакета, сопоставить текстовое правило списка с данными в этом сообщении и на основании этого принять решение, что делать с этим кадром/пакетом далее. В EcoRouterOS применяются следующие списки доступа (краткая характеристика ниже, более подробно о работе с каждым в соответствующих подразделах данного раздела):

- 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-map 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.

21.1 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 передаётся 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>`.

Таблица 68 — Параметры команды `policy-filter-list`

Параметр	Описание
<code>PFL_NAME</code>	Номер списка фильтрации. Нумерация списков осуществляется из диапазона от 1 до 99 и от 1300 до 1999
<code>permit</code> <code>deny</code>	Тип правила: разрешить (permit) или запретить (deny)
<code>ADDRESS</code>	IP-адрес сети, задаётся в виде A.B.C.D . Если под правило должны попадать все адреса, значение параметра должно быть any
<code>WILDCARD</code>	Инверсная маска, задаётся в виде A.B.C.D

После создания списка фильтров, он должен быть применён к определённому процессу маршрутизации на устройстве.

Команды добавления фильтров различаются в зависимости от протокола.

Таблица 69 — Команда добавления списка фильтров

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

21.1.1 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 применён к определённому процессу маршрутизации на устройстве.

Команды добавления фильтров различаются в зависимости от протокола.

21.1.2 Настройка фильтрации маршрутной информации в BGP

Настройка списков фильтрации делается аналогично OSPF.

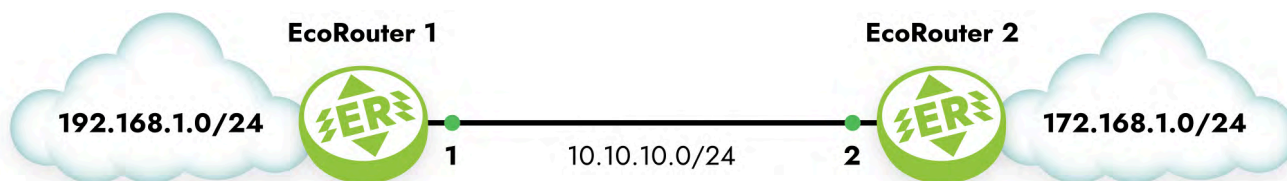


Рисунок 23

Применение списка фильтрации отличается.

Для фильтрации маршрутных обновлений BGP список фильтров применяется к определённому 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
```

21.1.3 Настройка фильтрации маршрутной информации в IS-IS

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

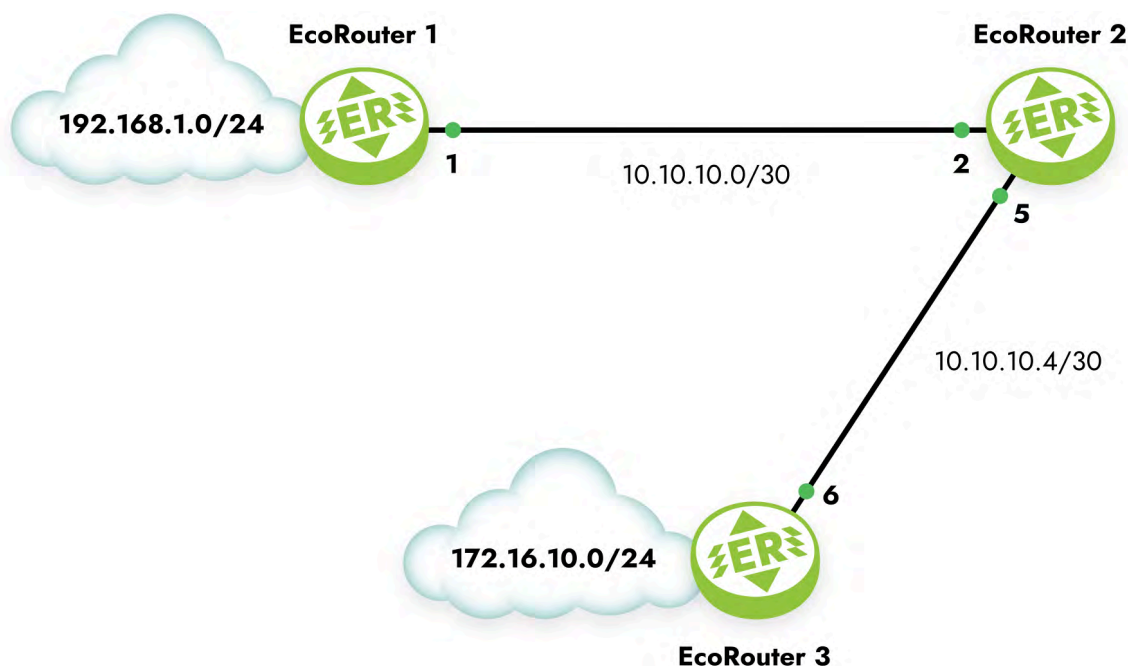


Рисунок 24

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 te1  
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

21.1.4 Настройка фильтрации маршрутной информации в OSPF

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

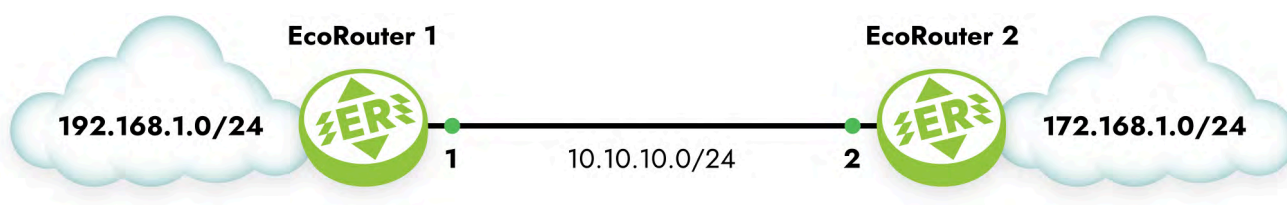


Рисунок 25

> [!table] *Таблица* — Текущая конфигурация на маршрутизаторах

EcoRouter 1	EcoRouter 2
<pre>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</pre>	<pre>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</pre>

Таблица 70 — Вывод таблицы маршрутизации на EcoRouter_1 и EcoRouter_2

EcoRouter 1	EcoRouter 2
<pre>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</pre>	<pre>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</pre>

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 список 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 0x1fcd
EcoRouter_1#
```

Информация о этой сети присутствует в базе состояния каналов OSPF.

21.2 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.

21.2.1 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 учётом 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. Приведём пример для 6 указанных подсетей:

- 10.0.0.0/8
- 10.128.0.0/9
- 10.1.1.0/24
- 10.1.2.0/24
- 10.128.10.4/30

- 10.128.10.8/30

Таблица 71 — Соответствие префиксных списков

Команда	Номера подсетей, соответствующие условию
ip prefix-list permit 10.0.0.0/8	1
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 версии 3.2. OS при конфигурации префиксных списков в BRAS параметры **ge**, **le**, **eq** не учитываются.

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

21.2.2 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>` выдаёт статистику по совпадениям в списке префиксов (hit count) и по совпадению в приложениях (route-map), где используется список префиксов (refcount).

Таблица 72 — Команды просмотра списков префиксов

Команда	Описание
---------	----------



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

21.3 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**.

При большом количестве фильтрующих правил, для оптимизации построения списка контроля доступа (ACL) и снятия нагрузки с нулевого (зарезервированного) ядра используйте команду **filter-map optimize**. Наибольшую эффективность команда приобретает в сочетании с командой **hw reserved-cores {0 | 10}** которая позволяет зарезервировать больше ядер процессора для плоскости управления (control-plane).

filter-map locally-generated — включает применение карты фильтров к локальному (производимому самим маршрутизатором) трафику.

21.3.1 Настройка L2 filter-map

Ещё 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 создаётся 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**
<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>` .

Таблица 73 — Параметры filter-map ethernet

Параметр	Описание
<code>FILTER_MAP_ETHERNET_LIST</code>	Имя списка фильтрации, может принимать любое значение
<code>SEQUENCE_NUMBER</code>	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
<code>SOURCE_MAC</code>	MAC-адрес источника, задаётся в одном из трёх 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> .
<code>SRC_WILDCARD</code>	Инверсная маска источника, задаётся в одном из трёх форматов: - XX-XX-XX-XX-XX-XX , - XX:XX:XX:XX:XX:XX , - XXXX.XXXX.XXXX .
<code>DESTINATION_MAC</code>	MAC-адрес назначения, задаётся в одном из трёх 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> .
<code>DST_WILDCARD</code>	Инверсная маска назначения, задаётся в одном из трёх форматов: - XX-XX-XX-XX-XX-XX , - XX:XX:XX:XX:XX:XX , - XXXX.XXXX.XXXX .
<code>ETHERTYPE</code>	Значение поля ethernet. Может быть указано шестнадцатеричное значение поля в диапазоне (0x600

Параметр	Описание
	<ul style="list-style-type: none"> — 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)
set port <NAME>	Пакеты, попавшие под действие правила, перенаправляются в указанный порт. NAME — имя порта (обозначения портов подробнее описаны в разделе Сервисные интерфейсы)
set port <NAME> push <TAG>	Пакеты, попавшие под действие правила, перенаправляются в указанный порт с добавлением VLAN-тега. Где NAME — имя порта, TAG — номер VLAN
set port <NAME> pop <NUMBER>	Пакеты, попавшие под действие правила, перенаправляются в указанный порт со снятием VLAN-тегов. Где NAME — имя порта, NUMBER — количество тегов, которое необходимо снять

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>]` .

21.3.1.1 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.ff00 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
```

21.3.2 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 применён к нескольким интерфейсам одновременно.

Использование filter-map производится в два этапа.



- Создание и наполнение правилами.
- 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):

- 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.
- Вторая строка. Указать правило, на соответствие которому будут проверяться пакеты, следующего вида: `match [fragments] <PROTOCOL> <SRC_ADDRESS> [<PORT_CONDITION>] <DST_ADDRESS> [<PORT_CONDITION>] [dscp <DSCPVALUE>] [<TTL>] [<FLAG>]`. Подробнее параметры описаны в таблицах ниже.
- 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**.

Таблица 74 — Общие параметры `filter-map ipv4`

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
fragments	Указывает на то, что отфильтровать необходимо только вторичные фрагменты пакетов. Нефрагментированные пакеты и первичные фрагменты отфильтрованы не будут. Использование данного ключевого слова исключает возможность использования параметра PORT_CONDITION .

Параметр	Описание
PROTOCOL	<p>Значение поля protocol. Может быть указано значение поля в диапазоне (0–255) или одно из следующих обозначений:</p> <ul style="list-style-type: none"> - ipinip; - icmp; - gre; - igmp; - pim; - 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	<p>IP-адрес источника, задается в одном из следующих форматов:</p> <ul style="list-style-type: none"> - 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	<p>IP-адрес назначения, задается в одном из следующих форматов:</p> <ul style="list-style-type: none"> - 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	<p>Значение DSCP (Differentiated Services Code Point) для проверки пакета, целое число от 0 до 63</p>
TTL	<p>Значение TTL (time to live) пакета, задаётся в виде точного значения — eq, всех значений более указанного — gt, всех значений менее указанного — lt, диапазона — range. Полный синтаксис параметра TTL:</p> <pre>ttl { { eq gt lt } <0-255> range <0-255> <0-255> }</pre>
set <ACTION>	
set accept	Разрешить
set discard	Запретить без отправки ICMP-уведомления
set reject	Запретить с отправкой ICMP-уведомления

Параметр	Описание
<code>set nexthop <A.B.C.D></code>	Указать IP-адрес next hop. Пакеты, попавшие под действие правила, отсылаются на адрес next-hop с учётом существующих маршрутов в RIB
<code>set class-map <NAME></code>	Пакетам, попавшим под действие правила, присваивается указанный класс трафика class-map. Класс должен быть заранее создан (подробнее см.)
<code>set vrf <VRF_NAME> [<A.B.C.D>]</code>	Для пакетов, попавших под действие правила, будет использоваться таблица маршрутизации vrf, где VRF_NAME — имя необходимого vrf. Для данного vrf можно при необходимости указать IP-адрес next hop

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>] [<TTL>]`.

Таблица 75 — Дополнительные параметры при указании **udp**

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

Таблица 76 — Дополнительные параметры при указании **tcp**

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	<p>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</p> <ul style="list-style-type: none"> - 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. <p>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</p>
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)

Параметр	Описание
<code>www</code>	TCP(HTTP-80)
<code><0-65535></code>	Точный номер порта, любое значение из указанного диапазона
<code>range <0-65535> <0-65535></code>	Номер порта входит в диапазон

21.3.3 Фильтрация фрагментов пакетов

Когда пакеты слишком велики (в связи с ограничениями MTU) для передачи по сети, они делятся на несколько меньших фрагментов. Первый фрагмент в этом наборе называется первичный фрагмент и содержит заголовки четвёртого уровня. Последующие, вторичные фрагменты не содержат L4 заголовков.

В `filter-map ipv4` существует возможность создания правила исключительно для вторичных фрагментов. Для их фильтрации с помощью команды `match` используется ключевое слово `fragments`. Нефрагментированные пакеты и первичные фрагменты такой командой отфильтрованы не будут.

Пример:

```
filter-map ipv4 TEST_TCP 10
match fragments any host 20.0.0.3 host 10.0.0.2
set reject
```

Данная `filter-map`, прикреплённая к интерфейсу, будет запрещать дальнейший транзит вторичных фрагментов пакетов любых L4 протоколов от хоста 20.0.0.3 к хосту 10.0.0.2.

Попытка ввести правило вида `match fragments udp 10.10.10.0/24 20.20.20.0/24 eq 22`, т.е. правило с указанием L4-порта, приведёт к ошибке, поскольку в заголовках вторичных фрагментов не может быть указан L4-порт.

21.3.2.1 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 правилу, также будет выполняться ассерт, 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 правилу, также будет выполняться ассерт, 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 задаётся 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
```

21.3.3 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.

Таблица 77 — Команды просмотра L2 filter-map

Консоль	Комментарий
<code>ecorouter# show filter-map ethernet</code>	Вывести информацию обо всех списках фильтрации L2
<pre>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</pre>	Вывод информации обо всех списках фильтрации L2
<code>ecorouter# show filter-map ethernet FILTER</code>	Вывести информацию о списке фильтрации с именем FILTER
<pre>Filter map FILTER Filter 10 match host 0000.0000.0001 host 0000.0000.0004</pre>	Вывод информации о списке с именем FILTER

Консоль	Комментарий
<pre>match host 0000.0000.0001 any 0x806 set accept</pre>	

21.3.3.1 Просмотр счётчиков

Для просмотра показателей счётчиков для L2 списков фильтрации используется команда режима администрирования `show counters port <NAME> filter-map {in | out}`.

Параметры команды описаны в таблице ниже.

Таблица 78 — Параметры команды просмотра счётчиков

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

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

Таблица 79 — Команды просмотра счётчиков

Консоль	Комментарий
<pre>ecorouter#show counters port te0 filter-map in</pre>	Вывести значения счётчиков filter-map для порта te0 по входящему трафику
<pre>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 set accept Filter 20 [6 packets] match host 0000.0000.0002 any set discard</pre>	Вывод команды

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

Таблица 80 — Отображение списков фильтрации привязанных к данному порту

Консоль	Комментарий
---------	-------------

Консоль	Комментарий
<code>ecorouter# show port te0</code>	Вывести информацию о порте te0
<pre>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</pre>	Вывод команды

21.3.4 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 определённого 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 присоединённых списков доступа L3 на определённом 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
```

21.3.5 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 её 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 её параметров.

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

Таблица 81 — Параметры subscriber-policy

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
```

21.3.5.1 Создание и настройка filter-map policy

Создание filter-map policy производится при помощи команды конфигурационного режима `filter-map policy ipv4 <NAME>`, где **NAME** — имя создаваемой сущности.

```
ecorouter(config)#filter-map policy ipv4 <NAME>
ecorouter(config-filter-map-policy-ipv4)#
```

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):

- 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.
- 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.
- 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.

Таблица 82 — Общие параметры filter-map policy

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	<p>Значение поля protocol. Может быть указано значение поля в диапазоне (0-255) или одно из следующих обозначений:</p> <ul style="list-style-type: none"> - ipinip; - icmp; - gre; - igmp; - pim; - 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)
SRC_ADDRESS	<p>IP-адрес источника, задается в одном из следующих форматов:</p> <ul style="list-style-type: none"> - 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	<p>IP-адрес назначения, задается в одном из следующих форматов:</p> <ul style="list-style-type: none"> - A.B.C.D/M (IP-адрес с маской), - A.B.C.D K.L.M.N (IP-адрес с инверсной маской),

Параметр	Описание
	- 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	
set accept	Разрешить. Если в subscriber-policy, где используется данная filter-map policy, задана полоса пропускания (параметр bandidwth), то для этого типа трафика будет применено ограничение скорости до указанных в bandwidth значений
set discard	Запретить без отправки ICMP-уведомления
set nexthop <A.B.C.D>	Указать IP-адрес next hop. Пакеты, попавшие под действие правила, отсылаются на адрес next-hop с учётом существующих маршрутов в RIB
set redirect	Перенаправить HTTP GET на указанный <REDIRECTNAME>, где <REDIRECTNAME> — имя заранее заданного URL (адрес для перенаправления должен начинаться с http://). Пример настройки перенаправления приведён ниже.
set reject	Запретить с отправкой ICMP-уведомления
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 можно при необходимости указать IP-адрес next hop

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>]`.

Таблица 83 — Дополнительные параметры при указании **udp**

Параметр	Описание
PORT_CONDITION	Условие для значения порта. Может быть указано одно из следующих значений: <code>{{eq gt lt} {tftp bootp <0-65535>} range <0-65535> <0-65535>}</code>
Значения PORT_CONDITION	
eq	Номер порта равен

Параметр	Описание
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>]`.

Таблица 84 — Дополнительные параметры при указании **tcp**

Параметр	Описание
PORT_CONDITION	Условие для значения порта. Может быть указано одно из следующих значений: <code>{eq gt lt} {ftp ssh telnet www <0-65535>} range <0-65535> <0-65535>}</code>
FLAG	<p>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</p> <ul style="list-style-type: none"> - 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.

Параметр	Описание
	Можно перечислить несколько флагов через пробел. При этом правило сработает, если в пакете будут установлены все перечисленные флаги. Например, правило <code>not-rst syn</code> аск сработает, если пакет содержит флаги SYN и ACK, но не содержит RST
Значения PORT_CONDITION	
<code>eq</code>	Номер порта равен
<code>gt</code>	Номер порта больше, чем
<code>lt</code>	Номер порта меньше, чем
<code>ftp</code>	TCP(21)
<code>ssh</code>	TCP(22)
<code>telnet</code>	TCP(23)
<code>www</code>	TCP(HTTP-80)
<code><0-65535></code>	Точный номер порта, любое значение из указанного диапазона
<code>range <0-65535> <0-65535></code>	Номер порта входит в диапазон

21.3.5.2 Задание адреса для перенаправления

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

21.3.5.3 Пример настроек для обработки трафика в абонентской сессии

В данном примере настроен статический IPoE.

В результате выполнения приведённых 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
  ...
```


22 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 определённые 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**.

22.1 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 85 — Условия команды **match**

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



Условие	Описание
<code>extcommunity</code>	Наличие в BGP маршруте атрибута <code>extcommunity</code> , который содержит данные, совпадающие с указанными в <code>ip extcommunity-list</code>
<code>interface</code>	Совпадение с выходным интерфейсом локального маршрутизатора на основе таблицы маршрутизации
<code>ip address <policy-filter-list></code>	Сопоставление префикса с <code>policy-filter-list</code>
<code>ip address <prefix-list></code>	Сопоставление префикса с <code>prefix-list</code>
<code>ip nexthop</code>	Проверяется <code>next-hop</code> адрес маршрута
<code>ip peer</code>	Проверяется BGP сосед для определенного префикса
<code>metric</code>	Проверяется метрика маршрута
<code>origin</code>	Проверяется значение атрибута <code>origin</code>
<code>route-type</code>	Проверяет тип маршрута для OSPF и IS-IS (<code>external</code> , <code>internal</code> , <code>type-1</code> , <code>type-2</code>)
<code>tag</code>	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**.

22.2 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 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 строка 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>**.

23 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-источника туннеля и точкой-назначения туннеля.

23.1 GRE

GRE (Generic Routing Encapsulation) — протокольный механизм, использующий IP (UDP) как транспортный протокол. GRE может быть использован для переноса различных протоколов внутри себя.

Возможности:

- GRE-туннели предоставляют возможность создавать многопротокольные локальные сети поверх однопротокольной магистральной сети.
- GRE-туннели предоставляют обходные пути для реализации сетей содержащих протоколы с ограниченным количеством переходов (hops).
- GRE-туннели объединяют разрозненные подсети.
- GRE-туннели позволяют реализовать соединения VPN поверх WAN-сетей.

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.

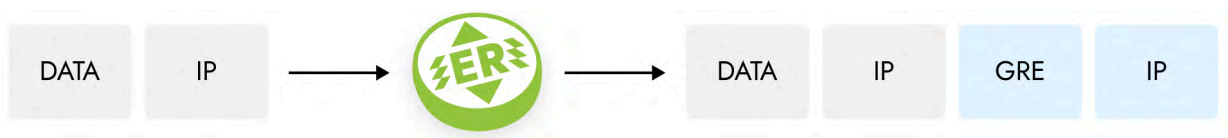


Рисунок 26

23.1.1 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.

Таблица 86 — команды настройки

Команда	Описание
<code>interface tunnel.<номер></code>	Создание интерфейса туннеля, где номер произвольное число
<code>ip mtu <значение></code>	Задание значения mtu для интерфейса
<code>ip tunnel <source IP> <destination IP> mode <eoip gre ipip></code>	Задание IP-адресов начала и конца туннеля и типа туннеля

23.1.2 Keepalive в GRE

GRE-туннель не может автоматически отключиться, когда удалённая сторона недоступна. Туннельный интерфейс остаётся включённым, маршруты остаются в таблице, трафик идёт в никуда. Механизм keepalive — решает эту проблему, добавляя туннелям способность отслеживать состояние удалённой стороны.

Туннельный keepalive механизм работает даже если удалённая сторона не поддерживает keepalive. Сообщение keepalive отправляется удалённой стороне. При получении keepalive маршрутизатор декапсулирует GRE-пакет, в котором находится ещё один GRE-пакет с IP-заголовком и keepalive внутри, после чего маршрутизатор пересылает этот пакет по новому заголовку обратно отправителю keepalive.

Механизм keepalive включается в режиме конфигурации туннельного интерфейса командой `ip tunnel keepalive <0-65535> <1-255>`:

- `<0-65535>` — периодичность отправки сообщения keepalive в секундах. Значение по умолчанию — 10 секунд. Значение равное нулю отключает функцию keepalive.
- `<1-255>` — допустимое количество не вернувшихся keepalive сообщений. Значение по умолчанию равно трём. Система ожидает ответное keepalive в течении одной секунды; если сообщение не получено, счётчик попыток увеличивается на единицу. При достижении установленного значения туннель отключается.

- После отключения туннеля отправка сообщений keeralive продолжается с заданной периодичностью. При получении ответного keeralive туннель восстанавливается.

В EcoRouterOS существует возможность статически задать входящий и исходящий IP адреса внутреннего заголовка GRE пакетов keeralive. Это делается для того чтобы механизм keeralive проверял не только доступность конца туннеля, но и статус удалённого туннельного интерфейса:

```
ip tunnel keeralive nat-avoidance source <A.B.C.D>
```

```
`ip tunnel keeralive nat-avoidance destination <A.B.C.D>
```

23.1.3 Пример базовой настройки туннеля GRE

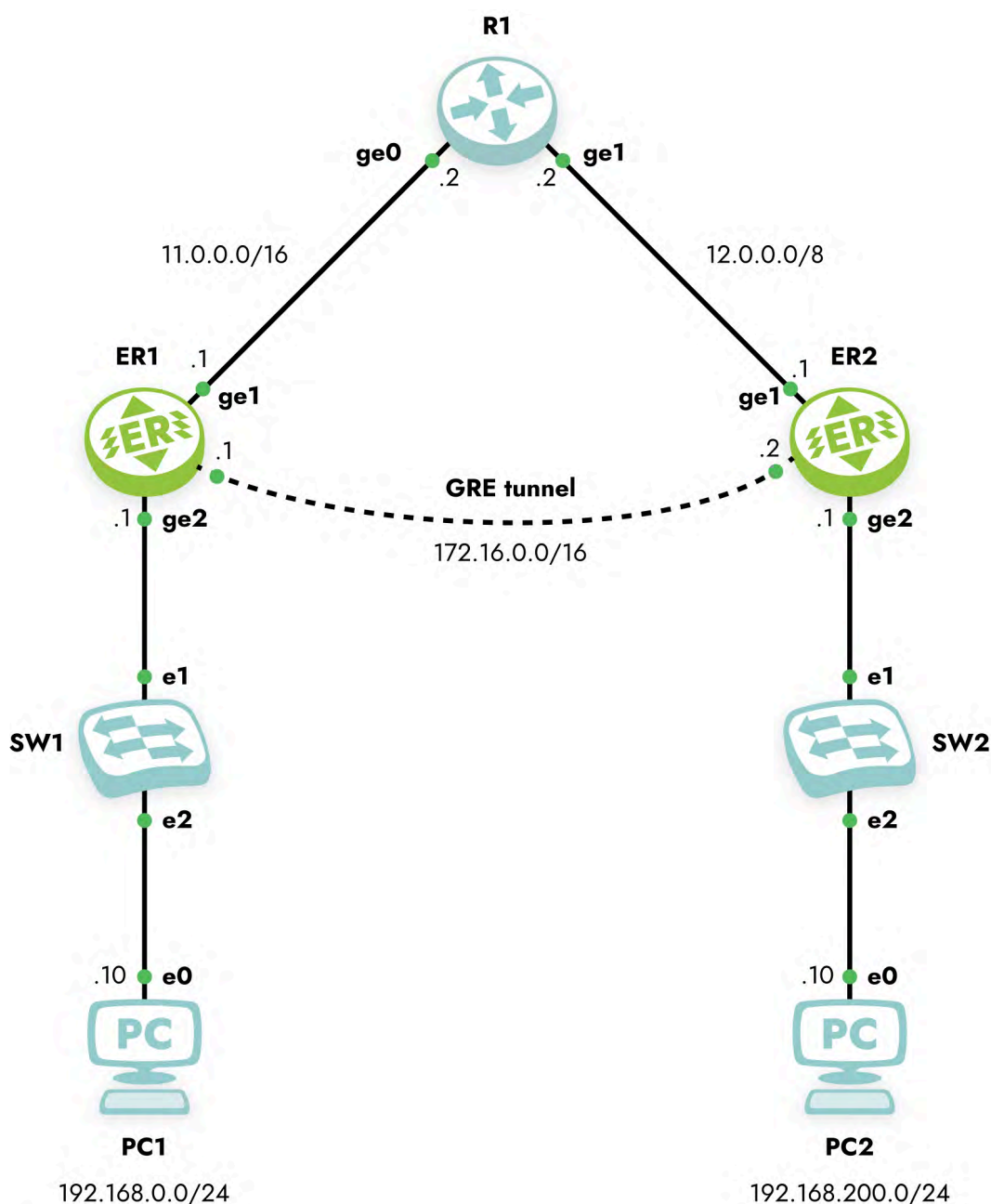


Рисунок 27

Настройте туннель GRE между устройствами ER1 и ER2.

Шаг 1. Настройте интерфейсы и порты на маршрутизаторах EcoRouter.

```
enable
configure

interface 2ge1
  ip address 11.0.0.1/16

interface 2ge2
  ip address 192.168.0.1/24

port ge1
  service-instance 4ge1
  encapsulation untagged
  connect ip interface 2ge1

port ge2
  service-instance 4ge2
  encapsulation untagged
  connect ip interface 2ge2
```

Шаг 2. Создайте интерфейс туннеля с именем tunnel.0

```
ER1(config)#interface tunnel.0
```

Шаг 3. Назначьте IP-адрес

```
ER1(config-if)#ip add 172.16.0.1/16
```

Шаг 4. Установите параметр MTU

```
ER1(config-if)#ip mtu 1400
```

Шаг 5. Задайте режим работы туннеля GRE и адресов начала и конца туннеля

```
ER1(config-if)#ip tunnel 11.0.0.1 12.0.0.1 mode gre
```

Шаг 6. Настройте маршрутизацию трафика в туннель

```
ER1(config)#ip route 12.0.0.0/8 11.0.0.2
ER1(config)#ip route 192.168.200.0/24 172.16.0.2
```



Произведите аналогичную настройку на ER2.

```
enable
configure

interface 2ge1
  ip address 12.0.0.1/8

interface 2ge2
  ip address 192.168.200.1/24

port ge1
  service-instance 4ge1
  encapsulation untagged
  connect ip interface 2ge1

port ge2
  service-instance 4ge2
  encapsulation untagged
  connect ip interface 2ge2

interface tunnel.0
  ip add 172.16.0.2/16
  ip mtu 1400
  ip tunnel 12.0.0.1 11.0.0.1 mode gre

ip route 11.0.0.0/16 12.0.0.2
ip route 192.168.0.0/24 172.16.0.1
```

Шаг 7. Настройте интерфейсы и порты на маршрутизаторе R1.

```
interface 2ge0
  ip address 11.0.0.2/16

interface 2ge1
  ip address 12.0.0.2/8

port ge0
  service-instance 4ge0
  encapsulation untagged
  connect ip interface 2ge0

port ge1
  service-instance 4ge1
```



```
encapsulation untagged  
connect ip interface 2ge1
```

Шаг 8. Установите адрес и шлюз для PC1 и PC2.

```
PC1> ip 192.168.0.10/24 192.168.0.1  
PC2> ip 192.168.200.10/24 192.168.200.1
```

23.1.4 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
```

23.2 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 передаётся с обычным, внутренним IP заголовком дальше.



23.2.1 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.

23.2.2 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 фрейма), то фрейм будет отброшен.

Таблица 87 — Флаги в IP in IP, команды настройки

Команда	Описание
<code>interface tunnel.<номер></code>	Создание интерфейса туннеля, где номер произвольное число
<code>ip mtu <значение></code>	Задание значения mtu для интерфейса
<code>ip tunnel <source IP> <destination IP> mode <gre ipip></code>	Задание ip-адресов начала и конца туннеля и типа туннеля

23.2.3 Пример базовой настройки туннеля IP in IP

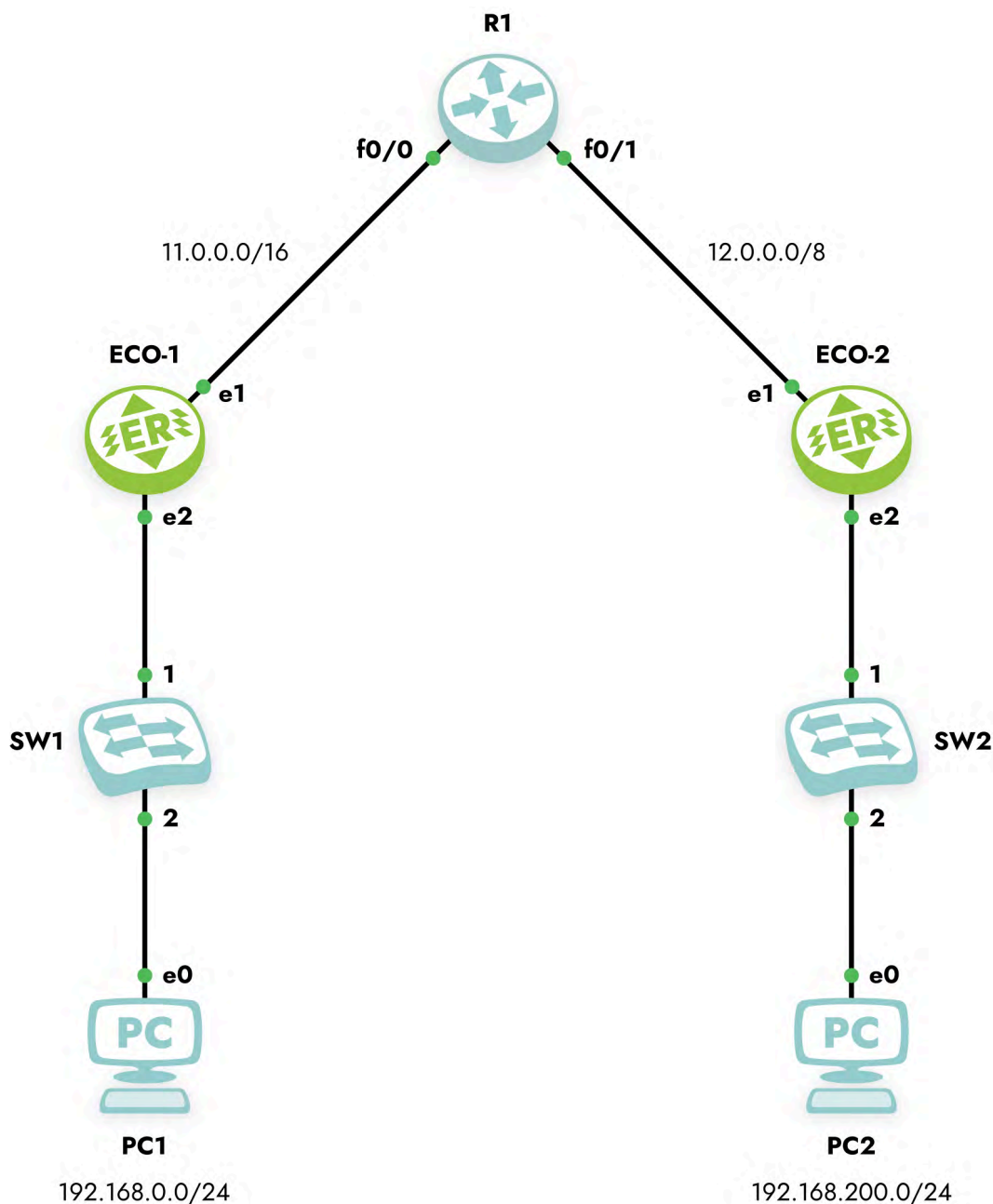


Рисунок 29

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.



24 IPsec

IPsec (Internet Protocol Security) — это фреймворк, который обеспечивает защиту IP-трафика на сетевом уровне, поскольку сам IP-протокол не имеет никаких функций безопасности.

IPsec защищает передаваемые данные с помощью следующих функций:

- **Конфиденциальность:** зашифрованные данные не сможет прочесть никто кроме отправителя и получателя.
- **Целостность:** уверенность в том, что никто не изменил данные в передаваемых пакетах. С помощью хэш-сумм отправитель и получатель могут проверить были ли произведены изменения данных.
- **Аутентификация:** отправитель и получатель могут аутентифицировать друг друга и понимать, что это тот, кто нужно.
- **Защита от повторов:** даже зашифрованные и аутентифицированные пакеты можно перехватить и отправить повторно. Благодаря порядковым номерам IPsec не будет передавать дублирующие пакеты.

Способы использования IPsec:

- Между двумя маршрутизаторами для создания site-to-site VPN который обеспечивает мостовое соединение двух LANов вместе.
- Между firewall и windows host для удалённого доступа по VPN.
- Между двумя linux-серверами для защиты небезопасных протоколов без шифрования, таких как telnet.

24.1 Введение в IPsec

Для начала, чтобы обеспечить защиту каких-либо IP-пакетов, нужны два узла IPsec, которые построят между собой туннель IPsec.

Для создания туннеля IPsec используется протокол IKE (Internet Key Exchange) или статически заданные параметры IPsec. В случае IKE, есть две фазы построения IPsec туннеля: IKE фаза 1 и IKE фаза 2.

На IKE этапе 1 два узла договариваются о протоколах шифрования, аутентификации, хеширования и других протоколах, которые они хотят использовать, а



также о некоторых других необходимых параметрах. На этом этапе устанавливается сессия ISAKMP (Internet Security Association and Key Management Protocol). Это также называется туннелем ISAKMP или туннелем первой фазы IKE.

Набор параметров, которые будут использовать два устройства, называется SA (Security Association). Вот пример двух маршрутизаторов, которые установили туннель фазы 1 IKE:

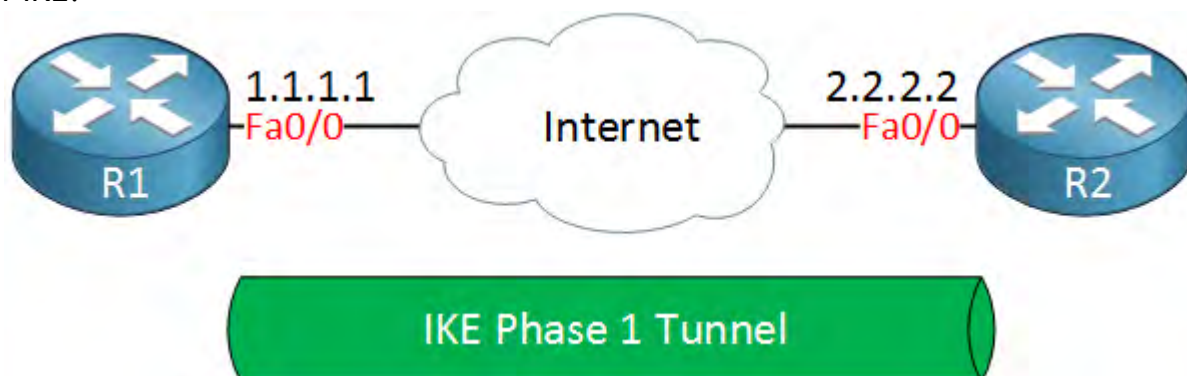


Рисунок 30

Туннель IKE фазы 1 используется только для управляющего трафика. Этот туннель используется как безопасный метод для организации второго туннеля, называемого туннелем IKE фазы 2 или туннелем IPsec. Второй туннель предназначен уже для непосредственной передачи пользовательских данных а также для управляющих данных, например, пакетов keepalive.

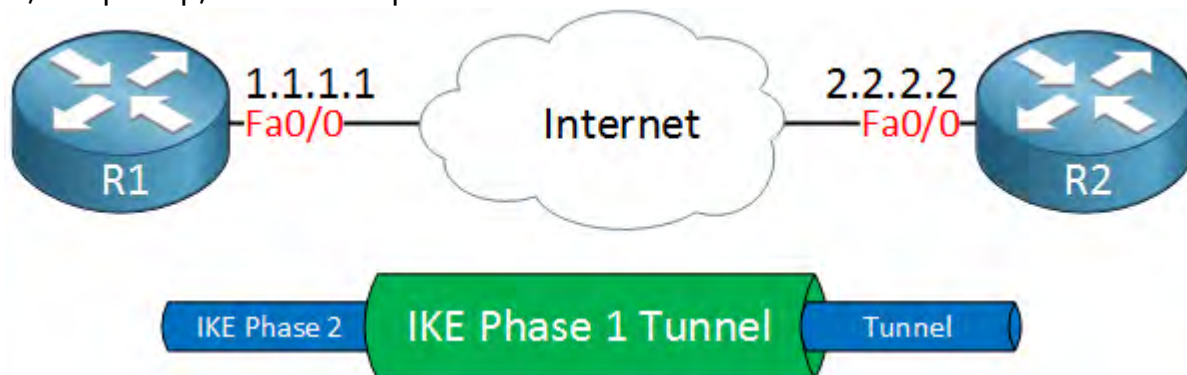


Рисунок 31

После завершения фазы 2 IKE появится туннель фазы 2 IKE (или туннель IPsec), который можно использовать для защиты пользовательских данных. Эти данные будут передаваться через туннель фазы 2 IKE:

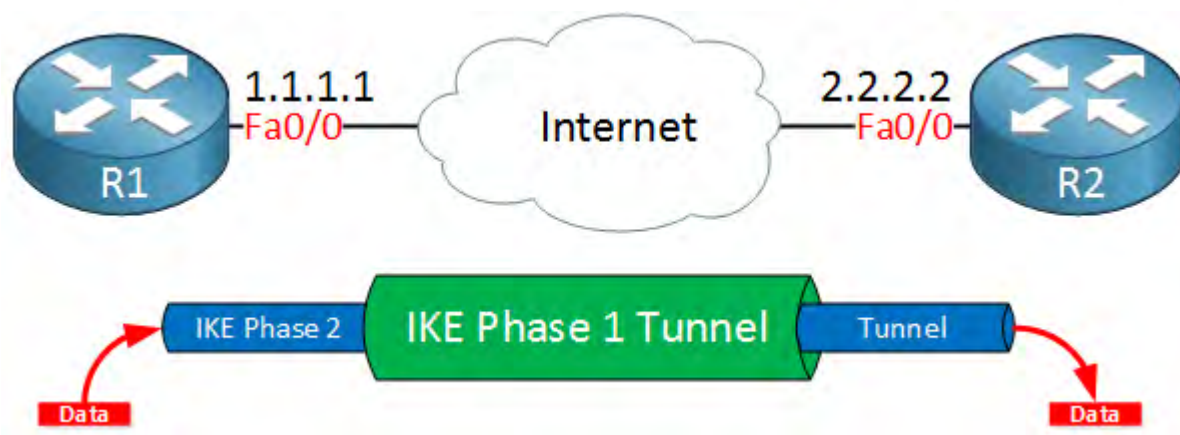


Рисунок 32

IKE создаёт туннели, но не аутентифицирует и не шифрует пользовательские данные. Для этого используется протокол ESP (Encapsulating Security Payload).

Протокол ESP может работать либо в туннельном либо в транспортном режиме. Основное различие между ними заключается в том, что в транспортном режиме используется исходный IP-заголовок, а в туннельном — новый. Диаграмма ниже даёт наглядное представление об этом:

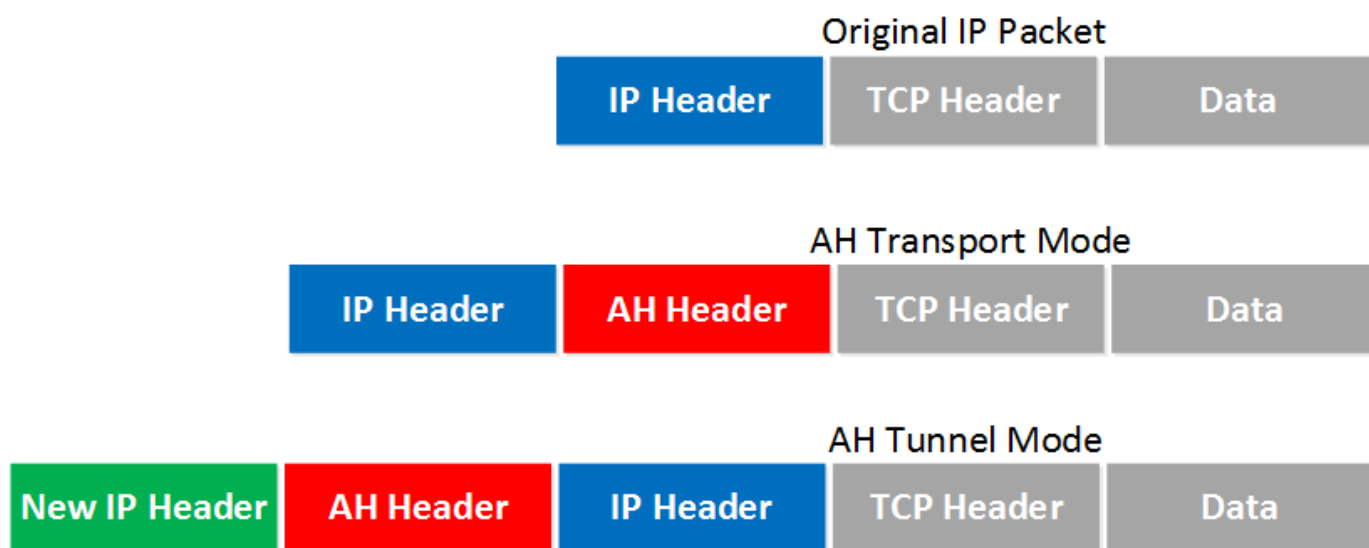


Рисунок 33

Транспортный режим часто используется между двумя устройствами, которым требуется защитить часть незащищённого трафика (например, трафик Telnet). Туннельный режим обычно используется для VPN типа «сеть-сеть», где требуется инкапсуляция исходного IP-пакета, поскольку это в основном частные IP-адреса, которые невозможно маршрутизировать через Интернет. Ниже будет подробно рассказано об этих двух режимах.

Весь процесс IPsec состоит из следующих четырёх шагов:

- Инициирование: инициирование IPsec происходит после завершения конфигурации профиля и создания crypto-map.

- IKE фаза 1: согласуются параметры для построения туннеля фазы 1 IKE (туннеля ISAKMP).
- IKE фаза 2: внутри туннеля фазы 1 IKE создаётся туннель фазы 2 IKE (туннель IPsec).
- Передача данных: пользовательские данные передаются в защищённом виде через туннель IKE фазы 2.

24.1.1 IKE фаза 1

Главное назначение IKE фазы 1 это установить защищённый туннель для IKE фазы 2. Этот процесс можно разделить на три простых шага:

Шаг 1: Переговоры

Узел, данные которого необходимо защитить, инициирует согласование фазы 1 IKE. Два узла согласуют следующие вопросы:

- Хеширование: используется алгоритм хеширования MD5 или SHA (sha1, sha256, sha512) для проверки целостности данных.
- Аутентификация: каждый одноранговый узел должен подтвердить, кто он с помощью предварительного ключа (pre-shared key).
- Назначение группы криптографического алгоритма DH (Диффи-Хеллмана): группа DH определяет стойкость ключа, который используется в процессе обмена ключами. Чем больше номер группы, тем надёжнее она защищена, но и вычисляется дольше.
- Назначение времени жизни: как долго будет работать туннель IKE фазы 1. Чем короче время жизни, тем туннель надёжнее, потому что его перестройка означает, что будут использованы новые криптографические ключи. В EcoRouterOS значение по умолчанию составляет 14 400 секунд (4 часа), но варьируется в пределах 10% в меньшую сторону во избежание коллизий.
- Шифрование: какой алгоритм будет использован для шифрования: 3DES или AES (AES128, AES192, AES256).

Шаг 2: Обмен ключами DH

После успешного согласования оба узла будут знать, какую политику безопасности использовать. Теперь они будут использовать согласованную группу DH для обмена ключами. В результате у обоих узлов будет общий ключ.

Шаг 3: Аутентификация



На последнем шаге оба узла аутентифицируют друг друга, используя метод аутентификации, согласованный в ходе переговоров. Успешное прохождение аутентификации означает завершение фазы 1 IKE. В результате получается двусторонний туннель фазы 1 IKE (он же туннель ISAKMP). Это означает, что оба узла могут отправлять и получать данные по этому туннелю.

Три шага, описанных выше, могут быть проделаны в двух режимах: в основном режиме и в агрессивном. В основном режиме придётся использовать шесть сообщений, тогда как в агрессивном — лишь три. Агрессивный режим быстрее основного, поскольку добавляет всю информацию, необходимую для обмена ключами DH, в первые два сообщения. Основным режим считается более безопасным, поскольку идентификация зашифрована, в то время как агрессивный режим делает это в открытом виде.

После этих шагов у обоих участников есть всё необходимое: последнее сообщение от инициатора — это хеш, который используется для аутентификации. Туннель фазы 1 IKE теперь запущен и работает, и всё готово для перехода к фазе 2 IKE.

24.1.2 IKE фаза 2

Туннель IKE фазы 2 (туннель IPsec) фактически используется для защиты пользовательских данных. Существует только один режим для построения туннеля фазы 2 IKE, который называется быстрым режимом.

Как и в фазе 1 IKE, одноранговые узлы договариваются о ряде пунктов:

- Протокол IPsec: использовать ли AH (проверку целостности) или ESP (шифрование).
- Режим инкапсуляции: транспортный или туннельный режим.
- Шифрование: какой алгоритм шифрования будет использован 3DES, AES128, AES192 или AES256?
- Аутентификация: какой алгоритм аутентификации используется: MD5, SHA1, SHA256 или SHA512.
- Обмен DH (необязательно): используется для PFS (Perfect Forward Secrecy, совершенная прямая секретность). PFS не является обязательным и заставляет одноранговые узлы снова запустить обмен DH для генерации нового общего ключа в каждом быстром режиме IKE фазы 2.

В IKEv2 нет основного или агрессивного режима для фазы 1, а также нет быстрого режима для фазы 2. Однако он по-прежнему состоит из двух фаз: первая называется IKE_SA_INIT, а вторая — IKE_AUTH. Для всего обмена требуется всего четыре сообщения.

После завершения второй фазы IKE IPsec готов к защите определённых пользовательских данных. Далее будет объяснено, как это будет реализовано.

24.1.3 Протокол ESP

ESP позволяет шифровать IP-трафик. Его можно использовать в транспортном или туннельном режиме.

В транспортном режиме используется исходный IP-заголовок, и под ним добавляется заголовок ESP. Вот как это выглядит:

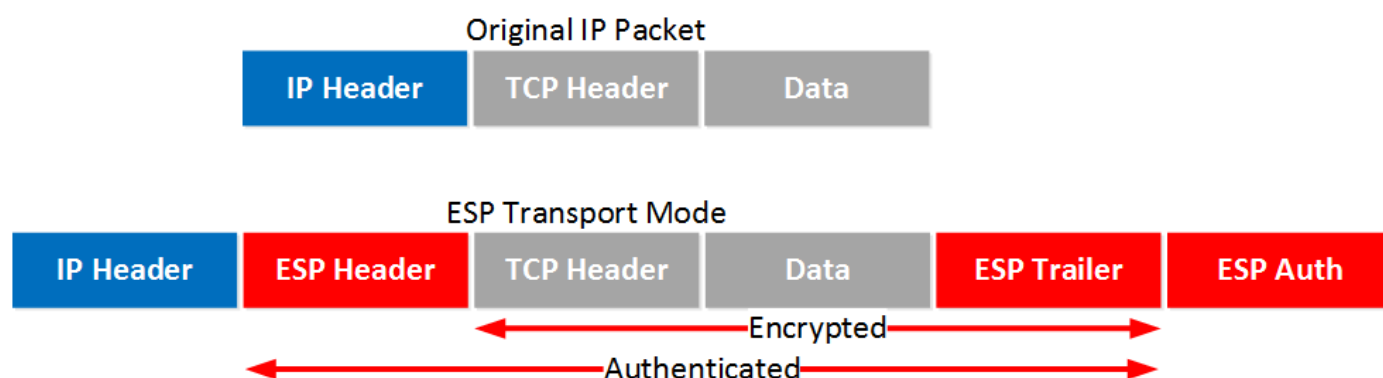


Рисунок 34

Как видно из рисунка выше, к пакету добавляется заголовок и концевик ESP. Транспортный уровень (например TCP) и полезная нагрузка будут зашифрованы. Транспортный режим ESP также обеспечивает аутентификацию, но, в отличие от АН, она распространяется не на весь IP-пакет. IP заголовок передаётся открытым текстом, а всё прочее зашифровано.

В туннельном режиме используется новый заголовок IP, что полезно в случаях организации VPN между удалёнными сетями:

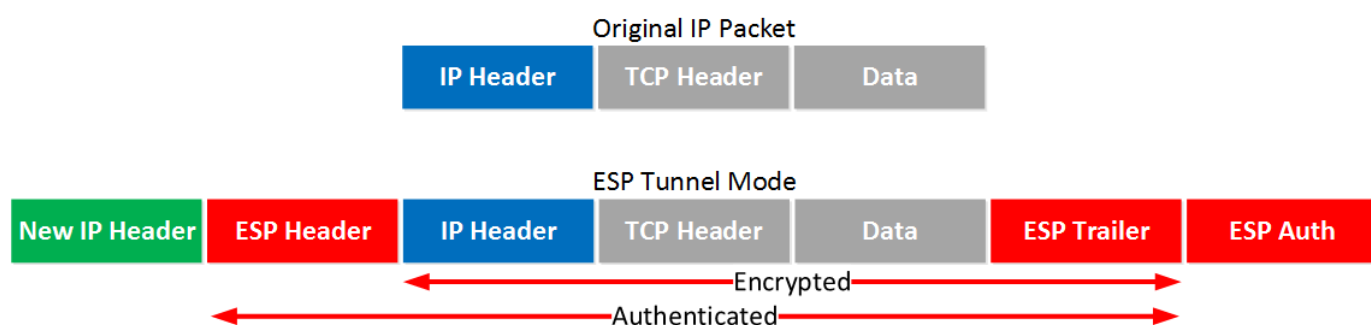


Рисунок 35

Туннельный режим похож на транспортный, но к пакету добавляется новый заголовок. Исходный IP-заголовок теперь также зашифрован.

24.2 Статический IPsec

В операционной системе маршрутизатора предусмотрена возможность создания статических IPsec-туннелей, то есть туннелей без автоматического создания, установления, изменения и удаления SA (Security Association) между двумя хостами сети посредством протокола IKE (Internet Key Exchange). Все используемые туннелем ключи, алгоритмы и протоколы задаются вручную и должны совпадать на обоих концах туннеля.

На данный момент устройство поддерживает только протокол защиты передаваемых данных ESP (Encapsulating Security Payload), туннельный и транспортный режимы работы. Для шифрования доступны алгоритмы AES, 3DES, а для хеширования — MD5, SHA1/256/512.

Основные параметры туннеля задаются в профиле IPsec. Для перехода в режим его конфигурирования необходимо в глобальном режиме конфигурирования ввести команду `crypto-ipsec profile <NAME> manual`, где NAME — имя профиля, а ключ `manual` означает, что туннель является статическим.

В первую очередь необходимо задать режим работы. Зададим туннельный режим. Данный режим задаётся командой `mode tunnel`.

Далее следует задать ключевые параметры IPsec (ESP) туннеля в двух направлениях: входящем, т. е. от удалённой точки до локального устройства (inbound) и исходящем, т. е. от локального устройства до удалённой точки (outbound). Переход в режим конфигурирования туннеля в исходящем или входящем направлении производится командами `ipsec-outbound esp` и `ipsec-inbound esp` соответственно.

Для каждого направления туннеля необходимо задать основные параметры для организации SA:

- `sp-index <NUMBER>` — номер SPI (Security Parameter Index);
- `authenticator sha1 | sha256 | sha512 | md5 <KEY>` — выбор алгоритма хеширования и задание ключа в шестнадцатеричном виде;
- `encryption 3des | aes <KEY>` — выбор алгоритма шифрования и задание ключа в шестнадцатеричном виде.

CLI устройства принимает ввод ключа как с префиксом 0x, так и без него в шестнадцатеричном виде. При неверной длине ключа устройство подскажет, какую длину следует использовать.

Заданные для обоих направлений параметры SA должны совпадать на обоих концах туннеля.



Затем с помощью криптографических карт crypto-map необходимо указать, к какому соседу следует применять соответствующий профиль IPsec. Переход в режим конфигурирования криптографической карты производится командой `crypto-map <NAME> <PRIORITY>`, где **NAME** — имя карты, а **PRIORITY** (иначе — номер в последовательности карт) определяет порядок обработки карты. Чем меньше номер, тем выше приоритет и вероятность того, что трафик IPsec будет обработан именно этой картой.

В настройках карты криптографической карты необходимо указать профиль IPsec и соседа, к которому должен быть применён данный профиль:

- `match peer <ADDRESS>`, где **ADDRESS** — IPv4-адрес соседа;
- `set crypto-ipsec profile <NAME>`, где **NAME** — имя профиля.

Ниже приведён пример для криптографической карты с именем TEST.

```
crypto-map TEST 10
  match peer 200.0.0.3
  set crypto-ipsec profile TEST1
crypto-map TEST 20
  match peer 200.0.0.3
  set crypto-ipsec profile TEST2
crypto-map TEST 30
  match peer 200.0.0.3
  set crypto-ipsec profile TEST3
```

При такой конфигурации к пиру могут быть применены 3 профиля, но обработка трафика IPsec от соседа с адресом 200.0.0.3 начнётся профиля TEST1.

Далее необходимо научить маршрутизатор перехватывать трафик, который должен быть обработан IPsec модулем. Для этого следует воспользоваться встроенными функциями списков контроля доступа filter-map ipv4 (см. главу "Списки доступа", раздел "Настройка L3 filter-map").

Для перехвата входящего трафика IPsec от определённого соседа следует создать правило match/set вида:

- `match esp host <Remote ADDRESS> host <Local ADDRESS>`, где **Remote ADDRESS** — IPv4-адрес соседа в туннеле, а **Local ADDRESS** — локальный IPv4-адрес маршрутизатора для IPSec туннеля;
- `set crypto-map <NAME> peer <Remote ADDRESS>`, где **NAME** — имя ранее созданной криптографической карты (crypto-map), а **Remote ADDRESS** — IPv4-адрес соседа в туннеле, для точного соответствия.

Для перехвата трафика IPsec, передаваемого из локальной сети в удалённую сеть, т. е. исходящего трафика, который должен быть зашифрован, следует создать правило



match/set вида:

- `match any <Local NETWORK> <Remote NETWORK>`, где **Local NETWORK** — локальная IPv4-подсеть, а **Remote NETWORK** — удалённая IPv4-подсеть. Таким образом, трафик, передаваемый из локальной подсети в удалённую подсеть, попадёт в туннель и будет зашифрован.
- `set crypto-map <NAME> peer <Remote ADDRESS>`, где **NAME** — имя ранее созданной криптографической карты, а **Remote ADDRESS** — IPv4-адрес соседа в туннеле, для точного соответствия.

Последним действием следует применить командой **set** списки контроля доступа filter-map к необходимым L3-интерфейсам во входящем направлении. Пример для filter-map с именем ipsec_tunnel:

```
interface lan
ip mtu 1500
connect port te2 service-instance lan
ip address 192.168.100.100/24
set filter-map in ipsec_tunnel
!
interface wan
ip mtu 1500
connect port te0 service-instance wan
ip address 200.0.0.100/8
set filter-map in ipsec_tunnel
```

Для вывода информации о настроенных SA предусмотрена команда `show crypto sa`.

Ниже приведён пример настройки IPSec-туннелей для схемы с тремя соседями, LAG для WAN соединения и алгоритмами SHA1/256/512 и 3DES.

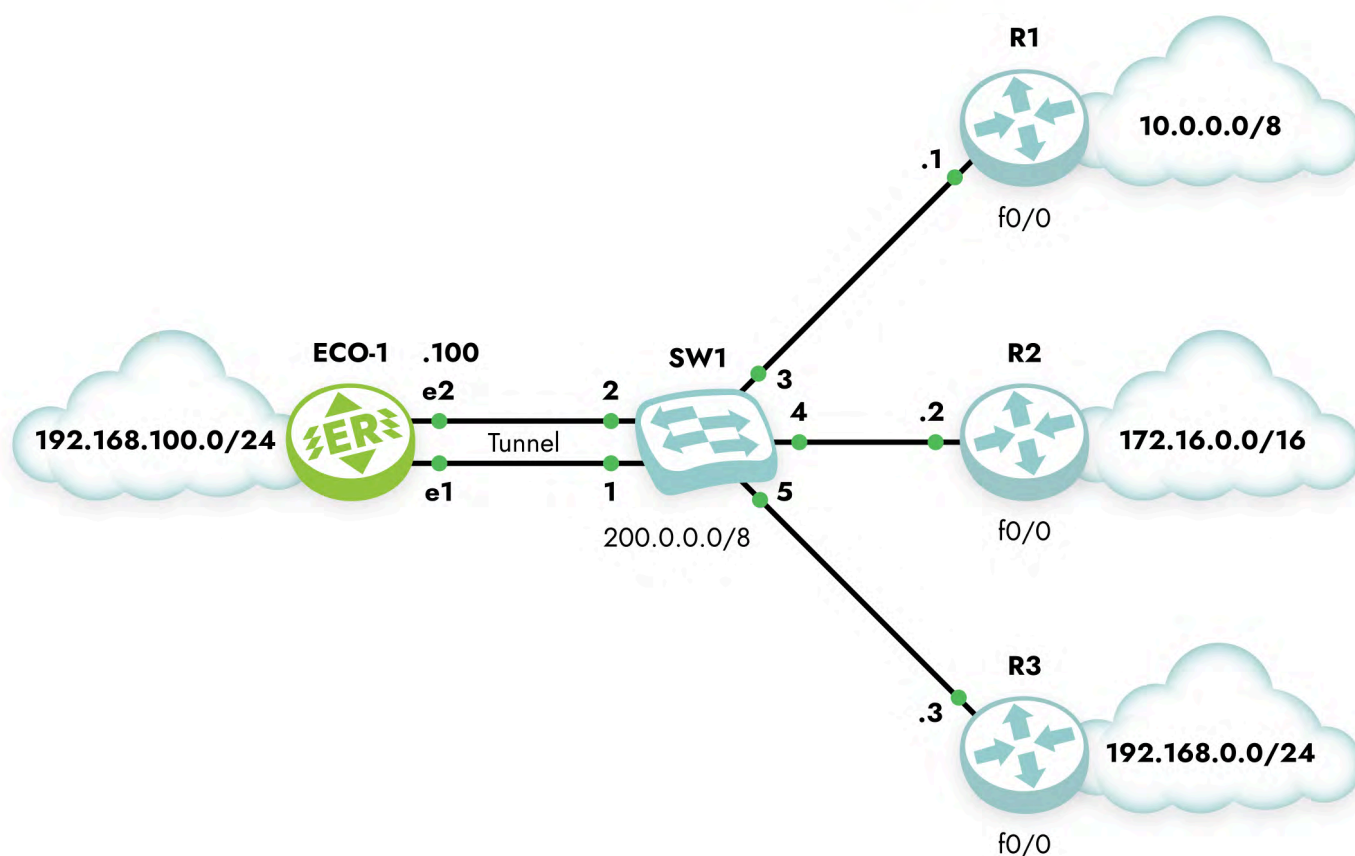


Рисунок 36

```
crypto-ipsec profile test1 manual
mode tunnel
ipsec-outbound esp
  sp-index 1000
  authenticator sha1 0x000102030405060708090a0b0c0d0e0f00000000
  encryption 3des 0x000102030405060708090a0b0c0d0e0faaaaaaaabbbbbbbb
ipsec-inbound esp
  sp-index 1001
  authenticator sha1 0x000102030405060708090a0b0c0d0e0f11111111
  encryption 3des 0x000102030405060708090a0b0c0d0e0faaaaaaaabbbbbbbb
```

```
crypto-ipsec profile test2 manual
mode tunnel
ipsec-outbound esp
  sp-index 2000
  authenticator sha256
0x000102030405060708090a0b0c0d0e0f00000000000102030405060708090a0b
  encryption 3des 0x000102030405060708090a0b0c0d0e0fbbbbbbbbbbcccccccc
ipsec-inbound esp
  sp-index 2001
  authenticator sha256
```

```
0x000102030405060708090a0b0c0d0e0f22222222000000002222222222222222
  encryption 3des 0x000102030405060708090a0b0c0d0e0fbbbbbbbbbbcccccccc

crypto-ipsec profile test3 manual
  mode tunnel
  ipsec-outbound esp
    sp-index 3000
                                authenticator                                sha512
0x000102030405060708090a0b0c0d0e0f00000000000102030405060708090a0b00010203040506
0708090a0b0c0d0e0f00000000000102030405060708090a0b
  encryption 3des 0x000102030405060708090a0b0c0d0e0fccccccccdddddddd
  ipsec-inbound esp
    sp-index 3001
                                authenticator                                sha512
0x000102030405060708090a0b0c0d0e0f3333333300000000333333333333333300010203040506
0708090a0b0c0d0e0f33333333000000003333333333333333
  encryption 3des 0x000102030405060708090a0b0c0d0e0fccccccccdddddddd

crypto-map ipsec 10
  match peer 200.0.0.1
  set crypto-ipsec profile test1

crypto-map ipsec 20
  match peer 200.0.0.2
  set crypto-ipsec profile test2

crypto-map ipsec 30
  match peer 200.0.0.3
  set crypto-ipsec profile test3

filter-map ipv4 ipsec_tunnel 5
  match esp host 200.0.0.1 host 200.0.0.100
  set crypto-map ipsec peer 200.0.0.1

filter-map ipv4 ipsec_tunnel 10
  match any host 192.168.100.1 host 10.0.0.1
  set crypto-map ipsec peer 200.0.0.1

filter-map ipv4 ipsec_tunnel 15
  match esp host 200.0.0.2 host 200.0.0.100
  set crypto-map ipsec peer 200.0.0.2

filter-map ipv4 ipsec_tunnel 20
```



```
match any host 192.168.100.2 host 172.16.0.2
set crypto-map ipsec peer 200.0.0.2

filter-map ipv4 ipsec_tunnel 25
match esp host 200.0.0.3 host 200.0.0.100
set crypto-map ipsec peer 200.0.0.3

filter-map ipv4 ipsec_tunnel 30
match any host 192.168.100.3 host 192.168.0.3
set crypto-map ipsec peer 200.0.0.3

port ae.0
bind te0
bind te1
service-instance wan
encapsulation untagged

port te2
service-instance lan
encapsulation untagged

interface lan
connect port te2 service-instance lan
ip address 192.168.100.100/24
set filter-map in ipsec_tunnel 10

interface wan
connect port ae.0 service-instance wan
ip address 200.0.0.100/8
set filter-map in ipsec_tunnel 10
```

Для полноты изложения рассмотрим пример конфигурации маршрутизатора Cisco R1.

```
hostname R1
!
crypto ipsec transform-set ipsec_tunnel esp-3des esp-sha-hmac
mode tunnel
crypto map ipsec 10 ipsec-manual
set peer 200.0.0.100
set session-key inbound esp 1000 cipher
000102030405060708090a0b0c0d0e0faaaaaaaabbbbbbbb
authenticator
000102030405060708090a0b0c0d0e0f00000000
set session-key outbound esp 1001 cipher
```




```
000102030405060708090a0b0c0d0e0faaaaaaaabbbbbbbb      authenticator
000102030405060708090a0b0c0d0e0f11111111

set transform-set ipsec_tunnel
match address 100

interface Loopback0
ip address 10.0.0.1 255.0.0.0

interface FastEthernet0/0
ip address 200.0.0.1 255.0.0.0
crypto map ipsec

access-list 100 permit ip host 10.0.0.1 host 192.168.100.1
```

EcoRouter поддерживает протокол IKE (Internet Key Exchange) версий 1 и 2. IKEv1 и IKEv2 — это разные версии протокола IKE, который используется для автоматического установления защищённых туннелей IPsec. Хотя они решают одну и ту же задачу (создание SA — Security Associations), IKEv2 является значительной эволюцией, предлагая большую безопасность, эффективность и надёжность.

IKEv1 и IKEv2 настраиваются схожим образом. В примере настройки из следующего подраздела использованы обе версии протокола, в данном подразделе подробно разбирается настройка IKE версии 2.

24.3 Настройка протокола IKEv2

Для включения протокола IKE в конфигурационном режиме необходимо дать команду `crypto-ipsec ike enable`.

Основные параметры туннеля задаются в профиле IPsec. Для перехода в режим настройки профиля необходимо в конфигурационном режиме ввести команду `crypto-ipsec profile <NAME> ike-v2`, где **NAME** — имя профиля, а ключ **ike-v2** означает, что в туннеле будет реализована вторая версия протокола IKE.

В контекстном режиме профиля IPsec задаётся режим работы протокола `transport` или `tunnel`. В транспортном режиме шифруется только полезная нагрузка пакета, в туннельном — пакет целиком, вместе с заголовком.

Далее задаются параметры IPsec (ESP) туннеля для первой и второй фаз IKE. Для перехода в режим настройки каждой из фаз нужно дать соответствующую команду `ike-phase1` или `ike-phase2`.

Для IKE фазы 1 необходимо задать предложение `proposal` по установке безопасного соединения. Предложение задаётся строкой вида `aes128-sha256-`



`modp2048s256`, где: `aes128` — алгоритм шифрования; `sha256` — алгоритм целостности (HMAC); `modp2048s256` — группа и подгруппа (если есть) алгоритма Диффи-Хеллмана (DH). Полный набор комбинаций можно увидеть в выводе команды `proposal ?`.

Также для IKE фазы 1 необходимо задать предустановленный (заранее согласованный) ключ (Pre-Shared Key, PSK): `auth pre-shared-key <KEY>`. В качестве ключа рекомендуется задавать как минимум 20 случайных символов.

Альтернативным и более полным способом задать предустановленные ключи являются команды `local-psk id <ID> data <KEY>` и `remote-psk id <ID> data <KEY>`, где **ID** это идентификатор локального или удалённого маршрутизатора соответственно, а **KEY** — ключ аналогичный описанному в предыдущем абзаце. ID может быть задан в виде IP-адреса, доменного имени, адреса электронной почты или произвольной строки.

Для IKE фазы 2 прежде всего необходимо установить протокол ESP: `protocol esp`. Необходимо также задать предложение с алгоритмом шифрования и алгоритмом целостности. Пример: `proposal aes256-sha256`.

Команды `local-ts <IP>` и `remote-ts <IP>` являются селекторами трафика (Traffic Selectors), которые определяют, какой именно трафик должен быть защищён с помощью IPsec-туннеля. **IP** записывается в виде одного или нескольких (через запятую) адресов хостов или подсетей. Только трафик, соответствующий `local-ts` и `remote-ts`, будет направляться через зашифрованный туннель. Значения `local-ts` и `remote-ts` должны быть идентичны адресам заданным командой `match` в соответствующем списке контроля доступа `filter-map` фильтрующим трафик для шифрования (см. далее по тексту).

Наконец команда `rekey-time <0-75600>` задаёт таймер в секундах по истечению которого будет производиться обновление ключей шифрования.

Заданные для обоих направлений параметры SA должны совпадать на обоих концах туннеля.

Затем с помощью криптографических карт `crypto-map` необходимо указать, к какому пиру следует применять соответствующий профиль IPsec. Переход в режим конфигурирования криптографической карты производится командой `crypto-map <NAME> <PRIORITY>`, где **NAME** — имя карты, а **PRIORITY** определяет порядок обработки карты.

В настройках карты необходимо указать профиль IPsec и соседа, к которому должен быть применён данный профиль:

- `match peer <ADDRESS>`, где **ADDRESS** — IPv4-адрес соседа;
- `set crypto-ipsec profile <NAME>`, где **NAME** — имя профиля.

Далее следует настроить фильтр-карты. Для каждого маршрутизатора необходимо выделить исходящий трафик который нужно зашифровать и входящий, который нужно дешифровать.

Исходящий (из локальной сети в туннель) трафик фильтруется по адресам локальной и удалённой подсети. Фильтруется любой тип трафика. К фильтр-карте привязана криптографическая карта `crypto-map`, которая, в свою очередь, ссылается на профиль `crypto-ipsec`. Трафик шифруется в соответствии с параметрами указанными в профиле IPsec и отправляется в туннель.

При работе с IKEv2 протоколом, используется только туннельный режим IPsec и технологию NAT-traversal которая инкапсулирует ESP в UDP. Такая инкапсуляция делается для преодоления NAT серверов.

Для дешифрации входящего трафика необходимо создать фильтр-карту с правилом `match`, в котором должны быть указаны: адрес удалённого маршрутизатора в качестве источника данных, адрес локального маршрутизатора в качестве адресата, протокол UDP и порт 4500. Команда `set` должна указывать на ту же криптографическую карту, с помощью которой трафик был зашифрован. После действия такой фильтр-карты будут сняты заголовки UDP и ESP, данные будут дешифрованы и отправлены в локальную сеть.

Последним действием — командой `set` следует применить фильтр-карты к необходимым L3-интерфейсам во входящем направлении.

24.3.1 Пример конфигурации

Разберём сценарий объединения двух локальных сетей.

Постройте схему в соответствии с рисунком ниже.

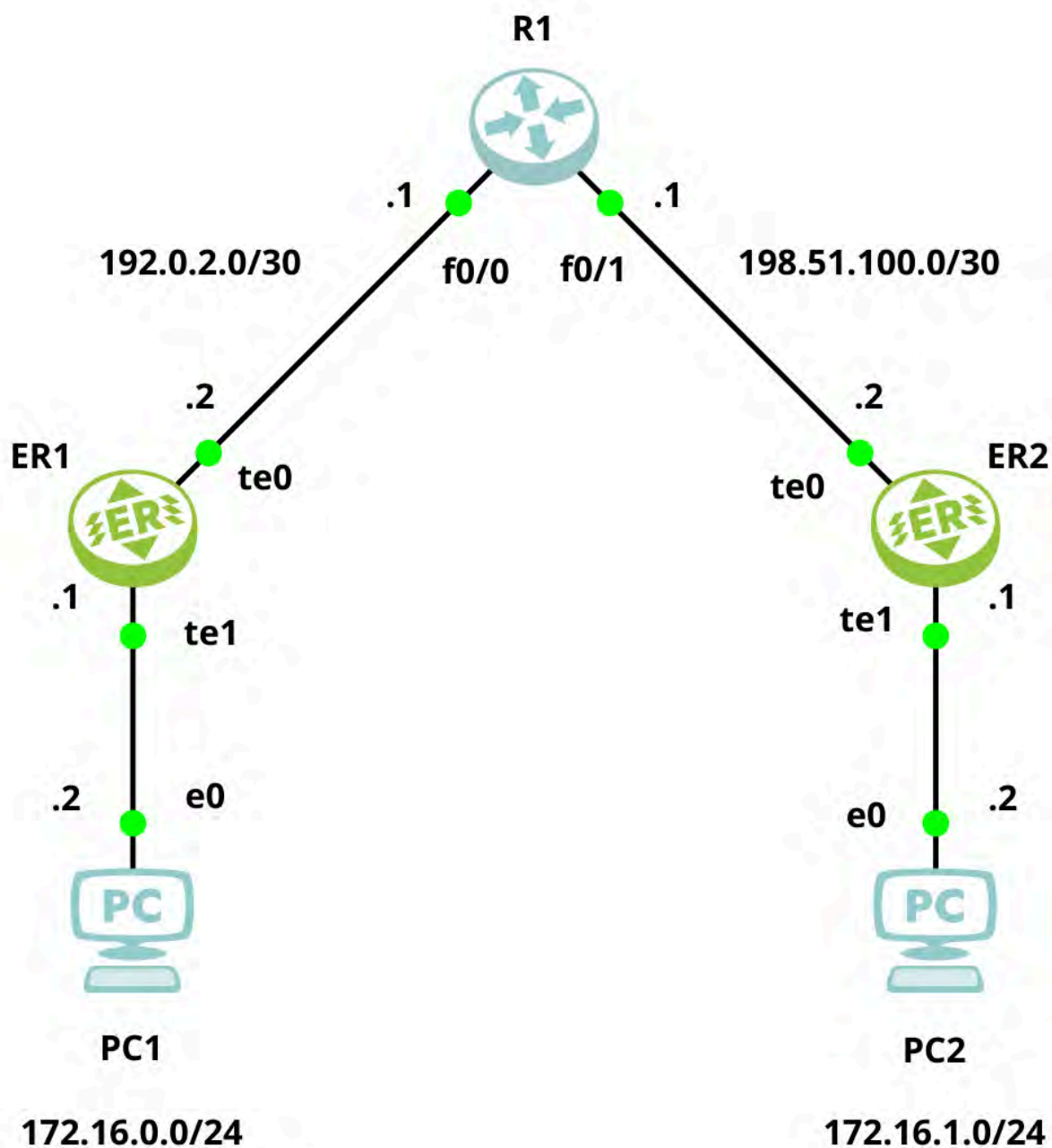


Рисунок 37

Настройте туннели между двумя маршрутизаторами с использованием IKEv2, в соответствии с примерами конфигурации ниже.

Конфигурация ER1:

```
crypto-ipsec ike enable

crypto-ipsec profile CIPROFILE0 ike-v2
mode tunnel
ike-phase1
```



```
proposal aes256-sha256-modp2048
auth pre-shared-key ExamplePassword0
ike-phase2
protocol esp
proposal aes256-sha256
local-ts 172.16.0.0/24
remote-ts 172.16.1.0/24

crypto-map CMAP0 1
match peer 198.51.100.2
set crypto-ipsec profile CIPROFILE0

filter-map ipv4 FMAP0 10
match any 172.16.0.0/24 172.16.1.0/24
set crypto-map CMAP0 peer 198.51.100.2

filter-map ipv4 FMAP0 20
match udp host 198.51.100.2 eq 4500 host 192.0.2.2 eq 4500
set crypto-map CMAP0 peer 198.51.100.2

filter-map ipv4 FMAP0 30
match any any any
set accept

ip route 0.0.0.0/0 192.0.2.1

port te0
service-instance SIWAN
encapsulation untagged

port te1
service-instance SILAN
encapsulation untagged

interface IFWAN
connect port te0 service-instance SIWAN
ip address 192.0.2.2/30
set filter-map in FMAP0 10

interface IFLAN
connect port te1 service-instance SILAN
ip address 172.16.0.1/24
set filter-map in FMAP0 10
```

Конфигурация ER2:

```
crypto-ipsec ike enable

crypto-ipsec profile CIPROFILE0 ike-v2
  mode tunnel
  ike-phase1
    proposal aes256-sha256-modp2048
    auth pre-shared-key ExamplePassword0
  ike-phase2
    protocol esp
    proposal aes256-sha256
    local-ts 172.16.1.0/24
    remote-ts 172.16.0.0/24

crypto-map CMAP0 1
  match peer 192.0.2.2
  set crypto-ipsec profile CIPROFILE0

filter-map ipv4 FMAP0 10
  match any 172.16.1.0/24 172.16.0.0/24
  set crypto-map CMAP0 peer 192.0.2.2

filter-map ipv4 FMAP0 20
  match udp host 192.0.2.2 eq 4500 host 198.51.100.2 eq 4500
  set crypto-map CMAP0 peer 192.0.2.2

filter-map ipv4 FMAP0 30
  match any any any
  set accept

ip route 0.0.0.0/0 198.51.100.1

port te0
  service-instance SIWAN
  encapsulation untagged

port te1
  service-instance SILAN
  encapsulation untagged

interface IFWAN
  connect port te0 service-instance SIWAN
  ip address 198.51.100.2/30
```



```
set filter-map in FMAP0 10

interface IFLAN
  connect port te1 service-instance SILAN
  ip address 172.16.1.1/24
  set filter-map in FMAP0 10
```

24.3.2 Команды просмотра

Посмотреть настроенные подключения можно с помощью команды `show crypto-ipsec ike connections`:

```
ER1>show crypto-ipsec ike connections
connections:
name: CIPROFILE0
local: %any
remote: 198.51.100.2
version: IKEv2
childrens:
name: CIPROFILE0
mode: TUNNEL
local-ts: 172.16.0.0/24
remote-ts: 172.16.1.0/24
rekey-time: 3600
ecorouter>
```

Посмотреть состояние туннелей можно с помощью команды `show crypto-ipsec ike security-associations`:

```
ER1>show crypto-ipsec ike security-associations
IPSec security associations:
name: CIPROFILE0
uniqueid: 2
version: IKEv2
state: ESTABLISHED
local_host: 192.0.2.2
local_port: 4500
local_id: 192.0.2.2
remote_host: 198.51.100.2
remote_port: 4500
remote_id: 198.51.100.2
Is initiator: false
initiator_spi: bc7ba6ca3151919b
```



```
responder_spi: 9e05c52a6bb20a47
nat_local: false
nat_remote: true
nat_fake: false
nat_any: true
encr_alg: AES_CBC
encr_keysize: 256
integ_alg: HMAC_SHA2_256_128
prf_alg: PRF_HMAC_SHA2_256
dh_group: MODP_2048
established: 33
rekey_time: 12965
child-sas:
name: CIPROFILE0
uniqueid: 1
reqid: 1
state: INSTALLED
mode: TUNNEL
protocol: ESP
encap: true
spi_in: 3621fd07
spi_out: 41b7aa27
encr_alg: AES_CBC
encr_keysize: 256
integ_alg: HMAC_SHA2_256_128
esn: false
rekey_time: 3250
life_time: 3927
install_time: 33
local-ts: 172.16.0.0/24
remote-ts: 172.16.1.0/24
ecorouter>
```

24.4 Настройка GRE over IPsec

В этом подразделе представлен более сложный пример настройки маршрутизации с шифрованием, так называемый "GRE over IPsec", в топологии "Hub-and-Spoke".

Сочетание GRE поверх IPsec — это классическое решение для преодоления ключевых ограничений "чистого" IPsec. Только благодаря GRE можно реализовать

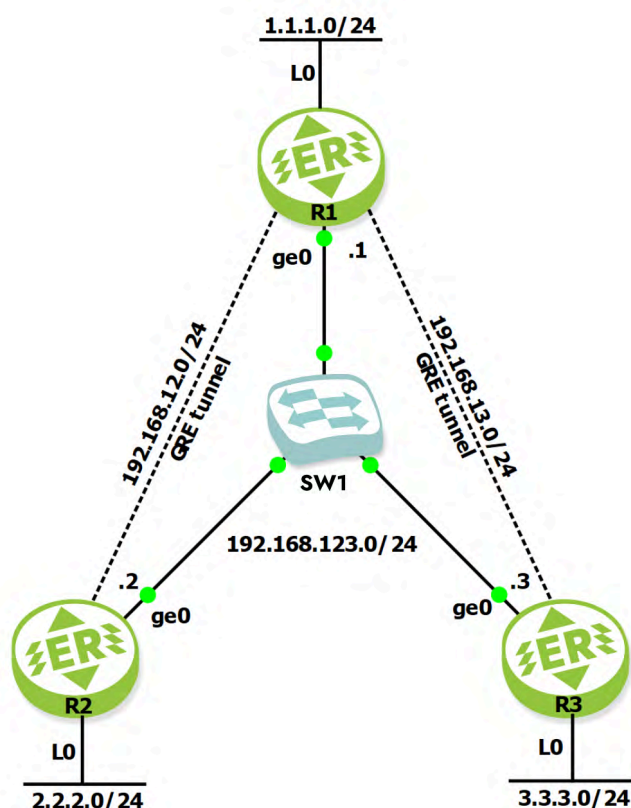


динамическую маршрутизацию с шифрованием. Так происходит потому, что IPsec сам по себе не предполагает многоадресную передачу данных, необходимую, например, для работы протокола OSPF.

GRE over IPsec создаёт поверх безопасного IPsec-туннеля универсальный туннель GRE, который умеет инкапсулировать любой трафик (multicast, широковещательный, не-IP) и создаёт виртуальную сеть между конечными точками.

Hub-and-Spoke — это одна из самых фундаментальных и распространённых топологий сети, особенно в контексте VPN и корпоративных WAN (Wide Area Network). Hub (центральная часть) — это главный, центральный офис или дата-центр. Spokes (расходящиеся лучи) — это удалённые филиалы, магазины или офисы меньшего размера. Вся связь между Spokes происходит только через Hub.

Постройте схему в соответствии с рисунком ниже.



Значение и синтаксис большинства команд раскрыты в других разделах документации. Настройка IPsec, в основном, была дана в предыдущем подразделе "Настройка протокола IKEv2", здесь будут только некоторые замечания.

Обратите внимание, что для обоих туннелей используется одна карта фильтрации, которая последовательно обрабатывает исходящий и входящий трафик R2 затем исходящий и входящий трафик R3 и оканчивается разрешающим правилом для прочего трафика.

Для шифрования туннеля R1–R2 используется протокол IKEv1 и включён режим преодоления NAT `nat-traversal`. Для шифрования туннеля R1–R3 используется протокол IKEv2, для которого NAT traversal включён по умолчанию. Таким образом

данные на входящей стороне туннелей инкапсулируются в датаграммы UDP-порт 4500 и должны фильтроваться правилом вида `match gre host 192.168.123.1 host 192.168.123.2`.

Настройте R1:

```
enable
configure
hostname R1

# Настройка интерфейсов и портов
interface loopback.0
ip address 1.1.1.1/24

interface 2ge0
ip address 192.168.123.1/24
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0

# Настройка туннелей GRE
interface tunnel.12
ip address 192.168.12.1/24
ip mtu 1400
ip tunnel 192.168.123.1 192.168.123.2 mode gre

interface tunnel.13
ip address 192.168.13.1/24
ip mtu 1400
ip tunnel 192.168.123.1 192.168.123.3 mode gre

# Настройка маршрутизации OSPF
router ospf 1
network 1.1.1.0 0.0.0.255 area 0
network 192.168.12.0 0.0.0.255 area 0
network 192.168.13.0 0.0.0.255 area 0
exit

# Включение протокола IKE
crypto-ipsec ike enable

# Настройка профилей IPsec
crypto-ipsec profile CIPROFILE12 ike-v1
```



```
nat-traversal
mode tunnel
ike-phase1
    proposal aes256-sha256-modp2048
    auth pre-shared-key ExamplePassword12
ike-phase2
    protocol esp
    proposal aes256-sha256
    local-ts 192.168.123.1
    remote-ts 192.168.123.2

crypto-ipsec profile CIPROFILE13 ike-v2
mode tunnel
ike-phase1
    proposal aes256-sha256-modp2048
    auth pre-shared-key ExamplePassword13
ike-phase2
    protocol esp
    proposal aes256-sha256
    local-ts 192.168.123.1
    remote-ts 192.168.123.3

# Задание крипто-карт
crypto-map CMAP12 10
match peer 192.168.123.2
set crypto-ipsec profile CIPROFILE12

crypto-map CMAP13 20
match peer 192.168.123.3
set crypto-ipsec profile CIPROFILE13

# Задание карт фильтрации
filter-map ipv4 FMAP123 5
    match gre host 192.168.123.1 host 192.168.123.2
    set crypto-map CMAP12 peer 192.168.123.2

filter-map ipv4 FMAP123 10
    match udp host 192.168.123.2 eq 4500 host 192.168.123.1 eq 4500
    set crypto-map CMAP12 peer 192.168.123.2

filter-map ipv4 FMAP123 15
    match gre host 192.168.123.1 host 192.168.123.3
    set crypto-map CMAP13 peer 192.168.123.3
```



```
filter-map ipv4 FMAP123 20
  match udp host 192.168.123.3 eq 4500 host 192.168.123.1 eq 4500
  set crypto-map CMAP13 peer 192.168.123.3

filter-map ipv4 FMAP123 30
  match any any any
  set accept

# Присвоение карты фильтрации интерфейсам
interface 2ge0
set filter-map in FMAP123 10

interface tunnel.1
set filter-map in FMAP123 10

interface tunnel.13
set filter-map in FMAP123 10
```

Обратите внимание, что при настройке маршрутизации OSPF указываются не физическая подсеть объединяющая все три маршрутизатора, а подсети образуемые туннелями GRE.

Настройте R2:

```
enable
configure
hostname R2

# Настройка интерфейсов и портов
interface loopback.0
ip address 2.2.2.2/24

interface 2ge0
ip address 192.168.123.2/24
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0

# Настройка туннеля GRE
interface tunnel.21
ip address 192.168.12.2/24
ip mtu 1400
ip tunnel 192.168.123.2 192.168.123.1 mode gre
```



```
# Настройка маршрутизации OSPF
router ospf 1
network 2.2.2.0 0.0.0.255 area 0
network 192.168.12.0 0.0.0.255 area 0
exit

# Включение протокола IKE
crypto-ipsec ike enable

# Настройка профилей IPsec
crypto-ipsec profile CIPROFILE21 ike-v1
nat-traversal
mode tunnel
ike-phase1
proposal aes256-sha256-modp2048
auth pre-shared-key ExamplePassword12
ike-phase2
protocol esp
proposal aes256-sha256
local-ts 192.168.123.2
remote-ts 192.168.123.1

# Задание крипто-карт
crypto-map CMAP21 10
match peer 192.168.123.1
set crypto-ipsec profile CIPROFILE21

# Задание карт фильтрации
filter-map ipv4 FMAP21 10
match gre host 192.168.123.2 host 192.168.123.1
set crypto-map CMAP21 peer 192.168.123.1

filter-map ipv4 FMAP21 20
match udp host 192.168.123.1 eq 4500 host 192.168.123.2 eq 4500
set crypto-map CMAP21 peer 192.168.123.1

filter-map ipv4 FMAP21 30
match any any any
set accept

# Присвоение карты фильтрации интерфейсам
interface 2ge0
```



```
set filter-map in FMAP21 10
```

```
interface tunnel.21  
set filter-map in FMAP21 10
```

Настройте R3:

```
enable  
configure  
hostname R3  
  
# Настройка интерфейсов и портов  
interface loopback.0  
ip address 3.3.3.3/24  
  
interface 2ge0  
ip address 192.168.123.3/24  
port ge0  
service-instance 4ge0  
encapsulation untagged  
connect ip interface 2ge0  
  
# Настройка туннеля GRE  
interface tunnel.31  
ip address 192.168.13.3/24  
ip mtu 1400  
ip tunnel 192.168.123.3 192.168.123.1 mode gre  
  
# Настройка маршрутизации OSPF  
router ospf 1  
network 3.3.3.0 0.0.0.255 area 0  
network 192.168.13.0 0.0.0.255 area 0  
exit  
  
# Включение протокола IKE  
crypto-ipsec ike enable  
  
crypto-ipsec profile CIPROFILE31 ike-v2  
mode tunnel  
ike-phase1  
proposal aes256-sha256-modp2048  
auth pre-shared-key ExamplePassword13  
ike-phase2  
protocol esp
```



```
proposal aes256-sha256
local-ts 192.168.123.3
remote-ts 192.168.123.1

# Задание крипто-карт
crypto-map CMAP31 10
match peer 192.168.123.1
set crypto-ipsec profile CIPROFILE31

# Задание карт фильтрации
filter-map ipv4 FMAP31 10
match gre host 192.168.123.3 host 192.168.123.1
set crypto-map CMAP31 peer 192.168.123.1

filter-map ipv4 FMAP31 20
match udp host 192.168.123.1 eq 4500 host 192.168.123.3 eq 4500
set crypto-map CMAP31 peer 192.168.123.1

filter-map ipv4 FMAP31 30
match any any any
set accept

# Присвоение карты фильтрации интерфейсам
interface 2ge0
set filter-map in FMAP31 10

interface tunnel.31
set filter-map in FMAP31 10
```

Проверьте работоспособность туннелей и маршрутизации командами вида `ping 1.1.1.1 source 2.2.2.2`.

Проверьте статистику фильтрации пакетов командой `show counters interface 2ge0 filter-map in`.

Проверьте настроены ли подключения IPsec командой `show crypto-ipsec ike connections`.

Проверьте состояние туннелей командой `show crypto-ipsec ike security-associations`.

- Поле **state** туннеля IKE фазы 1 должно иметь значение **ESTABLISHED**.
- Поле **state** дочернего туннеля IKE фазы 2 должно иметь значение **INSTALLED**.



Проверьте соединения программой перехвата трафика. Если конфигурация произведена верно, вы должны увидеть передачу зашифрованных пакетов ESP.

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 распространённый 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 определённого IP-адреса на другой, также заранее определённый. 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 определённых 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 определённых 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.

Таблица 88 — Описание команд для настройки NAT на EcoRouter

Команда	Описание
<code>ip nat inside</code>	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"
<code>ip nat outside</code>	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
<code>ip nat source static A.B.C.D Q.W.E.R [vrf]</code>	Команда вводится в конфигурационном режиме (config). В результате выполнения этой команды будет создана статическая трансляция адрес-в-адрес отправителя в направлении inside-to-outside . Параметр vrf является необязательным и без указания определённого vrf правило будет создано для default vrf
<code>ip nat destination static A.B.C.D Q.W.E.R [hairpin] [vrf]</code>	Команда вводится в конфигурационном режиме (config). В результате выполнения этой команды будет создана статическая трансляция адрес-в-адрес получателя в направлении inside-to-outside. Параметр hairpin включает возможность доступа к ресурсу в локальной сети по IP адресу, который используется для доступа к ресурсу из интернета. Параметр vrf является необязательным и без указания определённого vrf правило будет создано для default vrf
<code>ip nat source static network A.B.C.D Q.W.E.R mask [vrf]</code>	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 определённого vrf правило будет создано для default vrf

Команда	Описание
<code>ip nat source static A.B.C.D interface <IF_NAME> [vrf]</code>	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 определённого vrf правило будет создано для default vrf
<code>ip nat pool <POOL_NAME> <RANGE></code>	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
<code>ip nat source dynamic inside pool <POOL_NAME> overload A.B.C.D [vrf]</code>	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 определённого vrf правило будет создано для default vrf
<code>ip nat source dynamic inside pool <POOL_NAME> overload interface <IF_NAME> [vrf]</code>	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 определённого vrf правило будет создано для default vrf
<code>ip nat translation (icmp-timeout tcp-timeout udp-timeout) <30-14400></code>	Команда вводится в конфигурационном режиме (config). В результате выполнения этой команды можно изменить значения по-умолчанию времени жизни трансляций для разных протоколов. Значения по умолчанию 3600 секунд для TCP и 300 секунд для всех остальных протоколов

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 NAT port forwarding

Функционал NAT port forwarding подразумевает статический проброс NAT-портов (открытие портов за NAT) для организации удалённого статического доступа к оборудованию в локальной сети через NAT. Этот функционал позволяет создавать статические (существующие всегда и работающие в разных направлениях передачи трафика) правила трансляций для конкретных IP-адресов источника и получателя, а также указывать для каких TCP/UDP портов эта трансляция предусмотрена. Для создания подобных правил применяется следующая команда конфигурационного режима: `ip nat source static [tcp | udp] <IP src> <port src> <IP dst> <port dst>`

Параметры данной команды описаны в таблице ниже. Все параметры являются обязательными!

Таблица 89 — Параметры команды `ip nat source static`

Параметр	Описание
<code>tcp</code> или <code>udp</code>	Ключевые слова для указания транспортного протокола

Параметр	Описание
IP src	IP-адрес источника
port src	L4 порт источника. Может быть задан диапазон портов, для чего необходимо указать начальное и конечное значения через пробел. Размер диапазона портов источника и получателя должен совпадать (см. пример ниже)
IP dst	IP-адрес получателя
port dst	L4 порт получателя. Может быть задан диапазон портов, для чего необходимо указать начальное и конечное значения через пробел. Размер диапазона портов источника и получателя должен совпадать (см. пример ниже)

Приведём пример использования NAT port forwarding и dynamic PAT:

```
ip nat pool TEST 10.0.0.0-10.0.0.254
ip nat source dynamic inside pool TEST overload interface wan

interface wan
ip address 77.0.0.1/30
ip nat outside

interface lan
ip address 10.0.0.1/24
ip nat inside
```

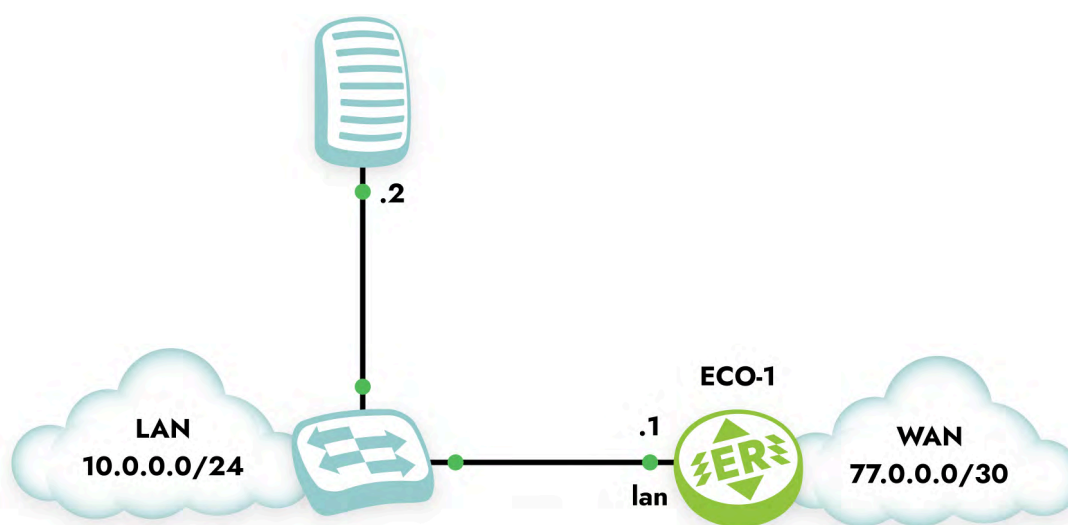


Рисунок 38

Задачу организации удалённого доступа к LAN серверу с адресом 10.0.0.2 можно решить при помощи создания статического правила трансляции и определения

конкретных TCP/UDP портов. Правило, которое позволит подключаться к LAN-серверу со стороны WAN, при попытке TCP подключения на адрес 77.0.0.1 и L4 порт 2222, будет выглядеть следующим образом:

```
ip nat source static tcp 10.0.0.2 22 77.0.0.1 2222
```

Для организации доступа по SSH к хосту 10.0.0.2:22 из подсети 10.0.0.0/24 по адресу и L4 порту 77.0.0.1:2222 следует воспользоваться NAT Hairpin правилом:

```
ecorouter(config)#ip nat destination static tcp 77.0.0.1 2222 10.0.0.2 22 hairpin
```

Пример правила с указанием диапазона портов:

```
ip nat source static tcp 10.0.0.1 100 300 7.0.0.1 400 600
```

25.2 Пример конфигурации static source NAT

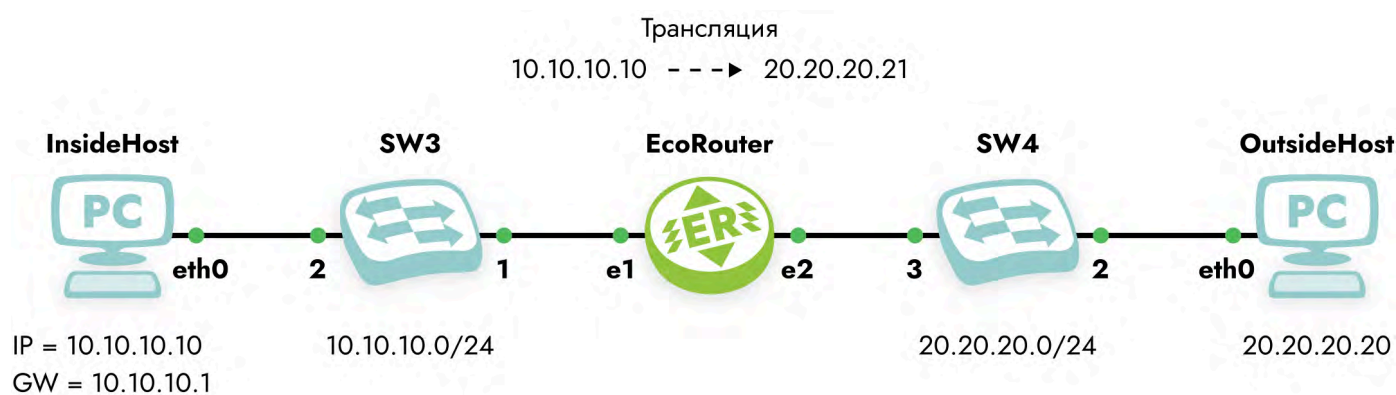


Рисунок 39

Конфигурация EcoRouter:

Настройка интерфейсов и портов:

```
port te0
service-instance si0
encapsulation untagged

port te1
service-instance si1
encapsulation untagged
```

```
interface in
  ip address 10.10.10.1/24
  ip nat inside
  connect port te0 service-instance si0

interface out
  ip address 20.20.20.1/24
  ip nat outside
  connect port te1 service-instance si1
```

Задание статической трансляции:

```
ip nat source static 10.10.10.10 20.20.20.21
```

25.3 Пример конфигурации static source PAT

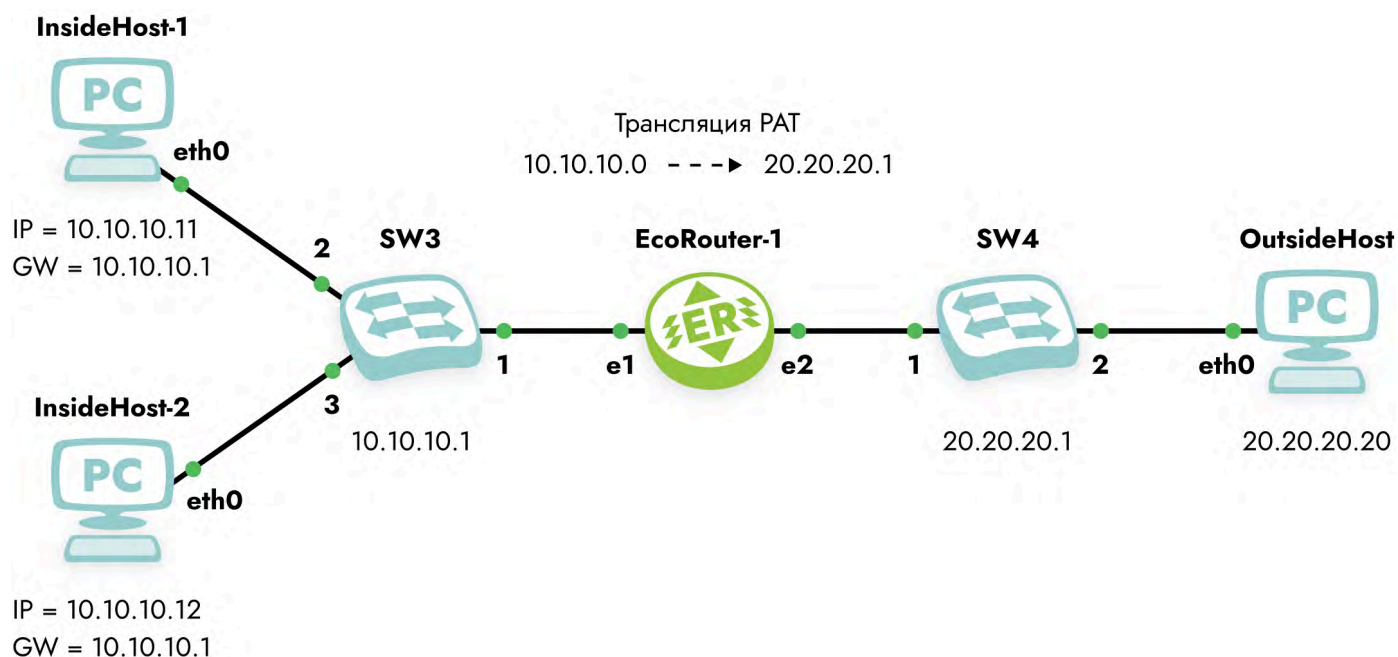


Рисунок 40

Конфигурация EcoRouter:

Настройка интерфейсов и портов:

```
port ge0
  service-instance si0
  encapsulation untagged
```

```
port ge1
  service-instance si1
  encapsulation untagged

interface in
  ip address 10.10.10.1/24
  ip nat inside
  connect port ge0 service-instance si0

interface out
  ip address 20.20.20.1/24
  ip nat outside
  connect port ge1 service-instance si1
```

Создание пула адресов для входящего трафика:

```
ip nat pool POOL 10.10.10.0-10.10.10.20
```

Задание правила трансляций:

```
ip nat source dynamic inside pool POOL overload 20.20.20.21
```



26 Коммутация по меткам MPLS

MPLS (multiprotocol label switching) многопротокольная коммутация по меткам — это технология, использующая при передаче данных от одного узла к другому метки вместо адресов сетевого уровня.

Каждому пакету, проходящему через MPLS-сеть, независимо от типа этого пакета, назначается определенная метка, на основе которой принимается решение о маршрутизации. Содержимое и заголовки пакетов при этом не изучается.

В EcoRouterOS обрабатывается не более двух меток: внешняя — транспортная для передачи по сети MPLS, описанная выше, и внутренняя — сервисная, определяющая принадлежность к сервису.

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 90 — Основные команды настройки MPLS

Команда	Описание
<code>mpls ac-group <имя> <номер></code>	Создание новой группы каналов доступа
<code>mpls bandwidth-class</code>	
<code>mpls disable-all-interfaces</code>	Отключение MPLS на всех интерфейсах
<code>mpls egress-ttl <0-255></code>	Задание значения TTL для выхода с маршрутизатора
<code>mpls enable-all-interfaces</code>	Включение MPLS на всех интерфейсах
<code>mpls ftn-entry <ip-префикс> <метка> <ip-адрес ожидающего интерфейса> <имя исходящего интерфейса></code>	Настройка метки для FEC при входе в MPLS облако
<code>mpls ilm-entry <приходящая метка> <имя входного интерфейса> swap <исходящая метка> <имя исходящего интерфейса> <ip-адрес ожидающего интерфейса> <ip-префикс></code>	Настройка замены метки для FEC при транзите через LSR
<code>mpls ingress-ttl <0-255></code>	Задание значения TTL при входе на маршрутизатор



Команда	Описание
<code>mpls ldp <max-label-value min-label-value></code>	Задание диапазона значений выдаваемых меток. Возможные значения от 16 до 1048575
<code>mpls lsp-tunneling <имя входного интерфейса> <приходящая метка> <исходящая метка> <ip-префикс></code>	
<code>mpls map-route <ip-префикс ip-префикс/маска> <ip-префикс></code>	
<code>mpls propagate-ttl</code>	Управление переносом значения TTL из IP в MPLS
<code>mpls l2-circuit <имя> <ID> <ip-префикс></code>	Создание l2-circuit 5 типа
<code>mpls l2-circuit <имя> <ID> <ip-префикс> mode tagged svlan <VLAN> tpid <TPID></code>	Создание l2-circuit 4 типа

26.1 Настройка статического MPLS

Статический MPLS позволяет вручную настроить все операции с метками на маршрутизаторе. Для хранения используются таблицы ILM (Incoming Label Map) и FTN (FEC To Next hop). Настройки правила ILM применяются для проведения операций замены метки внутри домена MPLS. Настройки правила FTN применяются для навешивания или снятия метки на граничном маршрутизаторе домена MPLS.

На всех участвующих в коммутации по меткам интерфейсах должен быть включён режим `label-switching`.

Задание правила FTN:

```
ecorouter(config)#mpls ftn-entry 2.2.2.2/32 2222 10.0.0.2 e1
```

Где: **2.2.2.2/32** — FEC; **2222** — метка, которая будет добавлена; **10.0.0.2** — адрес следующего маршрутизатора(nexthop); **e1** — интерфейс для отправки.

Задание правила ILM:

```
ecorouter(config)#mpls ilm-entry 1111 e1 swap 2222 e2 10.0.0.1 2.2.2.2/32
```

Где: **1111** — метка, которая ожидается на интерфейсе e1; **2222** — новое значение метки и отправка её через интерфейс e2; **10.0.0.1** — адрес следующего маршрутизатора(nexthop), а **2.2.2.2/32** — FEC.

Для снятия (POP) метки и реализации механизма PHP (Penultimate Hop Popping) на предпоследнем перед Egress LER маршрутизаторе следует использовать правило ILM с параметром **swap** и заменить имеющуюся метку на метку 3 (implicit-null).

Если же необходимо сохранить метку при получении пакета на Egress LER, на предпоследнем маршрутизаторе следует использовать правило ILM с параметром **swap** и заменить имеющуюся метку на метку 0 (Explicit Null). Для Egress LER метка 0 означает, что это последняя метка в LSP, и её следует снять (POP). Такое поведение может быть полезно для получения на Egress LER значений QoS и TTL, которые хранятся в MPLS-заголовке вместе с меткой.

26.1.1 Пример статического MPLS

В данном примере выполняется построение MPLS-сети между четырьмя маршрутизаторами путём статической настройки таблиц FTN (FEC To Next Hop и ILM (Incoming Label Map) для организации сквозных LSP. Маршрутизаторы R1 и R4 выполняют роль LER, присваивая и удаляя метки на границах MPLS-домена, а R2 и R3 функционируют как транзитные LSR, осуществляя коммутацию меток на основе предварительно прописанных правил.

Постройте схему в соответствии с рисунком ниже.

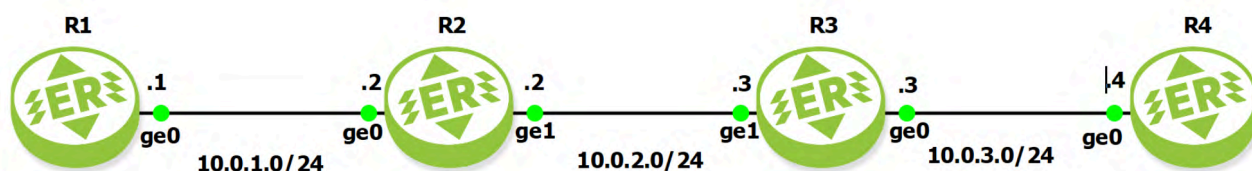


Рисунок 41

Настройте порты и интерфейсы на каждом из маршрутизаторов:

```
enable
configure
hostname R1

interface 2ge0
  label-switching
  ip address 10.0.1.1/24
port ge0
  service-instance 4ge0
  encapsulation untagged
  connect ip interface 2ge0
```

```
enable
configure
hostname R2

interface 2ge0
    label-switching
    ip address 10.0.1.2/24
port ge0
    service-instance 4ge0
        encapsulation untagged
        connect ip interface 2ge0

interface 2ge1
    label-switching
    ip address 10.0.2.2/24
port ge1
    service-instance 4ge1
        encapsulation untagged
        connect ip interface 2ge1
```

```
enable
configure
hostname R3

interface 2ge0
    label-switching
    ip address 10.0.3.3/24
port ge0
    service-instance 4ge0
        encapsulation untagged
        connect ip interface 2ge0

interface 2ge1
    label-switching
    ip address 10.0.2.3/24
port ge1
    service-instance 4ge1
        encapsulation untagged
        connect ip interface 2ge1
```

```
enable
configure
hostname R4
```



```
interface 2ge0
  label-switching
  ip address 10.0.3.4/24
port ge0
  service-instance 4ge0
  encapsulation untagged
  connect ip interface 2ge0
```

На R1 в конфигурационном режиме введите команду: `mpls ftn-entry 10.0.3.0/24 111 10.0.1.2 2ge0`. Команда вносит правило в FTN (FEC To Next hop) — таблицу сопоставления FEC и NextHop. Цель, к которой нужно проложить путь от R1, — это подсеть 10.0.3.0/24; именно этот префикс является FEC. В качестве входного MPLS-маршрутизатор (Ingress LER) R1 добавляет метку 111 и передаёт пакет на следующий узел — на интерфейс 2ge0 и адрес 10.0.1.2.

На R2 в конфигурационном режиме введите команду: `mpls ilm-entry 111 2ge0 swap 222 2ge1 10.0.2.3 10.0.3.0/24`. Команда вносит правило в ILM (Incoming Label Map) — таблицу, устанавливающую действие над пришедшим MPLS-пакетом в зависимости от его метки. На R2, пакет пришедший на интерфейс 2ge0 с меткой 111, перенаправляется на интерфейс 2ge1 и адрес 10.0.2.3 с заменой (SWAP) метки на 222.

На R3 в конфигурационном режиме введите команду: `mpls ilm-entry 222 2ge1 swap 3 2ge0 10.0.3.4 10.0.3.0/24`. Аналогично предыдущему действию, пакет перенаправляется в сторону R4 с меткой 3. Метка 3 имеет специфическое значение неявной нулевой метки (Implicit Null Label) и реализует механизм PHP (Penultimate Hop Popping), снимая метку уже на R3. В результате на R4 придёт уже обычный IP-пакет без MPLS заголовка.

Путь LSP (Label Switching Path) является однонаправленным. Для двусторонней связи необходимо проложить обратный путь.

На R4 в конфигурационном режиме введите команду: `mpls ftn-entry 10.0.1.0/24 333 10.0.3.3 2ge0`. Это полная аналогия правила введённого на R1, но теперь R4 выступает в роли Ingress LSR, а FEC — это адрес подсети 10.0.1.0/24.

На R3 в конфигурационном режиме введите команду: `mpls ilm-entry 333 2ge0 swap 444 2ge1 10.0.2.2 10.0.1.0/24`. R3 меняет пакету метку и отправляет дальше по цепочке на R2.

На R2 в конфигурационном режиме введите команду: `mpls ilm-entry 444 2ge1 swap 0 2ge0 10.0.1.1 10.0.1.0/24`. R2 предпоследний маршрутизатор перед Egress LSR R1. Здесь пакету присваивается специфическая метка 0 (явный ноль, Explicit Null), которая станет сигналом для R1, что метку нужно просто снять (POP) и обрабатывать пакет как обычный IP-пакет. При этом значения QoS и TTL придут на R1 вместе с MPLS-заголовком и будут доступны для обработки. Это можно легко увидеть, если применить анализатор трафика на соединении R1—R2.

На R1 выполните команду `ping 10.0.3.4`. Связь по всей цепочке R1—R2—R3—R4 построена средствами статического MPLS в обоих направлениях, и вы должны получить успешный обмен ICMP-пакетами.

Обратите внимания, что в примере не была настроена никакая, даже статическая, IP-маршрутизация, но при этом обеспечена связность и происходит успешный обмен данными. Это возможно только благодаря ручному вводу правил с указанием адресов для перехода. В любом другом случае маршрутизаторам будет необходимо каким-то образом "узнать" о сети, доступных соседях и адресах их интерфейсов (например с помощью протокола динамической маршрутизации).

26.2 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.

26.2.1 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
```

Включение LDP и функции работы с метками на интерфейсах.

```
ecorouter(config-if)#ldp enable ipv4  
ecorouter(config-if)#label-switching
```

После включения LDP на маршрутизаторе, по команде `show ldp` будет доступен просмотр текущих настроек LDP. Если вы не производили никаких изменений настроек в контекстном режиме `router ldp`, в выводе вы получите настройки по умолчанию.

Просмотр информации о LDP-соседстве.



```
ecorouter#sh mpls ldp neighbor
```

Далее по тексту следуют команды доступные в контекстном режиме `router ldp`.

Router ID — идентификатор LDP-маршрутизатора. Задаётся командой `router-id <A.B.C.D>` в виде IPv4 адреса. Идентификатор должен быть уникальным в пределах MPLS-домена. Если router ID не задан, в качестве идентификатора будет выбран адрес loopback-интерфейса с наибольшим значением. Если loopback-интерфейсы отсутствуют, будет выбран наибольший адрес из всех имеющихся интерфейсов. Хорошим правилом является использование в качестве Router ID интерфейса loopback.0.

Слияния LSP (Label Switched Path):

```
global-merge-capability {merge-capable | non-merge-capable}
```

Слияние путей переходов по меткам (LSP) подразумевает, что идущие через маршрутизатор и предназначенные для одного FEC (Forwarding equivalence class) LSP будут объединены в один маршрут, а все пакеты, принадлежащие этим LSP, будут отправлены далее под одной MPLS-меткой. Именно это поведение является поведением по умолчанию.

В противном случае, если слияние запретить, каждый LSP будет продолжен до пункта назначения FEC, каждый с новой меткой.

Команда `global-merge-capability` перекрывает настройку, установленную по умолчанию на всех интерфейсах заданной настройкой. `merge-capable` — это режим объединения LSP, `non-merge-capable` — это режим без объединения LSP. Режим `non-merge-capable` идентичен режиму по умолчанию.

Сохранение меток Label Preserve.

При изменении у FEC адреса next-hop (адрес следующего маршрутизатора) маршрутизатор генерирует для этого FEC новую метку и сообщает её своим соседям. Для того чтобы маршрутизатор использовал одну и ту же метку для одного FEC при изменении адреса next-hop, необходимо включить опцию `ldp label preserve`.

Метка сохраняется 30 секунд. Поэтому для корректной работы данной опции смена next-hop должна быть произведена за меньшее время.

Установка транспортного адреса маршрутизатора:

```
transport-address ipv4 <ip-address>
```

Переключение режима распространения меток:

```
advertisement-mode {downstream-on-demand | downstream-unsolicited}
```


В режиме `downstream-unsolicited` маршрутизатор будет рассылать метки для каждого известного ему FEC всем своим соседям. В режиме `downstream-on-demand` маршрутизатор будет ожидать запроса метки от своих соседей.

Переключение режима назначения меток:

```
control-mode {independent | ordered}
```

В режиме `independent` маршрутизатор независимо и немедленно назначает и рассылает локальную метку для FEC, как только этот FEC появляется в его таблице маршрутизации. В режиме `ordered` маршрутизатор назначает и рассылает метки для FEC только если он является исходящим (egress) маршрутизатором или он уже получил метку для этого FEC от своего нижестоящего (next-hop) соседа .

Переключение режима удержания меток:

```
label-retention-mode {conservative | liberal}
```

В режиме `conservative` маршрутизатор удерживает метки только лучших маршрутов (т.е. если они получены с допустимого следующего перехода согласно маршрутизации), остальные метки при этом удаляются. В режиме `liberal` маршрутизатор удерживает все метки полученные от других LSR, в том числе метки не лучших маршрутов.

Включение явной нулевой метки (Explicit Null Label):

```
explicit-null
```

В данном режиме предпоследний перед выходным (Egress LSR) маршрутизатор установит метку "0" (ноль) и передаст пакет последнему по пути LSR маршрутизатору. По умолчанию установлен режим неявной нулевой метки (Implicit Null Label), в котором предпоследний маршрутизатор устанавливает метку "3" (три) и передаёт на последний маршрутизатор пакет без метки, реализуя тем самым механизм Penultimate Hop Popping.

Включение функции выявления петель LSP:

```
loop-detection
```

Функция предотвращает заикливание сообщений о запросах меток в присутствии маршрутизаторов LSR, не обеспечивающих слияние LSP. При включении `loop-detection` устанавливаются максимальные значения (255) для команд `loop-detection-hop-count` и `loop-detection-path-vec-count` .

Поддерживаются два метода обнаружения петель: обнаружения количества переходов и обнаружения вектора пути.



- Количество переходов — в процессе построения LSP, в сообщениях LDP будет передаваться количество маршрутизаторов LSR, которые прошло данное сообщение. LSR, обнаруживший, что количество переходов достигло заданного в `loop-detection-hop-count` максимального значения, отреагирует так же, как если бы сообщение было зациклено.
- Вектор пути — в процессе построения LSP в сообщениях LDP будет передаваться список идентификаторов маршрутизаторов LSR, которые прошло данное сообщение. Если маршрутизатор обнаружит свой идентификатор в списке, станет понятно, что сообщение прошло круг и зациклено. Если длина пути достигнет максимальной длины, указанной в `loop-detection-path-vec-count`, маршрутизатор отреагирует так же, как если бы сообщение было зациклено.

Включение функции повторного запроса меток:

```
request-retry
```

Позволяет маршрутизатору совершить пять повторных запросов метки после того, как запрос на метку был отклонён по уважительной причине.

Команда `request-retry-timeout <1-65535>` работает в паре с `request-retry` и устанавливает интервал в секундах между повторными запросами метки.

Включение функции распространения сообщений об освобождении меток

```
propagate-release
```

Функция подразумевает, что уведомление об освобождении (release) метки пересылается (propagate) дальше по цепочке LSP от upstream-роутеров к downstream-роутерам. Маршрутизатор получивший сообщение об освобождении метки от вышестоящего LSR, не просто освобождает метку, но и передаёт сообщение об освобождении метки далее нижестоящему LSR.

Функция бережного перезапуска плоскости управления LDP:

```
graceful-restart enable
```

Благодаря этой функции маршрутизатор может перезагрузить LDP соседство без разрыва идущего MPLS-трафика и отношений сопряжения.

Механизм бережного перезапуска работает в связке между двумя роутерами:

- Restarter (Тот, кто перезагружается): Маршрутизатор, который выполняет перезагрузку.

- Helper (Тот, кто помогает): Соседний маршрутизатор, который знает о перезагрузке и помогает сохранить трафик и восстановить сессию.

Чтобы маршрутизатор работал именно в режиме Helper, нужно дать команду `graceful-restart helper-mode`.

Для функций бережного перезапуска в режиме Helper существуют два таймера:

- `graceful-restart timers neighbor-liveness <5-300>` — время в секундах, которое маршрутизатор будет ожидать перезапуска и восстановления подключения от своего соседа. Значение времени по умолчанию составляет 120 секунд.
- `graceful-restart timers max-recovery <15-600>` — время в секундах, которое маршрутизатор будет ожидать полной повторной синхронизации. Значение времени по умолчанию составляет 120 секунд. За указанное время Restarter должен повторно запросить и получить все метки от Helper'a.

Команда `graceful-restart disable` отменит любую из двух команд: `graceful-restart enable` или `graceful-restart helper-mode`.

Для удаления таймеров из конфигурации нужно использовать команды `no graceful-restart timers neighbor-liveness` и `no graceful-restart timers max-recovery` соответственно.

Отключение функции рассылки многоадресных приветственных сообщений:

```
no multicast-hellos
```

По умолчанию функция включена на всех интерфейсах, поддерживающих LDP. Многоадресная рассылка приветственных (Hello) сообщений используется для автоматического обнаружения соседних LDP-маршрутизаторов в напрямую подключённых сетях.

Установка интервала рассылки приветственных сообщений:

```
hello-interval <1-21845>
```

Команда задаёт частоту с которой маршрутизатор будет отправлять Hello сообщения своим соседям. Сообщение рассылается в виде UDP-дейтаграмм во все интерфейсы маршрутизатора где активирован LDP. Рассылка ведётся на мультикастовый адрес 224.0.0.2 и порт 646.

Заданное глобальное (в пределах маршрутизатора) значение `hello-interval` может быть переопределено на каждом интерфейсе командой `ldp hello-interval <1-21845>`.



Для достижения оптимальной производительности устанавливайте значение `hello-interval` не более чем в одну треть от значения времени удержания (`hold-time`).

Установка времени удержания до разрыва смежности:

```
hold-time <3-65535>
```

Время удержания определяет время, в течение которого LSR будет ожидать следующего приветственного сообщения, прежде чем сочтёт его неработоспособным и разорвёт с ним отношения смежности.

Пара LSR согласовывает время удержания, используемое для приветствий друг от друга. Каждый предлагает своё значение времени удержания, и LSR использует меньшее из двух значений.

Заданное глобальное (в пределах маршрутизатора) значение `hold-time` может быть переопределено на каждом интерфейсе командой `ldp hold-time <3-65535>`.

Для оптимальной производительности установите это значение не менее чем в три раза больше указанного интервала приветствий (`hello-interval`).

Targeted-peer в LDP — это механизм установления LDP-сессии с удалённым маршрутизатором, который не является непосредственным соседом на канальном уровне (т.е. между ними есть другие маршрутизаторы).

Для установки интервала отправки юникаст-hello-сообщений для всех целевых узлов необходимо использовать команду `targeted-peer-hello-interval <1-21845>` в режиме конфигурации `router ldp`.

Для установки времени удержания приветственных сообщений для всех целевых равноправных узлов необходимо использовать команду `targeted-peer-hold-time <3-65535>` в режиме конфигурации `router ldp`.

Установка аутентификации на соседские отношения:

```
neighbor {A.B.C.D | X:X::X:X} auth md5 password <0-7> PASS
```

Где: **A.B.C.D** — IPv4 адрес соседа; **X:X::X:X** — IPv6 адрес соседа; **<0-7>** — тип пароля; **PASS** — пароль. Идентичные тип пароля и пароль должны быть установлены на обоих соседних маршрутизаторах.

Аутентификация запрещает установление соседских отношений без пароля.

Переход в режим настройки и активация целевого равноправного узла:

```
targeted-peer {ipv4 | ipv6} <ADDRESS>
```

На маршрутизаторе LSR может быть настроено несколько целевых узлов.

В режиме настройки целевого узла доступны настройки `hello-interval` и `hold-time`, аналогичные по своему действию глобальным настройкам этих же параметров, но применённые к конкретному целевому узлу.

Применение команд настройки целевых узлов (`targeted-peer`) в различных сценариях будут разобраны в следующих подразделах этого раздела.

Установка интервала отправки сообщений подтверждения работоспособности:

```
keepalive-interval <10-21845>
```

Значение по умолчанию равно 10 секундам. Keepalive сообщения служат подтверждением сохранности TCP сессии между соседними маршрутизаторами.

Нет необходимости проверять TCP-соединение также часто, как искать соседей в broadcast-сегменте, поэтому рекомендуется устанавливать интервал рассылки Keepalive в разы больше чем интервал рассылки Hello сообщений.

Данное глобальное (в пределах маршрутизатора) значение перекрывается значением заданным на интерфейсе командой `ldp keepalive-interval <10-21845>`.

Установка интервала ожидания сообщения подтверждающего работоспособность:

```
keepalive-timeout <30-65535>
```

Значение по умолчанию равно 30 секундам.

Маршрутизатор в течение указанного времени ожидает от соседнего маршрутизатора PDU или Keepalive сообщение. По истечении указанного времени, если не было получено ни одного LDP-пакета (ни Keepalive, ни Label Mapping, ни любого другого), TCP-соединение считается разорванным, и LDP-сессия прекращается.

Обычно значение `keepalive-timeout` должно втрое превосходить значение `keepalive-interval`.

Данное глобальное (в пределах маршрутизатора) значение перекрывается значением заданным на интерфейсе командой `ldp keepalive-timeout <30-65535>`.

Включение опции оптимизации LDP

```
ldp-optimization
```

Опция производит два действия:

- Сбрасывает таймер ожидания сообщений Keepalive при получении приветственного сообщения.

- Сбрасывает таймер удержания смежности при получении любого управляющего сообщения LDP.

Такой режим работы может повысить стабильность и отказоустойчивость LDP-сессий в крупных и нестабильных сетях.

Advertise-labels. По умолчанию метки всех FEC объявляются всем LDP-соседям. С помощью команды контекстного режима `router ldp advertise-labels` можно гибко ограничивать рассылку меток LDP-соседям.

```
advertise-labels for any to none
advertise-labels for PREFIX_ACL to (PEER_ACL|any)
```

Первая команда означает: "Все метки для всех адресатов не отправлять никому." Другими словами — никакие метки не будут отправлены.

В случае второй команды, из префиксного списка **PREFIX_ACL** будут взяты подсети, чьи метки будут рассылаться, а из префиксного списка **PEER_ACL** соседние LSR-маршрутизаторы которым будет осуществляться рассылка. Параметр **any** заставит маршрутизатор рассылать отобранные метки всем доступным LSR.

Включение функции импорта BGP-маршрутов:

```
import-bgp-routes
```

По умолчанию маршруты BGP не импортируются в LDP. Импорт BGP-маршрутов в LDP необходим для того, чтобы расширить механизм распределения меток за пределы IGP-домена и обеспечить MPLS-транспортировку трафика до next-hop'ов, о которых известно только из BGP.

Нестрогий режим отзыва меток:

```
ldp withdraw-release non-strict
```

Режим позволяет принимать запросы на отзыв (withdraw) и ответы об удалении меток (release) с такими дополнительными параметрами, как: VC_FEC_OPT_IF_MTU, VC_FEC_OPT_IF_VLAN, VC_FEC_OPT_IF_VCCV. В противном случае LDP сессия разрывается.

26.2.2 Команды просмотра

Для просмотра конфигурации и статуса протокола LDP используются команды, представленные в таблице ниже.

Таблица 91 — Команды просмотра конфигурации и статуса протокола LDP



Команда	Описание
<code>show ldp adjacency</code>	Список LDP-связности
<code>show ldp advertise-labels</code>	Просмотр информации о метках
<code>show ldp downstream</code>	Просмотр распространения меток по методу downstream
<code>show ldp upstream</code>	Просмотр распространения меток по методу upstream
<code>show ldp fec</code>	Информация о Forwarding Equivalence Class
<code>show ldp fec-ipv4</code>	Информация о Forwarding Equivalence Class
<code>show ldp graceful-restart</code>	Статус механизма Graceful Restart
<code>show ldp igp</code>	Параметры IG
<code>show ldp interface</code>	Статус интерфейсов с функцией LDP
<code>show ldp lsp</code>	Просмотр пути прохождения пакета на основе протоколов LDP
<code>show ldp mpls-l2-circuit</code>	Просмотр конфигурации l2-circuit
<code>show ldp ms-pw</code>	Multi-Segment PW information
<code>show ldp routes</code>	Таблица NSM маршрутов LDP
<code>show ldp session</code>	Информация о сессии LDP
<code>show ldp statistics</code>	Просмотр статистики LDP
<code>show ldp targeted-peer</code>	Информация о пограничном MPLS маршрутизаторе
<code>show ldp targeted-peers</code>	List of targeted peers defined

26.2.3 LDP и IGP синхронизация

Функция синхронизации LDP и IGP обеспечивает полную конвергенцию LDP до того, как путь IGP будет использован для коммутации.

LDP отвечает за распределение меток MPLS, а IGP — за маршрутизацию IP-трафика. Если маршрут в IGP появился, но LDP ещё не установил LSP (Label Switched Path), трафик будет отбрасываться (так как нет метки для передачи). Например, после сбоя, IGP может восстановить маршрут быстрее, чем LDP, и пакеты будут уходить в никуда.

В случае если при изменении топологии, LDP должен перестроить LSP в соответствии с новыми IGP-маршрутами и если LDP и IGP работают не синхронно,

возможна ситуация, когда маршруты в IGP уже изменились, а LSP ещё нет. Это может вызвать петли маршрутизации.

При разрыве соединения, IGP быстро пересчитает маршруты, но если LDP будет запаздывать, данные будут теряться. LSP в MPLS должны перестраиваться синхронно с IGP, чтобы не было задержек в передаче трафика в чём и помогает синхронизация.

MPLS LDP IGP синхронизация выполняет следующие функции:

- Позволяет синхронизировать LDP и IGP для минимизации потери пакетов MPLS.
- Позволяет включить синхронизацию LDP IGP на любом интерфейсе, связанном с процессом IGP Open Shortest Path First (OSPF) или Intermediate System-to-System (IS-IS).
- Предотвращает потерю пакетов MPLS из-за конфликтов синхронизации.

Когда на интерфейсе установлено соседство IGP, но синхронизация LDP IGP ещё не достигнута или потеряна, IGP объявляет максимальную метрику на этом интерфейсе. Когда хотя бы одна сессия LDP через интерфейс сходится, IGP объявляет обычную метрику для интерфейса.

Все команды связанные с синхронизацией вводятся в режиме конфигурирования интерфейса.

Для включения на интерфейсе синхронизации с протоколом IS-IS воспользуйтесь командой: `mpls ldp-igp sync isis {level-1|level-1-2|level-2} [holddown-timer <1-2147483>]`.

Указание уровня IS-IS (Level-1, Level-1-2, Level-2) влияет на то, в каких зонах будет распространяться информация о метках и как LDP будет взаимодействовать с IS-IS. LDP должен знать, в каких уровнях искать маршруты для распределения меток.

Если синхронизировать только Level-1, LDP будет учитывать только маршруты внутри зоны (Area). Маршруты из других зон (через Level-2) не будут иметь меток — MPLS-трафик может быть отброшен на границе.

Если синхронизировать только Level-2, LDP будет учитывать только магистральные маршруты (Backbone). Маршруты внутри зон (Level-1) не будут иметь меток — могут возникнуть проблемы с MPLS внутри зоны.

Если синхронизировать Level-1-2 LDP будет учитывать и Level-1, и Level-2 маршруты. Это наиболее правильный вариант, если маршрутизатор работает на границе level-1 и level-2.

Для включения на интерфейсе синхронизации с протоколом OSPF воспользуйтесь командой: `mpls ldp-igp sync ospf [holddown-timer <1-2147483>]`

При синхронизации с обоими (OSPF и IS-IS) протоколами можно использовать таймер удержания `holddown-timer` в диапазоне от одной до 2147483 секунд. В случае



сбоя в сети и потере LDP-сессий, через время указанное в таймере удержания, IGP протокол перестанет ожидать от LDP информацию об успешном обмене MPLS-метками и восстановит значение метрик для маршрутов.

Для вывода данных по синхронизации для всех интерфейсов воспользуйтесь командой `show mpls ldp igp sync` в административном режиме.

Таймер задержки MPLS LDP IGP синхронизации

Таймер задержки MPLS LDP IGP позволяет настроить время задержки перед синхронизацией MPLS LDP и IGP. Обычно, когда настроена синхронизация LDP IGP, LDP уведомляет IGP, как только LDP синхронизируется. При настройке таймера задержки это уведомление задерживается на указанный период времени.

Команда для настройки таймера задержки вводится в режиме конфигурирования интерфейса и выглядит следующим образом: `mpls ldp-igp sync-delay <5-60>`, где `<5-60>` — время задержки в секундах.

Когда LDP полностью установлен и синхронизирован:

- Если таймер задержки настроен, LDP запускает таймер. Когда таймер истекает, LDP проверяет, что синхронизация все ещё действительна, и уведомляет процесс OSPF или IS-IS.
- Если таймер задержки не настроен, или если синхронизация отключена или не работает, или если интерфейс был удалён из процесса IGP, LDP немедленно уведомляет процесс OSPF или IS-IS.

Если вы настраиваете новое время задержки во время работы таймера, LDP сохраняет новое время задержки, но не перенастраивает работающий таймер.

26.3 Virtual Private Wire Service

VPWS (Virtual Private Wire Service) — это служба L2VPN типа "точка-точка", в основе которой лежит концепция PW (PseudoWire) — псевдопровода. Псевдопровод соединяет две клиентские сети, прозрачно передавая кадры канального уровня (Ethernet, VLAN, HDLC, PPP) через сеть провайдера с использованием MPLS или других туннельных технологий.

26.3.1 Настройка VPWS

Базовая настройка VPWS 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)#ldp enable 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 псевдопровод (PseudoWire).

```
ecorouter(config-router)#targeted-peer ipv4 <ip-address> .
```

Включение ldp и функции работы с метками на интерфейсах.

```
ecorouter(config-if)#ldp enable ipv4  
ecorouter(config-if)#label-switching
```


Псевдопровод конфигурируется в зависимости от типа создаваемой схемы.

Создание Pseudowire type 5: Ethernet Raw Mode.

```
l2vpn-vpws <name>
signaling ldp
neighbor <ip-address>
vc-id <Identifying value>
exit-vpws-neighbor
no shutdown
```

Где в качестве `<name>` задаётся идентификационное имя соединения, `<Identifying value>` — идентификационный номер псевдопровода (должен быть идентичен на обоих концах), `<ip-address>` — адрес граничного маршрутизатора на противоположном конце псевдопровода.

Создание l2-circuit type 4: Ethernet Tagged Mode.

```
l2vpn-vpws <name>
signaling ldp
neighbor <ip-address>
vc-id <Identifying value>
exit-vpws-neighbor
mode tagged svlan <SVLAN-ID> tpid [8100 | 88a8 | 9100]
no shutdown
```

Где в качестве `<name>` задаётся идентификационное имя соединения, `<Identifying value>` — идентификационный номер псевдопровода (должен быть идентичен на обоих концах), `<ip-address>` — адрес граничного маршрутизатора на противоположном конце псевдопровода, `<SVLAN-ID>` — номер виртуальной сети, значения при `tpid` (Tag Protocol Identifier):

- 8100 (стандартное значение) — указывает на наличие VLAN тега;
- 0x88a8 — указывает на наличие второго тега Q-in-Q, Service VLAN;
- 0x9100 — некоторые проприетарные реализации Q-in-Q.

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 псевдопровода (4 или 5) указывается тегированный или нетегированный трафик, параметр **name** — имя ранее созданного псевдопровода.

Просмотр состояния l2-circuit. Где **name** — имя ранее созданного псевдопровода.
`ecorouter#show mpls l2-circuit <name>`

Гибкая настройка различных операций с VLAN-тегами на service-instance позволяет передавать пакет через псевдопровод, предварительно проделав эти операции с VLAN-тегами. При этом используется тип инкапсуляции 5 (Ethernet Tagged Mode).

Поддерживаются следующие операции:

Снять обе метки с пакета перед отправкой в MPLS-туннель:

```
l2vpn-vpws pop_pop
signaling ldp
neighbor 2.2.2.2
vc-id 20
exit-vpws-neighbor
mode raw
no shutdown

port te1
service-instance pop_pop
encapsulation dot1q 40 second-dot1q 90
rewrite pop 2
mpls-l2-circuit pop_pop primary
```

Снять внешнюю метку с пакета с двумя метками, перед отправкой в MPLS-туннель:

```
l2vpn-vpws pop_sv
signaling ldp
neighbor 2.2.2.2
vc-id 30
exit-vpws-neighbor
mode tagged svlan 77 tpid 8100
no shutdown

port te1
service-instance pop_sv
encapsulation dot1q 40 second-dot1q 77
rewrite pop 1
mpls-l2-circuit pop_sv_any_cv primary
```

Снять внешнюю метку и заменить внутреннюю на произвольную перед отправкой в MPLS-туннель:

```
l2vpn-vpws pop_swap
signaling ldp
```



```
neighbor 2.2.2.2
  vc-id 40
  exit-vpws-neighbor
mode tagged svlan 77 tpid 8100
no shutdown

port te1
  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:

```
l2vpn-vpws push_sv
  signaling ldp
  neighbor 2.2.2.2
  vc-id 50
  exit-vpws-neighbor
mode tagged svlan 77 tpid 88a8
no shutdown

port te1
  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:

```
l2vpn-vpws push_two
  signaling ldp
  neighbor 2.2.2.2
  vc-id 60
  exit-vpws-neighbor
mode tagged svlan 77 tpid 88a8
no shutdown

port te1
  service-instance push_two
  encapsulation untagged
  rewrite push 77 88
  mpls-l2-circuit push_two primary
```

Заменить внешнюю метку перед отправкой в MPLS-туннель:



```
l2vpn-vpws swap_sv
signaling ldp
neighbor 2.2.2.2
vc-id 70
exit-vpws-neighbor
mode tagged svlan 77 tpid 88a8
no shutdown

port te1
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:

```
l2vpn-vpws swap_swap
signaling ldp
neighbor 2.2.2.2
vc-id 80
exit-vpws-neighbor
mode tagged svlan 77 tpid 88a8
no shutdown

port te1
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:

```
l2vpn-vpws swap_swap
signaling ldp
neighbor 2.2.2.2
vc-id 90
exit-vpws-neighbor
mode tagged svlan 77 tpid 88a8
no shutdown

port te1
service-instance swap_push
encapsulation dot1q 60 exact
```

```
rewrite translate 1-to-2 77 88
mpls-l2-circuit swap_push primary
```

26.3.1.1 Пример настройки VPWS

Постройте схему в соответствии с рисунком ниже:

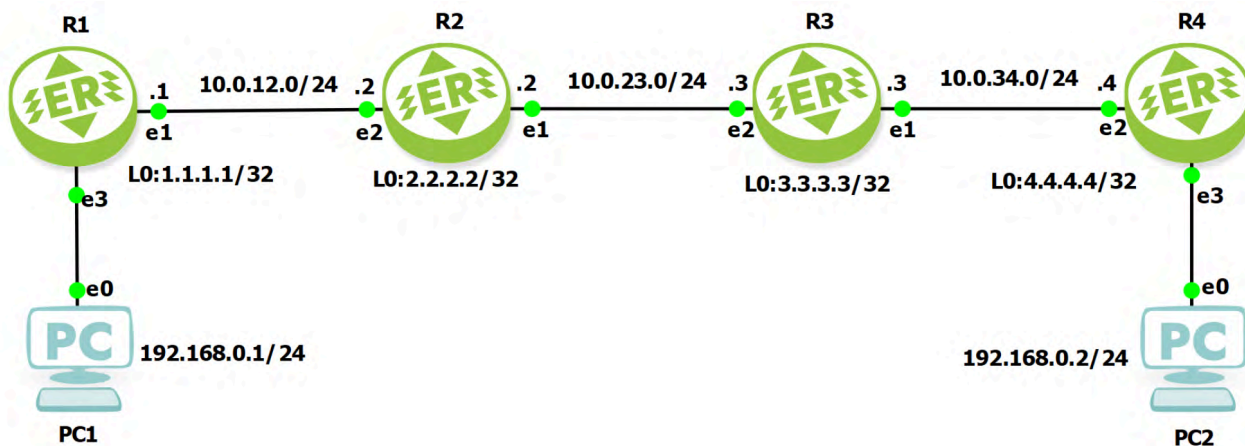


Рисунок 42

На R1 настройте маршрутизацию, порты и интерфейсы:

```
enable
configure
hostname R1

router isis 1
 net 10.0000.0000.0001.00

router ldp
 transport-address ipv4 1.1.1.1

interface loopback.0
 ip address 1.1.1.1/32
 ip router isis 1

interface 2ge1
 ip address 10.0.12.1/24
 ip router isis 1
 label-switching
 ldp enable ipv4

port ge1
 service-instance 4ge1
 encapsulation untagged
```

```
connect ip interface 2ge1
exit
```

По аналогии с R1 настройте R2, R3 и R4. Используйте данные со схемы и следующие сетевые NSAP адреса для протокола ISIS:

- R2: 10.0000.0000.0002.00
- R3: 10.0000.0000.0003.00
- R4: 10.0000.0000.0004.00

Обратите внимание, что на каждом интерфейсе каждого маршрутизатора включён режим коммутации по меткам (`label-switching`) и протокол LDP (`ldp enable ipv4`), а также подключена маршрутизация ISIS (`ip router isis 1`).

На PC1 и PC2 зеркально настройте IP-адрес и шлюз по умолчанию:

- PC1> ip 192.168.0.1/24 192.168.0.2
- PC2> ip 192.168.0.2/24 192.168.0.1

После произведённых настроек между маршрутизаторами появится IP-связность и коммутация по меткам (в пакеты добавлен заголовок MPLS — назначены метки).

На R1 настройте L2VPN VPWS (Virtual Private Wire Service):

```
l2vpn-vpws CIR
signaling ldp
neighbor 4.4.4.4
vc-id 127
exit-vpws-neighbor
no shutdown
```

Настройте удалённое целевое соседство:

```
router ldp
targeted-peer ipv4 4.4.4.4
```

Подключите сервис L2VPN VPWS к сервисному интерфейсу:

```
port ge3
service-instance 4ge3
encapsulation untagged
l2vpn-vpws CIR primary
```

Аналогично R1 настройте R4. Вместо адреса целевого соседа 4.4.4.4 используйте адрес 1.1.1.1.

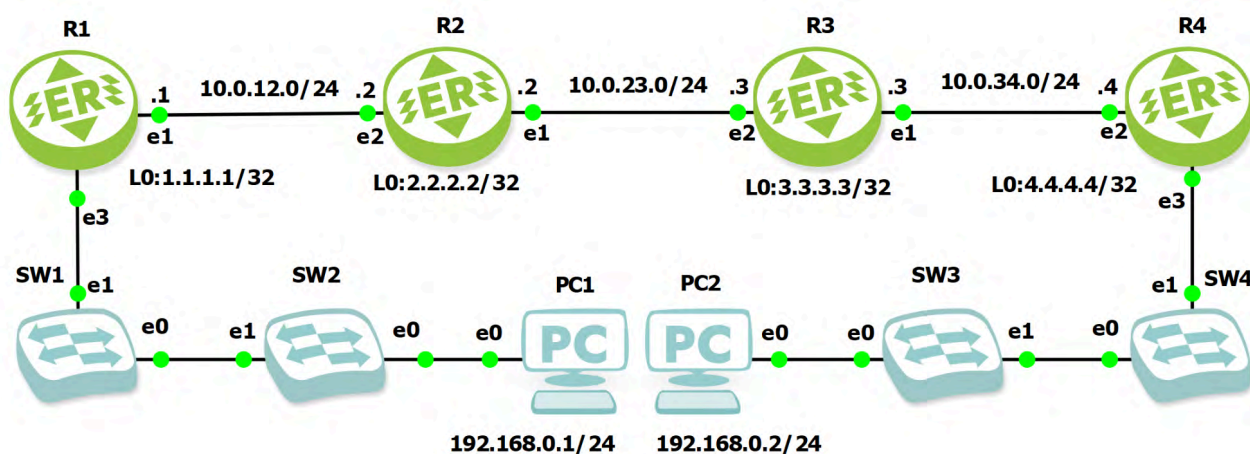
Проверьте на PC1 пинг в сторону PC2: `ping PC1 ping 192.168.0.2`.



Обратите внимание что для соединения R1 и R2 с PC1 и PC2 не настроен IP-адрес и интерфейс. Связность происходит на L2-уровне. PC1 и PC2 взаимодействуют как два напрямую соединённых устройства.

В программе перехвата трафика вы можете увидеть, как к изначальному Ethernet-кадру добавлены заголовки MPLS, и далее кадр снова инкапсулирован в Ethernet-кадр для передачи по условной сети провайдера.

Дополните схему в соответствии с рисунком ниже:



> [!picture] Рисунок

Настройте коммутаторы SW1, SW2, SW3, SW4 так, чтобы каждый порт e0 каждого коммутатора имел тип access и соответствующий тег VLAN (100 для SW2 и SW3; 200 для SW1, SW4), а порт e1 имел тип dot1q и тег VLAN 1. Таким образом кадры на выходе из SW1 и SW4 получат внутреннюю метку 100 и внешнюю 200.

Переконфигурируйте R1

```
# Настройки на порте
port ge3
service-instance 4ge3
# Временное отключение псевдопровод от порта
no l2vpn-vpws CIR primary
# Установка входящей инкапсуляции
encapsulation dot1q 200 second-dot1q 100
# Настройка операцию с метками
rewrite translate 2-to-1 77
exit

# Настройка псевдопровода тип 4 Tagged Mode
l2vpn-vpws CIR
mode tagged svlan 77 tpid 8100

# Повторное подключение псевдопроводу к порту
port ge3
```



```
service-instance 4ge3  
l2vpn-vpws CIR primary
```

Аналогичным образом настройте R4.

Напомним, что правила тегирования на портах работают зеркально в зависимости от того приходит ли кадр извне через порт на интерфейс или уходит через интерфейс на порт наружу.

На R1 проверьте ping в сторону R4: `ping 192.168.0.2`.

В программе перехвата трафика вы можете проверить на различных подключениях наличие и значения тегов VLAN.

26.3.2 Резервный псевдопровод

Резервный псевдопровод (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 удалённых граничных маршрутизаторов.

Для аварийного переключения на резервный псевдопровод 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
```

На каждом маршрутизаторе, входящем в сеть MPLS, включить режим передачи сообщений (PW Status TLV) о состоянии псевдопровода.

```
ecorouter(config-router)#pw-status-tlv
```

Включить LDP и коммутацию по меткам на интерфейсе в сторону MPLS сети.

```
ecorouter(config-if)#ldp enable ipv4
```




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

Далее необходимо настроить основной L2 туннель. Например, создать псевдопровод type 5 с именем vc1, Identifying value — 1111.

Для этого нужно создать псевдопровод type 5.

```
mpls l2-circuit vc1 1111 2.2.2.2
```

Создать резервный псевдопровод, с именем vc2, Identifying value — 2222.

```
mpls l2-circuit vc2 2222 2.2.2.2
```

Привязать оба псевдопровода к порту ge2.

Установить реверсивный режим, что позволит вернуться к основному псевдопроводу после его восстановления.

```
port ge2
service-instance ge2/e2
encapsulation untagged
l2vpn-vpws vc1 primary
l2vpn-vpws vc2 secondary
vc-mode revertive
```

26.3.2.1 Пример резервирования псевдопровода

Постройте схему в соответствии с рисунком ниже:

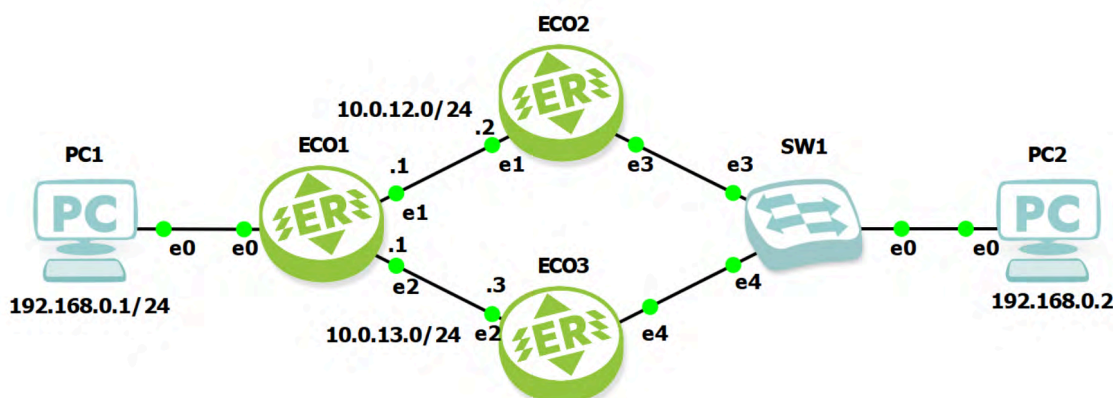


Рисунок 43

На PC1 и PC2 зеркально настройте IP-адрес и шлюз по умолчанию:

- PC1> ip 192.168.0.1/24 192.168.0.2
- PC2> ip 192.168.0.2/24 192.168.0.1

Настройте ECO1:

```
enable
configure
```

```
hostname EC01

router ldp
  transport-address ipv4 1.1.1.1
  targeted-peer ipv4 2.2.2.2
  exit
  targeted-peer ipv4 3.3.3.3
  exit

router ospf 111
  ospf router-id 1.1.1.1
  network 1.1.1.1 0.0.0.0 area 0
  network 10.0.12.1 0.0.0.0 area 0
  network 10.0.13.1 0.0.0.0 area 0

l2vpn-vpws CIR1
  signaling ldp
  neighbor 2.2.2.2
    vc-id 127
  exit-vpws-neighbor
  no shutdown

l2vpn-vpws RES1
  signaling ldp
  neighbor 3.3.3.3
    vc-id 128
  exit-vpws-neighbor
  no shutdown

interface loopback.0
  ip address 1.1.1.1/32

interface 2ge1
  ip address 10.0.12.1/24
  label-switching
  ldp enable ipv4

interface 2ge2
  ip address 10.0.13.1/24
  label-switching
  ldp enable ipv4

port ge1
```



```
service-instance 4ge1
  encapsulation untagged
  connect ip interface 2ge1

port ge2
  service-instance 4ge2
  encapsulation untagged
  connect ip interface 2ge2

port ge0
  service-instance 4ge0
  encapsulation untagged
  l2vpn-vpws CIR1 primary
  l2vpn-vpws RES1 secondary
  vc-mode revertive
```

Настройте ECO2 и ECO3 аналогичным образом. Учитывайте следующие моменты:

- У ECO2 и ECO3 на интерфейсах loopback.0 должны быть заданы IP-адреса 2.2.2.2 и 3.3.3.3 соответственно.
- В качестве `targeted-peer` для ECO2 и ECO3 должен выступать ECO1 с адресом 1.1.1.1.
- На ECO2 и ECO3 должно быть настроено по одному псевдопроводу.
- Значение `vc-id` в настройках псевдопровода на ECO2 и ECO3 должны соответствовать заданным на ECO1.
- К портам ge1 на ECO2 и ge2 на ECO3 должно быть подключено по одному псевдопроводу.
- В настройках порта на ECO2 и ECO3 не нужно включать реверсивный режим `vc-mode revertive`.

Проверьте ping между PC1 и PC2.

Программой перехвата трафика проверьте соединения ECO2—SW1 и ECO3—SW1. Трафик должен передаваться через ECO2—SW1, что соответствует первичному псевдопроводу `l2vpn-vpws CIR1 primary`.

Разорвите соединение ECO1—ECO3 и снова проверьте ping между PC1 и PC2.

Программой перехвата трафика проверьте соединения ECO2—SW1 и ECO3—SW1. Трафик должен передаваться через ECO3—SW1, что соответствует вторичному псевдопроводу `l2vpn-vpws RES1 secondary`.

Восстановите соединение ECO1—ECO3. Трафик должен вернуться в первичный псевдопровод.

26.4 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 передаёт 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.

26.4.1 Топология

Приведённая ниже схема реализует классический сценарий совместной работы протоколов BGP и MPLS, который явно демонстрирует все плюсы коммутации по меткам.

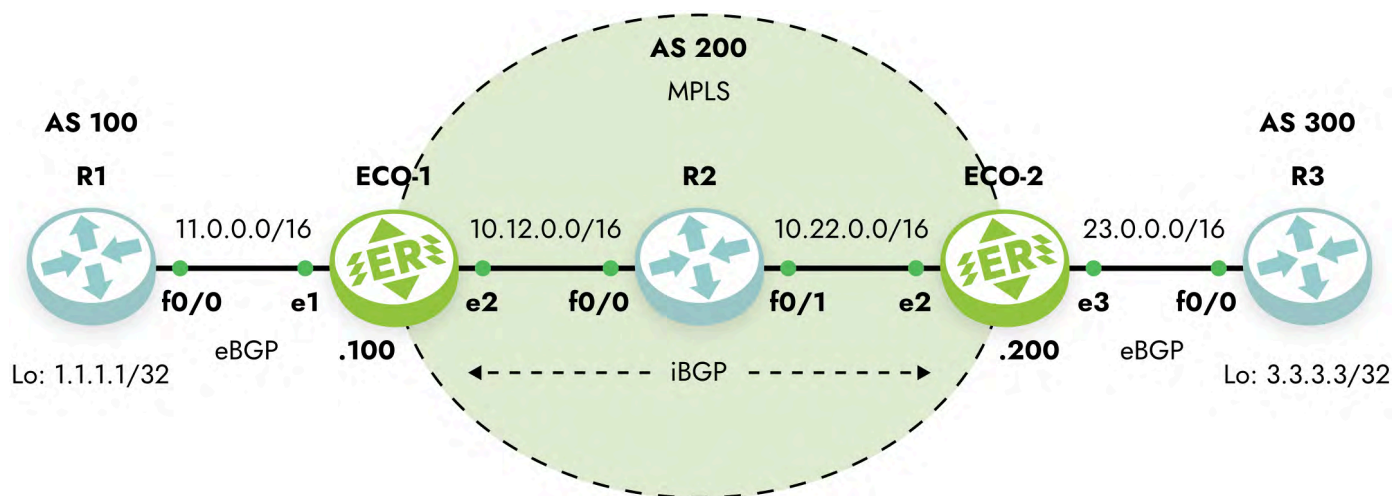


Рисунок 44

На схеме маршрутизаторы ECO-1, ECO-2 и R2 находятся в MPLS-облаке, и между ECO-1 и ECO-2 настроен iBGP. Маршрутизаторы R1 и R3 подключены к MPLS-облаку через eBGP. Локальные сети маршрутизаторов R1 и R3 представлены в виде loopback-интерфейсов. Необходимо создать связность между локальными сетями маршрутизаторов R1 и R3.

26.4.2 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  
service-instance te0/FastEthernet0/0  
  encapsulation untagged  
  
interface loopback.0  
  ip address 1.1.1.1/32  
  
interface FastEthernet0/0  
  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 te1
lacp-priority 32767
mtu 9728
service-instance te1/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 te1 service-instance te1/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
```

26.4.3 MPLS 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.

26.5 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.



26.5.1 MPLS VPN терминология

На рисунке ниже показана сеть оператора Connector с частными виртуальными сетями клиентов ComA и ComB.

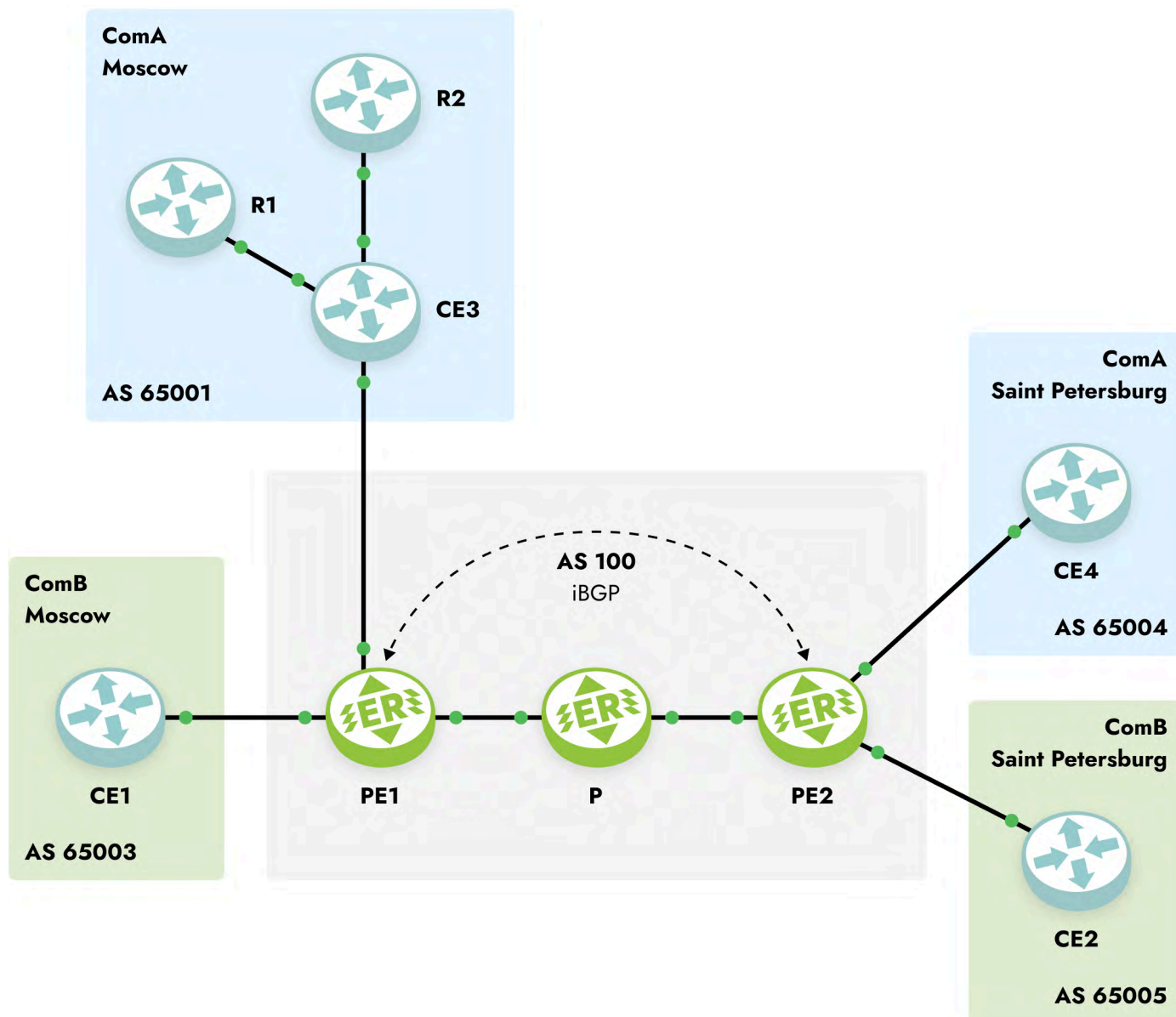


Рисунок 45

Пограничное устройство клиента (Customer Edge Router, CE) — маршрутизатор на стороне клиента, который присоединён 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 подключён 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 подключённые напрямую к сети оператора. На рисунке выше R1 и R2 — это клиентские маршрутизаторы.

26.5.2 Процесс маршрутизации сетей VPN

Процесс маршрутизации MPLS-VPN включает следующие этапы:

- Оператор предоставляет услугу VPN через PE-маршрутизаторы, которые подсоединены напрямую к клиентским CE-маршрутизаторам по Ethernet.
- 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.
- 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.
- В качестве BGP next-hop для VPNv4 маршрутов, а также в качестве LDP Router ID для распределения транспортных меток, используется уникальный идентификатор маршрутизатора (как правило, его loopback-адрес)
- 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.
- PE-маршрутизаторы на входе (ingress) прикрепляют к VPN-пакету, отправляющемуся по сети оператора, стек из двух меток. В него входят: сервисная метка — BGP-метка, определённая по таблице маршрутизации

(ассоциированной с входящим интерфейсом), которая указывает на конкретный VRF и выходной интерфейс на выходном (egress) PE; транспортная метка — LDP-метка из глобальной FTN таблицы, определяющая IP next hop.

- Операторский (P) маршрутизатор в сети пересылает 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 P 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. Если в LDP включена опция **explicit-null**, предпоследний маршрутизатор отправляет пакет с двумя метками, где значение верхней метки — 0.
- Выходной PE-маршрутизатор снимает BGP-метку, производит поиск по ней на исходящих интерфейсах и отправляет пакет соответствующему клиентскому CE-маршрутизатору.

26.5.3 Конфигурирование MPLS Layer-3 VPN

Для настройки MPLS L3 VPN требуется развёртывание нескольких взаимосвязанных компонентов. Основой служит транспортная MPLS-сеть, где на всех интерфейсах маршрутизаторов провайдера (P и PE) активируется коммутация по меткам и протокол LDP для их распределения. Критически важным является обеспечение IP-связности через IGP (OSPF или IS-IS), который должен распространять маршруты ко всем интерфейсам, включая loopback-адреса, используемые в качестве LDP Router ID и BGP next-hop.

Ключевым элементом изоляции клиентского трафика выступают VRF, создаваемые на каждом PE-маршрутизаторе. Для каждого клиента или группы виртуальных сетей настраивается отдельная VRF-таблица маршрутизации с уникальным Route Distinguisher, обеспечивающим уникальность IPv4-префиксов преобразуя их в уникальные VPNv4-адреса. Атрибут Route Target определяет политику импорта и экспорта маршрутов между VRF на разных PE, формируя требуемую топологию VPN.

Обмен VPN-маршрутами между граничными маршрутизаторами осуществляется через MP-BGP с активированным адресным семейством VPNv4. iBGP-сессии между PE



устанавливаются непосредственно или через Route Reflector, с обязательной настройкой extended communities для передачи RT атрибутов. При этом BGP next-hop сохраняется как loopback-адрес исходного PE, а маршруты переносятся вместе с VPN-метками, которые будут использоваться для идентификации VRF на принимающем PE.

Подключение клиентских устройств (CE) осуществляется через интерфейсы, привязанные к соответствующим VRF. Маршрутизация между PE и CE может настраиваться через статические маршруты, eBGP, OSPF или EIGRP, при этом протоколы динамической маршрутизации должны работать в VRF-контексте. Для завершения настройки необходимо активировать адресное семейство IPv4 unicast в BGP для каждого VRF, обеспечивая редистрибуцию клиентских маршрутов в MP-BGP и обратно.

Верификация работоспособности включает проверку LDP-соседств, установления BGP-сессий VPNv4, наличия VPN-маршрутов в BGP-таблице и их корректной инсталляции в VRF-таблицы маршрутизации. Дополнительно контролируется MPLS-форвардинг через show mpls forwarding и проверяется сквозная связность между клиентскими узлами с использованием ping vrf.

Процесс конфигурирования MPLS Layer-3 VPN состоит из следующих этапов:

- **Настройка портов и интерфейсов:**
Настройка физических интерфейсов и loopback-адресов на всех маршрутизаторах сети.
- **Развёртывание IGP (OSPF/IS-IS):**
Обеспечение полной связности всех интерфейсов, включая интерфейсы loopback, внутри MPLS-сети.
- **Активация MPLS транспорта:**
Включение коммутации по меткам (label switching) и протокола LDP на всех интерфейсах MPLS-сети, исключая интерфейсы loopback.
- **Настройка MP-BGP между PE-маршрутизаторами:**
Создание полносвязной iBGP-сети между всеми граничными маршрутизаторами (непосредственно или через Route Reflector) с активацией адресного семейства VPNv4.
- **Создание и настройка VRF:**
Определение виртуальных таблиц маршрутизации для каждого клиента с назначением Route Distinguisher (RD) и Route Target (RT).
- **Привязка клиентских интерфейсов:**
Привязка интерфейсов к VRF. Привязка IGP протокола (если имеется) между PE и CE к соответствующему VRF и каждого VRF к BGP в виде адресных семейств ipv4.

- Настройка маршрутизации между PE и CE.
Конфигурация протоколов маршрутизации (статика, OSPF, eBGP) в контексте VRF и их интеграция с MP-BGP через адресные семейства IPv4.
- Проверка работоспособности:
Проверка установления соседств, обмена маршрутами и сквозной связности между клиентскими объектами.

В приведённом ниже примере топологии к опорной MPLS-VPN сети оператора подключены два клиента: GOLD и SILVER. Сеть оператора состоит из трёх маршрутизаторов. ECO1 и ECO3 являются PE-маршрутизаторами, ECO2 — P-маршрутизатором.

Постройте схему в соответствии с рисунком ниже:

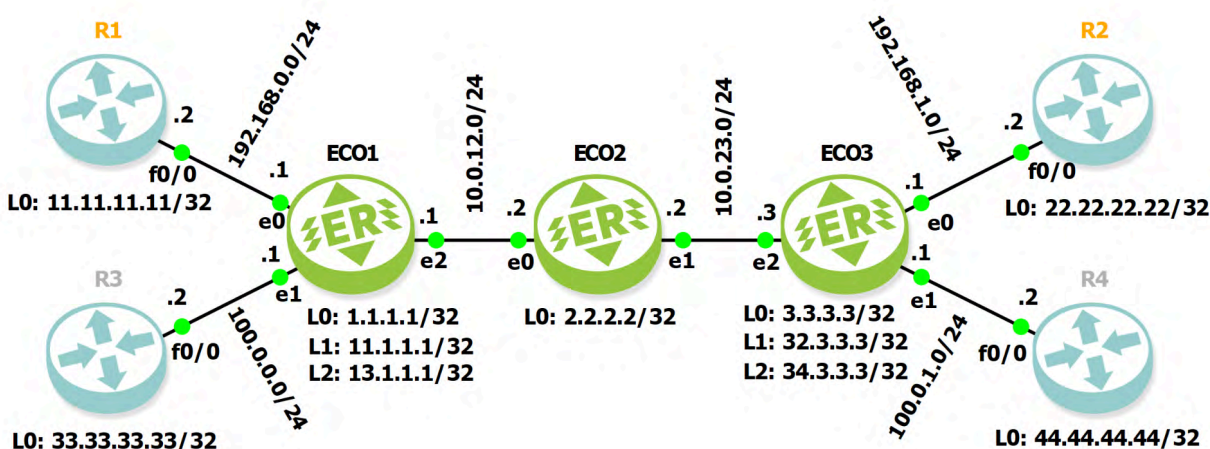


Рисунок 46

Полные конфигурации всех узлов доступны по ссылке <https://docs.ecorouter.ru/Руководство/901-Примеры-конфигураций/MPLS-L3-VPN-1>.

Настройте ECO1 и R1, как показано ниже.

ECO1:

```
enable
configure

hostname ECO1

ip vrf GOLD
# Route Distinguisher должен быть уникален для каждого PE в каждом VRF
rd 2000:1
# Route Target должна быть идентична для всех PE в каждом VRF
route-target both 2000:100
```

```
ip vrf SILVER
  rd 2000:11
  route-target both 2000:200

# Настройка LDP для распространения меток
router ldp
  # Настройка loopback.0 в качестве транспортного адреса
  transport-address ipv4 1.1.1.1
  # Включение режима распространения меток без запроса
  advertisement-mode downstream-unsolicited

# Настройка OSPF для создания сети между CE (клиентом) и PE (провайдером)
router ospf 1 GOLD
  # Включить маршруты из BGP в OSPF для их анонсирования
  redistribute bgp
  # Включить loopback-адрес в маршрутизацию
  network 11.1.1.1/32 area 0
  # Включить подсеть между CE и PE в маршрутизацию
  network 192.168.0.0 0.0.0.255 area 0
  # Отключить функции маршрутизации с учётом ограничений
  no capability cspf

router ospf 2 SILVER
  redistribute bgp
  network 13.1.1.1/32 area 0
  network 100.0.0.0 0.0.0.255 area 0
  no capability cspf

# Настройка isis в качестве IGP-протокола MPLS-сети
router isis 1
  net 10.0000.0000.0001.00

# Настройка MP-BGP между PE-маршрутизаторами
# а также настройка адресных семейств для клиентских VRF
router bgp 2000
  # Настройка удалённого BGP-соседа
  neighbor 3.3.3.3 remote-as 2000
  neighbor 3.3.3.3 update-source 1.1.1.1
  address-family vpnv4 unicast
    neighbor 3.3.3.3 activate
    exit-address-family
  # Настройка адресного пространства для каждого VRF
  address-family ipv4 vrf GOLD
```




```
# Включить маршруты из OSPF в BGP для их анонсирования
  redistribute ospf
  exit-address-family
address-family ipv4 vrf SILVER
  redistribute ospf
  exit-address-family

port ge0
  service-instance 4ge0
  encapsulation untagged

port ge1
  service-instance 4ge1
  encapsulation untagged

port ge2
  service-instance 4ge2
  encapsulation untagged

# Интерфейс маршрутизатора в сети MPLS
interface loopback.0
  ip address 1.1.1.1/32
  ip router isis 1

# Интерфейс маршрутизатора в сети PE-CE
interface loopback.1
  ip vrf forwarding GOLD
  ip address 11.1.1.1/32

interface loopback.2
  ip vrf forwarding SILVER
  ip address 13.1.1.1/32

# Интерфейс в сторону CE
interface 2ge0
  description To GOLD R1
  # Привязка интерфейса к VRF
  ip vrf forwarding GOLD
  connect port ge0 service-instance 4ge0
  ip address 192.168.0.1/24

# Интерфейс в сторону CE
interface 2ge1
```



```
description To SILVER R3
ip vrf forwarding SILVER
connect port ge1 service-instance 4ge1
ip address 100.0.0.1/24
```

Интерфейс в сторону соседнего маршрутизатора в сети MPLS

```
interface 2ge2
description To EC02
connect port ge2 service-instance 4ge2
ip address 10.0.12.1/24
ip router isis 1
label-switching
ldp enable ipv4
```

R1:

```
configure terminal
hostname R1
```

Настройка loopback интерфейса маршрутизатора

```
interface Loopback0
ip address 11.11.11.11 255.255.255.255
# Привязка интерфейса к процессу OSPF
ip ospf 1 area 0
```

Настройка интерфейса в сторону PE

```
interface FastEthernet0/0
ip address 192.168.0.2 255.255.255.0
duplex full
no shutdown
```

Настройка маршрутизации OSPF

```
router ospf 1
# Loopback0 объявляется пассивным чтобы с ним не формировалось соседство
passive-interface Loopback0
# Включить сеть между PE и CE в маршрутизацию
network 192.168.0.0 0.0.0.255 area 0
```

В соответствии со схемой и примерами настройки выше, настройте остальные узлы.

Для проверки конфигурации выполните команды:

- R1: `ping 22.22.22.22 source 11.11.11.11 ;`
- R3: `ping 44.44.44.44 source 33.33.33.33 .`



При необходимости отладки воспользуйтесь командами:

- `show ldp session` на ECO1-3 для проверки LDP соседства;
- `show bgp summary` для проверки соседства между PE;
- `show ip bgp vpnv4 all` для проверки установления BGP-сессий VPNv4.
- `show ip bgp vpnv4 vrf <NAME> labels` для проверки какие сервисные метки присвоены маршрутам;
- `show mpls vrf-table` для проверки сервисных меток, присланных удалённым PE;
- `show mpls ilm-table` - для проверки сервисных меток, которые были выделены локально;
- `show mpls forwarding-table` для проверки наличия транспортных меток до next-hop.

Для передачи маршрутной информации частных сетей через сеть оператора используется протокол BGP и его многопротокольное расширение MP-BGP. Это позволяет обмениваться информацией между опосредованно соединёнными маршрутизаторами, а также передавать маршрутную информацию сетей VPN, минуя маршрутизаторы опорной сети оператора (P) посредством коммутации по меткам. Через P-маршрутизаторы информация передаётся 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 прикреплённых меток. Таким образом, на P-маршрутизаторах не требуется конфигурацию сетей VPN (VRF) и маршрутизации BGP.

Каждый PE-маршрутизатор в опорной сети MPLS-VPN подсоединён к узлам, входящим в виртуальные частные сети клиентов. Для каждого узла действуют маршруты соответствующей сети VPN. Поэтому на PE-маршрутизаторе должны содержаться таблицы VRF для тех сетей VPN, к узлам которых он подключён.

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 бита (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 просмотра таблицы маршрутизации данного VRF используется команда `show ip route vrf <VRF_NAME>` или команда `show ip route vrf all` для всех VRF.



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.

Команда **route-target** создаёт 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 задаётся 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 определённой сети VPN приводит к полносвязной топологии — каждый сайт может посылать пакеты непосредственно тому сайту, в котором находится сеть назначения.

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 счёт распространения по сети маршрутной информации протоколами маршрутизации, такими как BGP, OSPF, IS-IS, RIP.

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 определённой VRF, связывая конкретный клиентский интерфейс с таблицей VRF и с определённым 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.

Заменяем протокол между CE и PE с OSPF на eBGP.

На ECO1 выполните следующую конфигурацию:

```
enable
configure
```



```
# Удалить настройки OSPF и лишние интерфейсы loopback
no router ospf 1
no router ospf 2
no interface loopback.1
no interface loopback.2

router bgp 2000
address-family ipv4 vrf GOLD
# Отменить команду о переназначении OSPF маршрутов
no redistribute ospf
# Установить eBGP соседство с CE
neighbor 192.168.0.2 remote-as 100
neighbor 192.168.0.2 activate
exit-address-family
address-family ipv4 vrf SILVER
no redistribute ospf
neighbor 100.0.0.2 remote-as 300
neighbor 100.0.0.2 activate
exit-address-family
```

Сконфигурируйте ECO3 аналогично ECO1.

На R1 выполните следующую конфигурацию:

```
configure terminal
# Удалить настройки маршрутизации OSPF
no router ospf 1

router bgp 100
# Включить адрес loopback0 в маршрутизацию
network 11.11.11.11 mask 255.255.255.255
# Установить eBGP соседство с PE
neighbor 192.168.0.1 remote-as 2000
```

Сконфигурируйте R2, R3 и R4 аналогично R1. Присвойте автономным системам номера 100, 200, 300 и 400 в соответствии с номером маршрутизатора на схеме.

Выполните проверку и отладку аналогично предыдущей конфигурации (с OSPF).

BGP-сессия между PE и CE-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 CE-маршрутизаторов с VRF.

PE и CE-маршрутизаторы должны быть напрямую подключены для BGP4-сессий; BGP multihop между ними не поддерживается.

26.5.4 VPN между AS с помощью eBGP

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 разных AS соединены между собой по eBGP, VRF к VRFy. Такой способ организации VPN между автономными системами называется BGP Inter-AS Option A (VRF-to-VRF).

Постройте схему в соответствии с рисунком ниже:

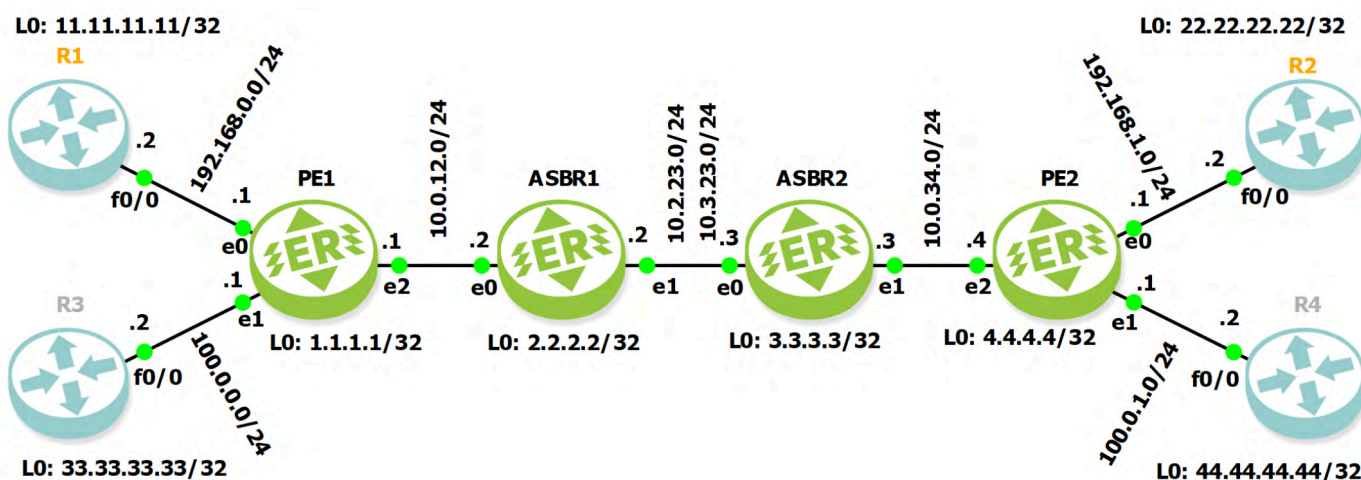


Рисунок 47

Полные конфигурации всех узлов доступны по ссылке <https://docs.ecorouter.ru/Руководство/901-Примеры-конфигураций/MPLS-L3-VPN-Inter-AS-Option-A>.

Схема во многом повторяет схему из пункта "Конфигурирование MPLS Layer-3 VPN" в её втором варианте с eBGP между CE и PE. Конфигурации R1, R2, R3, R4 полностью идентичны. Конфигурации PE1 и PE2 практически идентичны ECO1 и ECO3, за исключением настроек BGP. Их роль не меняется, они по-прежнему являются PE маршрутизаторами и связывают каждый CE со своим VRF, но теперь PE1 вместе с ASBR1 принадлежат одной автономной системе (AS 1000), а PE2 и ASBR2 другой (AS 2000).

Конфигурация BGP на PE1


```
# Автономная система BGP 1000
router bgp 1000
# Установить iBGP соседство с ASBR1
neighbor 2.2.2.2 remote-as 1000
neighbor 2.2.2.2 update-source 1.1.1.1
address-family vpnv4 unicast
    neighbor 2.2.2.2 activate
    exit-address-family
# Далее конфигурация без изменений
address-family ipv4 vrf GOLD
    neighbor 192.168.0.2 remote-as 100
    neighbor 192.168.0.2 activate
    exit-address-family
address-family ipv4 vrf SILVER
    neighbor 100.0.0.2 remote-as 300
    neighbor 100.0.0.2 activate
    exit-address-family
```

Адреса интерфейсов на PE1 не меняются. А вот на PE адрес интерфейса **2ge2** поменялся, так как появился новый маршрутизатор ASBR2 и новая подсеть 10.0.34.0/24.

Конфигурация BGP и интерфейса **2ge2** на PE2:

```
router bgp 2000
# Установить iBGP соседство с ASBR2
neighbor 3.3.3.3 remote-as 2000
neighbor 3.3.3.3 update-source 4.4.4.4
address-family vpnv4 unicast
    neighbor 3.3.3.3 activate
    exit-address-family
# Далее конфигурация без изменений
address-family ipv4 vrf GOLD
    neighbor 192.168.1.2 remote-as 200
    neighbor 192.168.1.2 activate
    exit-address-family
address-family ipv4 vrf SILVER
    neighbor 100.0.1.2 remote-as 400
    neighbor 100.0.1.2 activate
    exit-address-family
. . .
# Интерфейс связывающий PE2 с ASBR2
interface 2ge2
    connect port ge2 service-instance 4ge2
# Адрес в новой подсети 10.0.34.0/24 между PE2 и ASBR2
ip address 10.0.34.4/24
```



```
ip router isis 1
label-switching
ldp enable ipv4
```

Маршрутизаторов между PE1 и PE2 теперь два и они несут гораздо больше функций чем ECO3 из предыдущей схемы. Помимо транспорта MPLS они должны направлять трафик каждого VRF в соответствующие интерфейсы связанные eBGP соседством.

Полная конфигурация ASBR1:

```
enable
configure

hostname ASBR1

# Настройки VRF идентичны PE1 (для ASBR2 идентичны PE2)
ip vrf SILVER
  rd 2000:11
  route-target both 2000:200

ip vrf GOLD
  rd 2000:1
  route-target both 2000:100

router ldp
  # В качестве транспортного адреса LDP указывается loopback.0 маршрутизатора
  transport-address ipv4 2.2.2.2
  advertisement-mode downstream-unsolicited

router isis 1
  # NSAP присваивается соответственно loopback.0: 0001, 0002, 0003, 0004
  net 10.0000.0000.0002.00

router bgp 1000
  # Соседство с PE1 (для ASBR2 с PE2 по адресу 4.4.4.4)
  neighbor 1.1.1.1 remote-as 1000
  address-family vpnv4 unicast
    neighbor 1.1.1.1 activate
    exit-address-family
  # Соседство с интерфейсом связанным с VRF GOLD (для ASBR2 адрес 10.2.23.2)
  address-family ipv4 vrf GOLD
    neighbor 10.2.23.3 remote-as 2000
    neighbor 10.2.23.3 activate
    exit-address-family
```




```
# Соседство с интерфейсом связанным с VRF SILVER (для ASBR2 адрес 10.3.23.2)
address-family ipv4 vrf SILVER
  neighbor 10.3.23.3 remote-as 2000
  neighbor 10.3.23.3 activate
exit-address-family

# Порт в сторону PE1
port ge0
  service-instance 4ge0
  encapsulation untagged

# Порт в сторону ASBR2
port ge1
  # Сервисный интерфейс для VRF GOLD (на ASBR2 4ge02)
  service-instance 4ge12
  # Настройка VLAN-тега (на ASBR2 идентичная)
  encapsulation dot1q 231
  rewrite pop 1
  # Сервисный интерфейс для VRF SILVER (на ASBR2 4ge03)
  service-instance 4ge13
  # Настройка VLAN-тега (на ASBR2 идентичная)
  encapsulation dot1q 232
  rewrite pop 1

interface loopback.0
  ip address 2.2.2.2/32
  ip router isis 1

# Интерфейс к PE1 (на ASBR2 2ge1 к PE2)
interface 2ge0
  label-switching
  connect port ge0 service-instance 4ge0
  # На ASBR2 10.0.34.3/24
  ip address 10.0.12.2/24
  ip router isis 1
  ldp enable ipv4

# Интерфейс к VRF GOLD (на ASBR2 2ge02)
interface 2ge12
  # Привязка интерфейса к VRF
  ip vrf forwarding GOLD
  # На ASBR2 4ge02
  connect port ge1 service-instance 4ge12
```



```
# На ASBR2 10.2.23.3/24
ip address 10.2.23.2/24

# Интерфейс к VRF SILVER (на ASBR2 2ge03)
interface 2ge13
# Привязка интерфейса к VRF
ip vrf forwarding SILVER
# На ASBR2 4ge03
connect port ge1 service-instance 4ge13
# На ASBR2 10.3.23.3/24
ip address 10.3.23.2/24
```

Для проверки конфигурации выполните команды:

- R1: `ping 22.22.22.22 source 11.11.11.11 ;`
- R3: `ping 44.44.44.44 source 33.33.33.33 .`

При необходимости отладки воспользуйтесь командами:

- `show ldp session` на ECO1-3 для проверки LDP соседства;
- `show bgp summary` для проверки соседства между PE;
- `show ip bgp vpnv4 all` для проверки установления BGP-сессий VPNv4.
- `show ip bgp vpnv4 vrf <NAME> labels` для проверки какие сервисные метки присвоены маршрутам;
- `show mpls vrf-table` для проверки сервисных меток, присланных удалённым PE;
- `show mpls ilm-table` - для проверки сервисных меток, которые были выделены локально;
- `show mpls forwarding-table` для проверки наличия транспортных меток до next-hop.

26.5.5 Настройка eBGP между PE и RR и между ASBR

В данном примере PE-маршрутизатор соединён с Route-Reflector (RR), одним из клиентов которого является ASBR, соединённый 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.

На рисунке ниже приведена топология сети для данного примера.

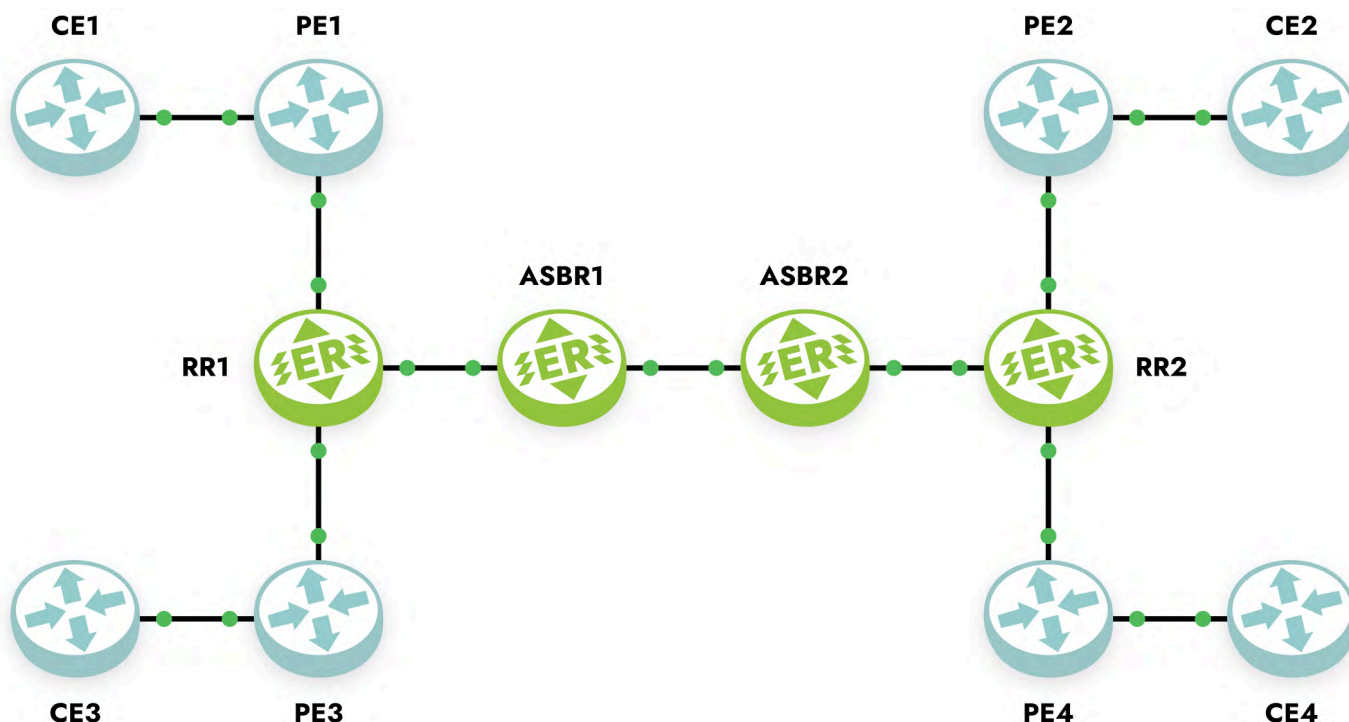


Рисунок 48

Ниже представлены команды конфигурирования маршрутизаторов CE, PE, RR и ASBR в соответствии с топологией сети.

Для настройки CE-маршрутизаторов используются те же команды, что и в примере "Настройка eBGP между PE и ASBR".

Для настройки 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.

Таблица 92 — Настройка Route Reflectors

Команда	Описание
<code>#configure terminal</code>	Вход в конфигурационный режим
<code>(config)#ip vrf IPI</code>	Создание VRF под названием IPI
<code>(config-vrf)#rd 1:100</code>	Назначение RD 1:100
<code>(config-vrf)#route-target both 100:200</code>	Настройка импорта маршрутов между RT расширенных сообществ 100 и 200
<code>(config-vrf)#exit</code>	Выход из конфигурирования VRF
<code>(config)#interface eth1</code>	Вход в режим конфигурирования интерфейса

Команда	Описание
<code>(config-if)#ip address 172.4.5.114/24</code>	Назначение IP-адреса
<code>(config-if)#exit</code>	Выход из режима конфигурирования интерфейса
<code>(config)#router bgp 1</code>	Определение процесса BGP маршрутизации для AS 1
<code>(config-router)#neighbor 172.5.6.116 remote-as 1</code>	Добавление ASBR в качестве однорангового iBGP устройства с IP-адресом 172.5.6.116 и AS 1
<code>(config-router)#neighbor 172.4.5.114 remote-as 1</code>	Добавление ASBR в качестве однорангового iBGP устройства с IP-адресом 172.4.5.114 и AS 1
<code>(config-router)#address-family vpnv4 unicast</code>	Вход в режим конфигурирования семейства адресов VPNv4
<code>(config-router-af)#neighbor 172.5.6.116 activate</code>	Активировать PE-маршрутизатор для обработки маршрутов сети VPN
<code>(config-router-af)#neighbor 172.5.6.116 route-reflector-client</code>	Добавить PE-маршрутизатор, как route-reflector-client
<code>(config-router-af)#neighbor 172.4.5.114 activate</code>	Активация ASBR-соседства, чтобы ASBR мог принимать маршруты сети VPN
<code>(config-router-af)#neighbor 172.4.5.114 route-reflector-client</code>	Добавить ASBR, как route-reflector-client
<code>(config-router-af)#exit-address-family</code>	Выход из режима конфигурирования семейства адресов VPNv4
<code>(config-router)#exit</code>	Выход из режима конфигурирования маршрутизации

Для проверки настроенной конфигурации используются команды административного режима `show ip bgp neighbors` , `show ip bgp vpnv4 all` .

Для настройки 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.

26.5.6 Соединение PE-маршрутизаторов с использованием eBGP Multi-hop

В данном примере PE-маршрутизаторы подключены друг к другу напрямую с использованием eBGP multi-hop.

Между CE 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, P и PE2 должен быть запущен протокол IGP.

На рисунке ниже приведена топология сети для данного примера.

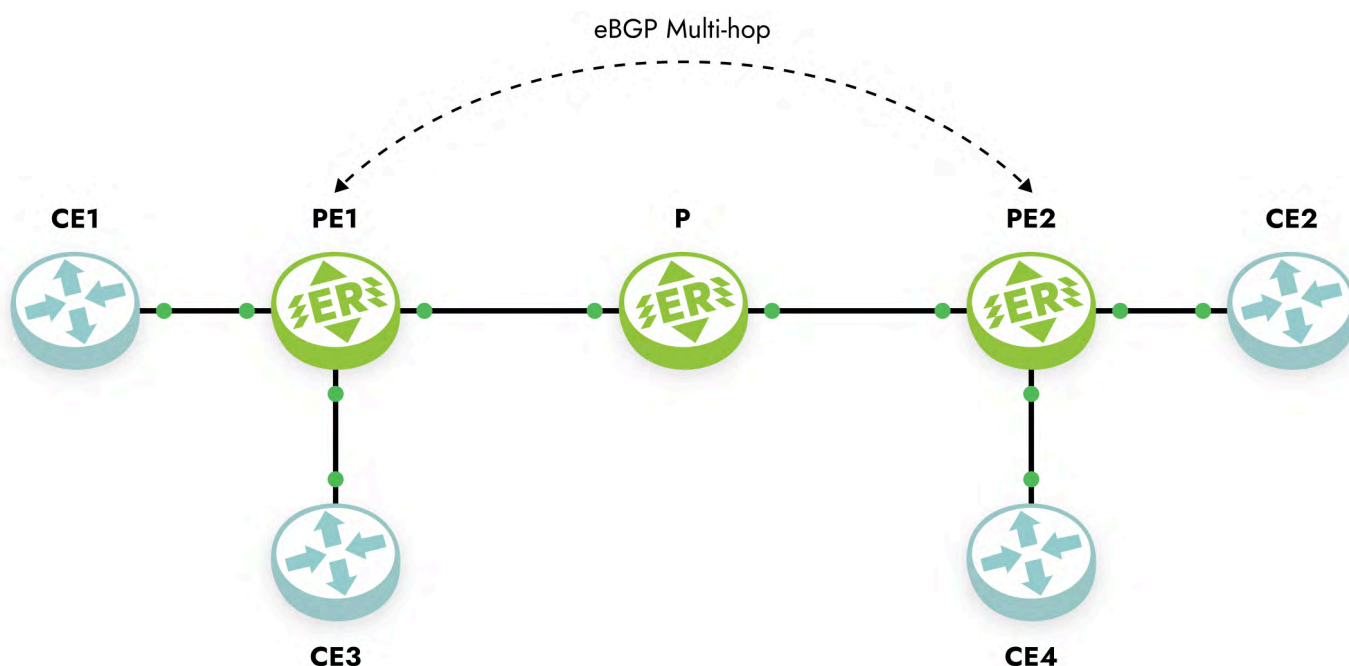


Рисунок 49

Ниже представлены команды конфигурирования маршрутизаторов CE и PE в соответствии с топологией сети.

На P-маршрутизаторах должен быть настроен только протокол IGP (в данном примере OSPF).

Таблица 93 — Настройка CE-маршрутизаторов

Команда	Описание
<code>#configure terminal</code>	Вход в конфигурационный режим
<code>(config)#interface eth1</code>	Вход в режим конфигурирования интерфейса
<code>(config-if)#ip address 172.6.7.117/24</code>	Назначение IP-адреса
<code>(config-if)#exit</code>	Выход из режима конфигурирования интерфейса

Команда	Описание
<code>(config)#router bgp 65001</code>	Определение процесса BGP маршрутизации для AS 65001
<code>(config-router)#neighbor 172.6.7.116 remote-as 1</code>	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`.

Таблица 94 — Настройка PE-маршрутизаторов

Команда	Описание
<code>#configure terminal</code>	Вход в конфигурационный режим
<code>(config)#ip vrf IPI</code>	Создание VRF под названием IPI
<code>(config-vrf)#rd 1:100</code>	Назначение RD 1:100
<code>(config-vrf)#route-target both 100:200</code>	Настройка импорта маршрутов между RT расширенных сообществ 100 и 200
<code>(config-vrf)#exit</code>	Выход из конфигурирования VRF
<code>(config)#interface eth3</code>	Вход в режим конфигурирования интерфейса
<code>(config-if)#ip vrf forwarding IPI`</code>	Bind the interface connected to the CE router with VRF IPI.
<code>(config-if)#ip address 172.6.7.116/24</code>	Назначение IP-адреса
<code>(config-if)#exit</code>	Выход из режима конфигурирования интерфейса
<code>(config)#router ospf 1</code>	Определение процесса маршрутизации OSPF
<code>(config-router)#network 172.5.6.0/24 area 0</code>	Рекламировать сеть между PE и P-маршрутизатором для того, чтобы обеспечить multi-hop
<code>(config-router)#exit</code>	Выход из режима конфигурации маршрутизации OSPF
<code>(config)#router bgp 1</code>	Определение процесса BGP маршрутизации для AS 1
<code>(config-router)#neighbor 172.4.5.114 remote-as 2</code>	Определение PE-маршрутизатора, как соседа. Здесь 172.4.5.114 — IP-адрес удаленного PE-маршрутизатора, 2 — номер AS

Команда	Описание
<code>(config-router)#neighbor 172.4.5.114 ebgp-multi-hop 255</code>	Установление PE-маршрутизатора в качестве однорангового устройства eBGP
<code>(config-router)#address-family vpnv4 unicast</code>	Вход в режим конфигурирования семейства адресов VPNv4
<code>(config-router-af)#neighbor 172.4.5.114 allow-ebgp-vpn</code>	Настройка удаленного PE-маршрутизатора для разрешения eBGP сетей VPN
<code>(config-router-af)#neighbor 172.4.5.114 activate</code>	Активация удаленного PE-маршрутизатора, чтобы он мог получать маршруты сети VPN
<code>(config-router-af)#exit-address-family</code>	Выход из режима конфигурирования семейства адресов VPNv4
<code>(config-router)#address-family ipv4 vrf IPI</code>	Вход в режим конфигурирования семейства адресов IPv4 для VRF IPI
<code>(config-router-af)#neighbor 172.6.7.117 remote-as 65001</code>	Определение CE-маршрутизатора, как соседа с IP-адресом 172.6.7.117 и номером AS 65001
<code>(config-router-af)#exit-address-family</code>	Выход из режима конфигурирования семейства адресов IPv4
<code>(config-router)#exit</code>	Выход из режима конфигурирования маршрутизации

Для проверки настроенной конфигурации используются команды административного режима `show ip bgp neighbors`, `show ip bgp vpnv4 all`.

26.5.7 Соединение PE-маршрутизаторов с RR через RR, используя eBGP multi-hop

В данном примере PE-маршрутизаторы подсоединены к Route-Reflector (RR), которые подсоединены к другим RR, используя eBGP-multi-hop.

Конфигурация аналогична предыдущему примеру "Соединение PE-маршрутизаторов с использованием eBGP Multi-hop", кроме того, что PE-маршрутизаторы подсоединены к RR по iBGP. EBGP multi-hop соединения остаются только между RR.

На рисунке ниже приведена топология сети для данного примера.

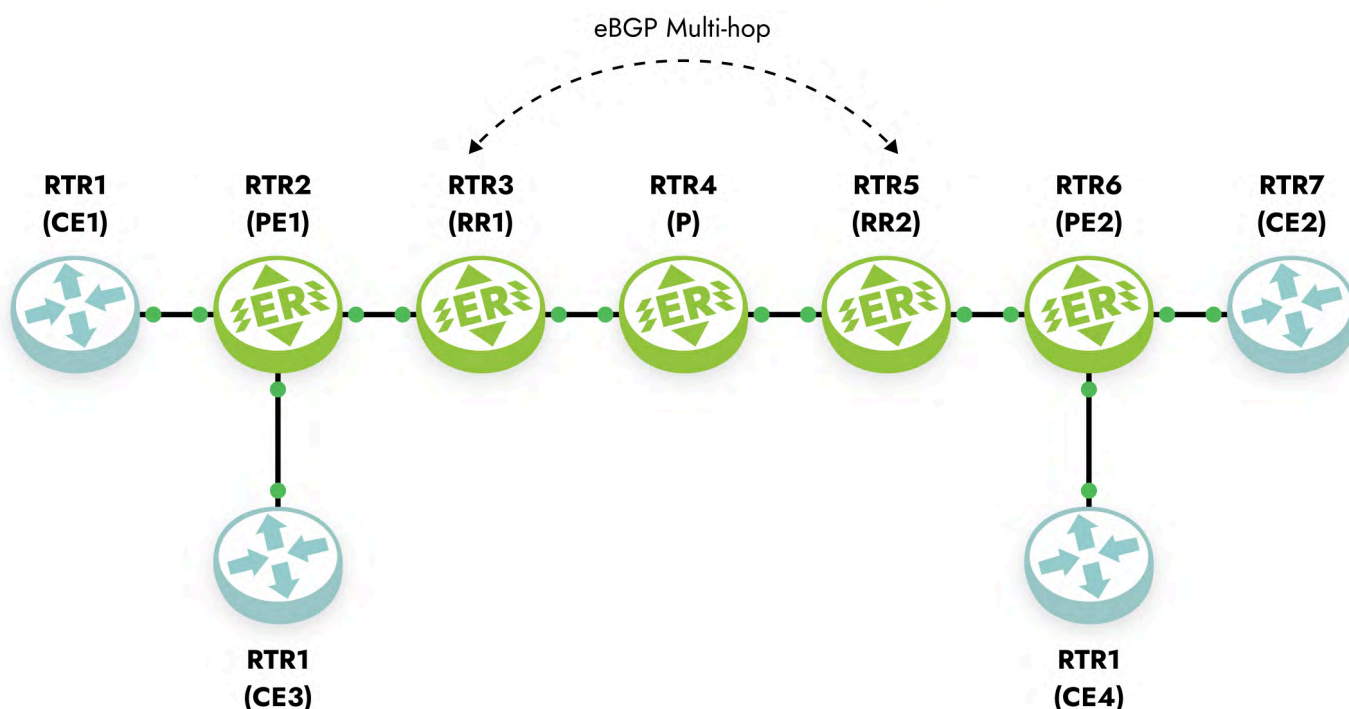


Рисунок 50

Ниже представлены команды конфигурирования маршрутизаторов CE, PE и RR в соответствии с топологией сети.

На P-маршрутизаторах должен быть настроен только протокол IGP (в данном примере OSPF).

CE-маршрутизаторы настраиваются аналогично примеру "Соединение PE-маршрутизаторов с использованием eBGP Multi-hop".

PE-маршрутизаторы настраиваются аналогично примеру "Соединение PE-маршрутизаторов с использованием eBGP Multi-hop", кроме того, что у PE-маршрутизаторов есть только одно iBGP соединение с RR.

Таблица 95 — Настройка Route Reflectors

Команда	Описание
<code>#configure terminal</code>	Вход в конфигурационный режим
<code>(config)#ip vrf IPI</code>	Создание VRF под названием IPI
<code>(config-vrf)#rd 1:100</code>	Назначение RD 1:100
<code>(config-vrf)#route-target both 100:200</code>	Настройка импорта маршрутов между RT расширенных сообществ 100 и 200
<code>(config-vrf)#exit</code>	Выход из конфигурирования VRF
<code>(config)#interface eth1</code>	Вход в режим конфигурирования интерфейса

Команда	Описание
<code>(config-if)#ip address 172.5.6.115/24</code>	Назначение IP-адреса
<code>(config-if)#exit</code>	Выход из режима конфигурирования интерфейса
<code>(config)#router bgp 1</code>	Определение процесса BGP маршрутизации для AS 1
<code>(config-router)#neighbor 172.5.6.116 remote-as 1</code>	Добавление ASBR в качестве однорангового iBGP устройства с IP-адресом 172.5.6.116 и AS 1
<code>(config-router)#neighbor 172.3.4.113 remote-as 2</code>	Добавление удалённого RR в качестве однорангового iBGP устройства с IP-адресом 172.3.4.113 и AS 2
<code>(config-router)#neighbor 172.3.4.113 ebgp-multi-hop 255</code>	Назначение удалённого RR-маршрутизатора в качестве однорангового устройства eBGP-multi-hop
<code>(config-router)#address-family vpnv4 unicast</code>	Вход в режим конфигурирования семейства адресов VPNv4
<code>(config-router-af)#neighbor 172.3.4.113 allow-ebgp-vpn</code>	Настройка удалённого RR, чтобы разрешить EBGP сети VPN
<code>(config-router-af)#neighbor 172.3.4.113 activate</code>	Активация соседства, чтобы удалённый RR мог принимать маршруты сети VPN
<code>(config-router-af)#neighbor 172.5.6.116 activate</code>	Активация PE-маршрутизатора для обработки маршрутов сети VPN
<code>(config-router-af)#neighbor 172.5.6.116 route-reflector-client</code>	Добавление PE-маршрутизатора в качестве route-reflector-client
<code>(config-router-af)#exit-address-family</code>	Выход из режима конфигурирования семейства адресов VPNv4
<code>(config-router)#exit</code>	Выход из режима конфигурирования маршрутизации
<code>(config)#router ospf 1</code>	Определение процесса маршрутизации OSPF
<code>(config-router)#network 172.4.5.0/24 area 0</code>	Рекламировать сеть между PE и P-маршрутизатором для того, чтобы обеспечить multi-hop
<code>(config-router)#exit</code>	Выход из режима конфигурирования маршрутизации

Для проверки настроенной конфигурации используются команды административного режима `show ip bgp neighbors`, `show ip bgp vpnv4 all`.

26.6 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, терминирующий 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 однонаправленный виртуальный канал;
- VSI (Virtual Switch Instance) — виртуальный Ethernet-bridge, терминирующий AC со стороны клиентов и VC со стороны сети провайдера. VPLS-instance — синоним VSI.

На схеме ниже изображены основные устройства и каналы VPLS-сети.

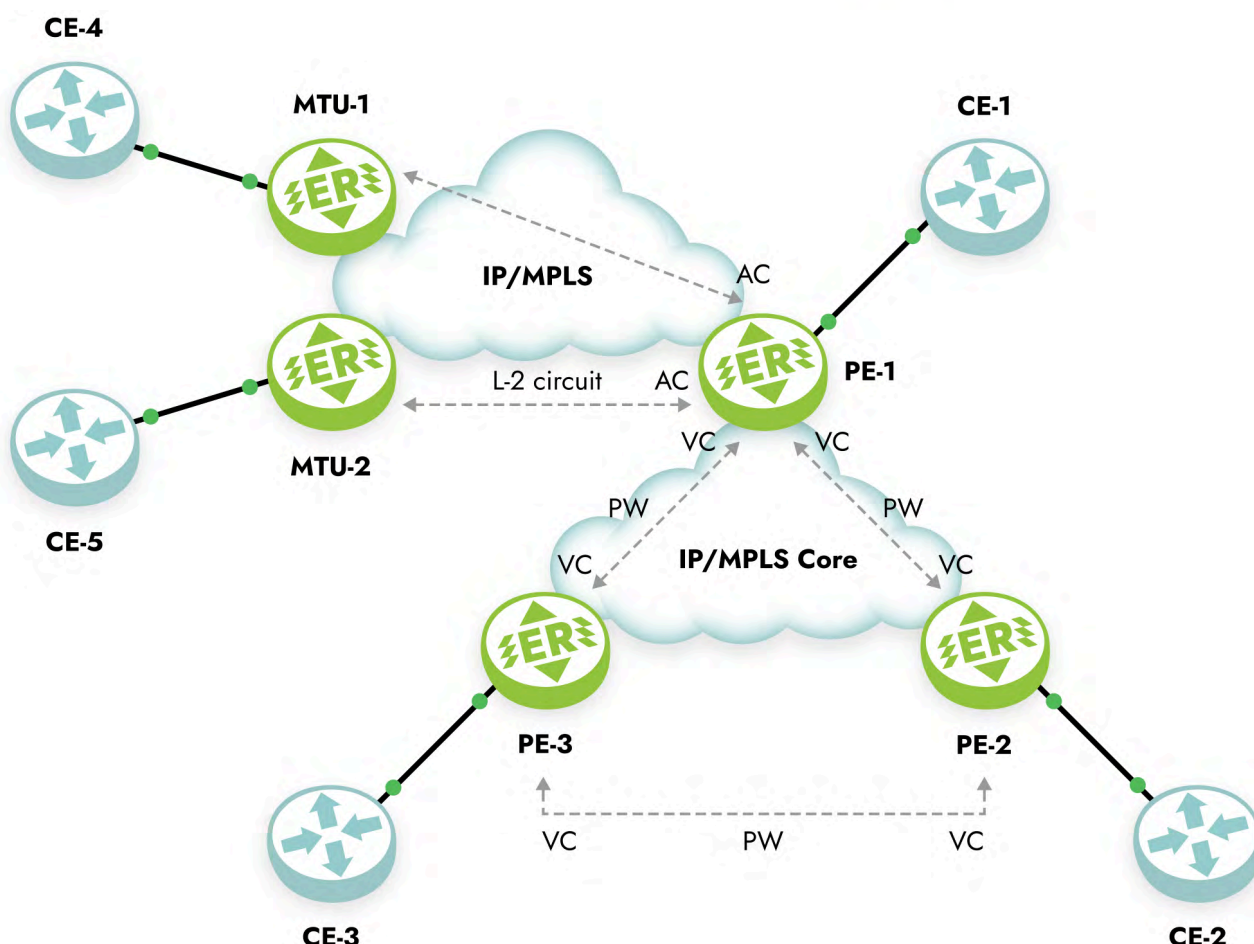


Рисунок 51

26.6.1 Общие требования для работы VPLS

Сервис VPLS работает поверх IP/MPLS-сети, соответственно, для организации его работы необходимо, чтобы между устройствами PE (Provider Edge, граничные устройства сети провайдера) была IP-связность, а также функционировал MPLS-транспорт на основе LDP. Между устройствами PE должна быть установлена tLDP или MP-BGP сессия, используемая для обмена сервисными MPLS-метками.

Аналогичные требования существуют для связности устройств PE и MTU-г (Multi-Tenant Unit router, вид PE-роутера, который агрегирует трафик от множества клиентов). Сами устройства MTU-г могут быть в разных сетях и не иметь IP-связности друг с другом.

26.6.2 Схема с одним PE, терминирующим L2-circuit

Простейшая схема использования сервиса VPLS выглядит следующим образом (см. рисунок ниже).

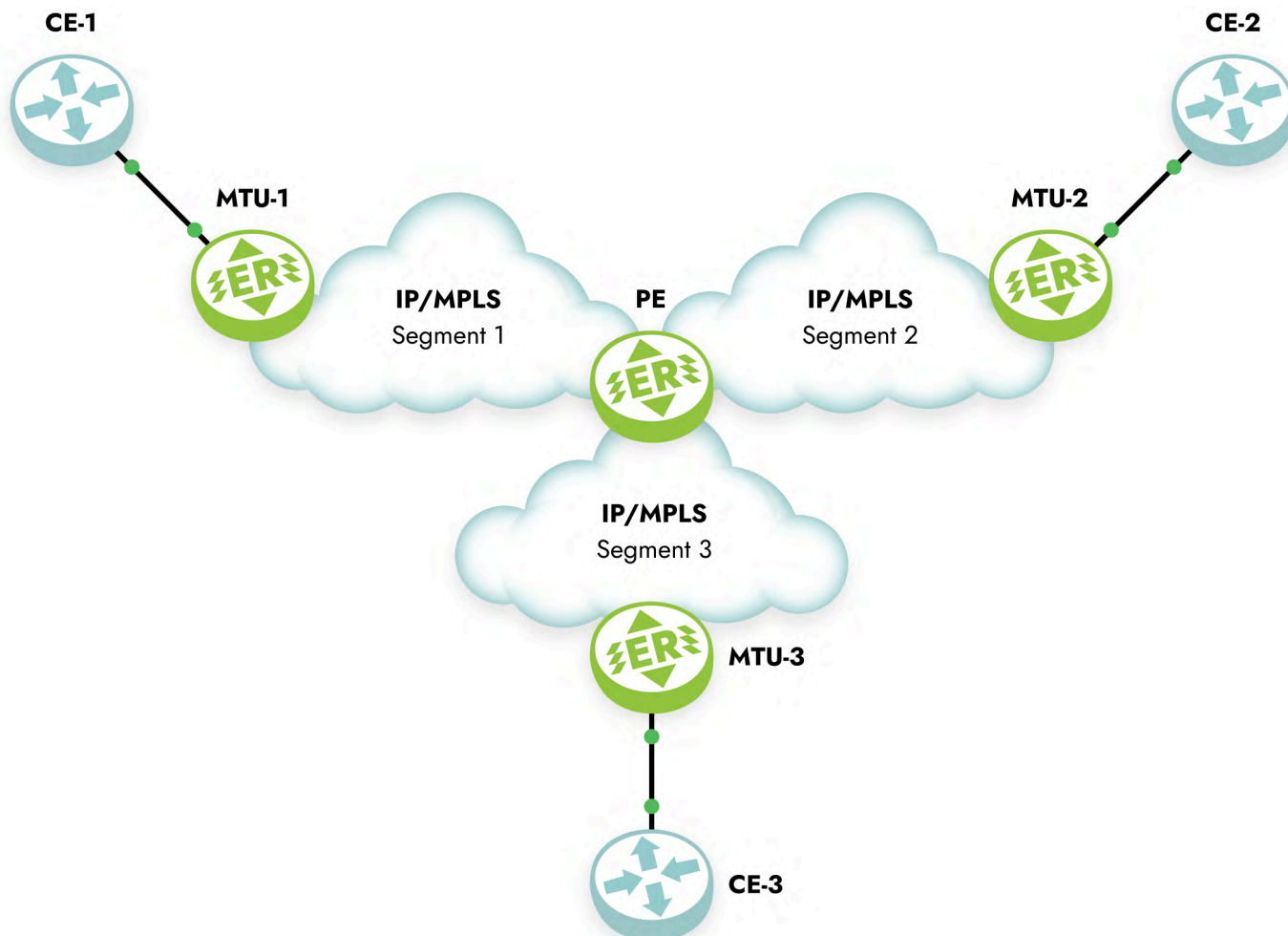


Рисунок 52

Устройство PE терминирует в одном VPLS-доме 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-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-г.

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
```

26.6.3 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).

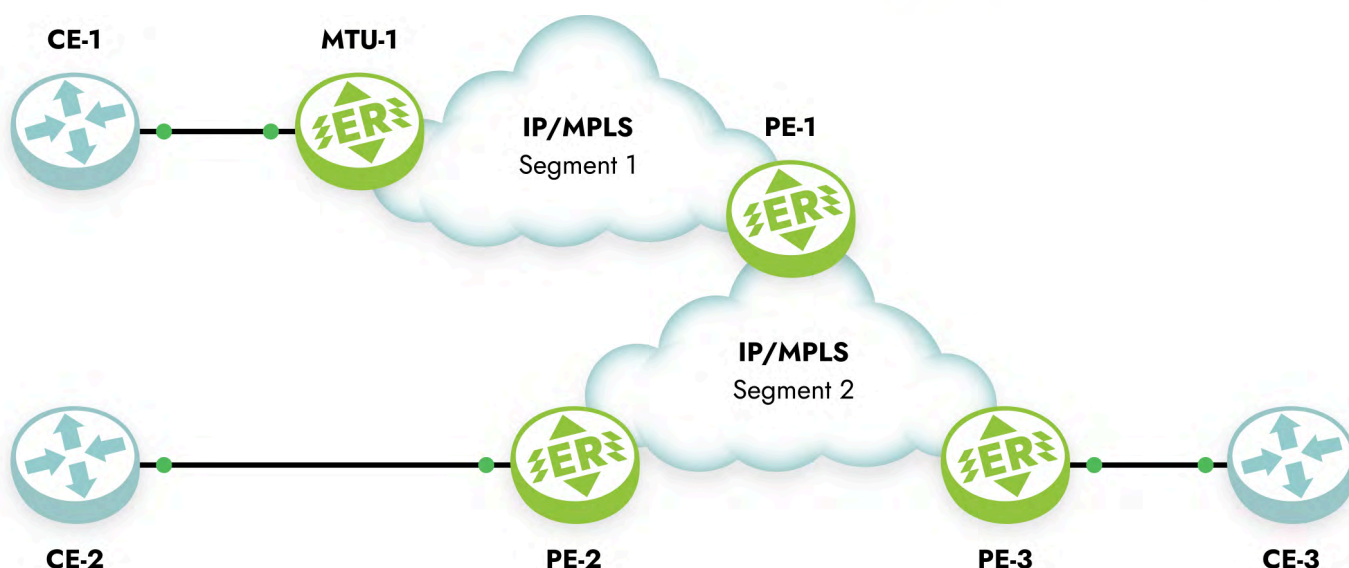


Рисунок 53

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 12-circuit vc10 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 порте.

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.

26.6.4 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
```


26.6.5 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 реер-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-г должны совпадать.

26.6.6 VPLS режим Martini

Реализация VPLS Martini требует развёртывания базовой MPLS-инфраструктуры с использованием LDP в качестве протокола транспорта и IGP (OSPF или IS-IS) для обеспечения связности между всеми PE-маршрутизаторами. Критически важным элементом являются Targeted LDP сессии, которые устанавливаются непосредственно между loopback-адресами всех участвующих в сервисах VPLS PE-устройств, образуя полносвязную сеть управляющих соединений. Эти сессии служат основой для обмена управляющими данными и установки псевдопроводов.

Основная конфигурация VPLS Martini заключается в создании виртуальных экземпляров VPLS на каждом PE-маршрутизаторе. В отличие от Kompella, где используется автоматическое обнаружение, в Martini необходимо вручную прописывать каждого удалённого PE-соседа в конфигурации экземпляра VPLS, что приводит к необходимости большого количества настроек соединений для полносвязной топологии.



Клиентский трафик передаётся через физические или логические интерфейсы, которые привязываются к соответствующим экземплярам VPLS. Поддерживаются различные типы инкапсуляции, включая чистый Ethernet для нетегированного трафика и Ethernet VLAN для работы с тегированными кадрами. Особое внимание следует уделить согласованию MTU на всех участках сети, поскольку недостаточный размер MTU может привести к фрагментации или потере MPLS-пакетов.

Основным ограничением подхода Martini является его плохая масштабируемость — каждое добавление нового маршрутизатора в сеть VPLS требует ручного изменения конфигурации на всех существующих PE. Однако этот подход остаётся популярным в небольших сетях благодаря своей простоте, предсказуемости и широкой поддержке оборудованием различных производителей.

Постройте схему в соответствии с рисунком ниже:

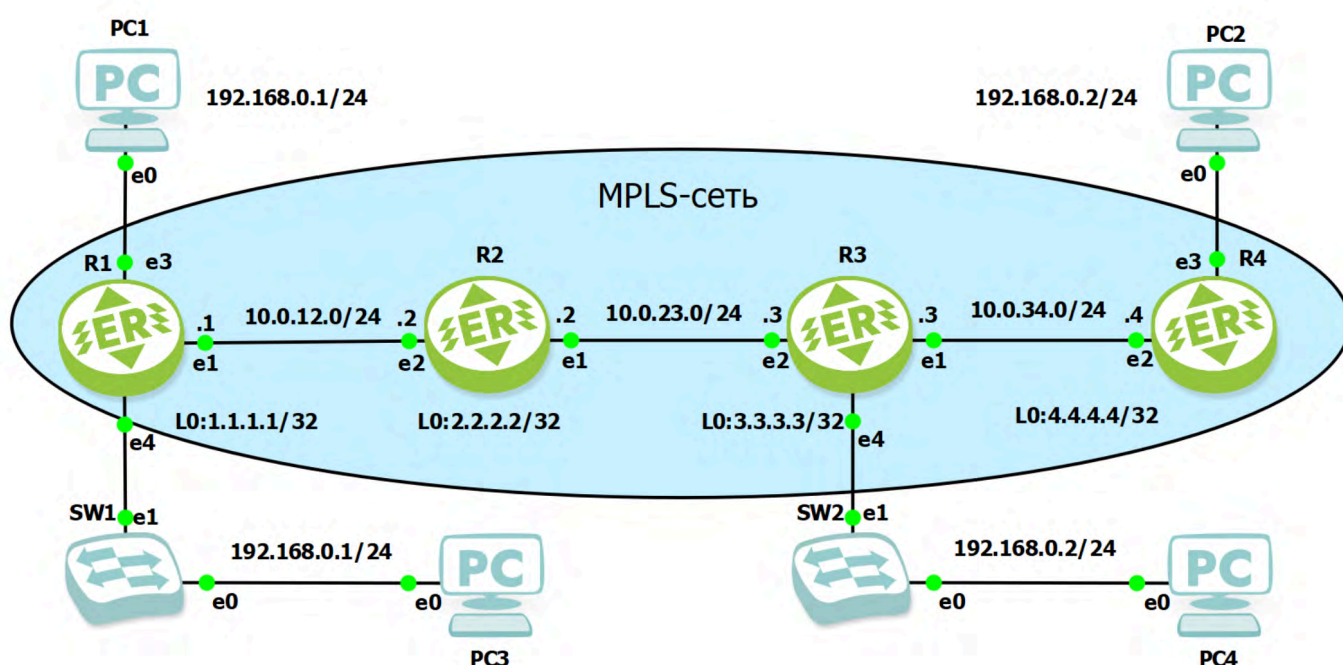


Рисунок 54

Настройте R1, как показано ниже.

```
enable configure
hostname R1

router ldp
  targeted-peer ipv4 3.3.3.3
  exit-targeted-peer-mode
  targeted-peer ipv4 4.4.4.4
  exit-targeted-peer-mode
  transport-address ipv4 1.1.1.1

router isis 1
```

```
net 10.0000.0000.0001.00

port ge1
  service-instance 4ge1
  encapsulation untagged

port ge3
  service-instance 4ge3
  encapsulation untagged

port ge4
  service-instance 4ge4
  encapsulation dot1q 120
  rewrite translate 1-to-1 1000

interface loopback.0
  ip address 1.1.1.1/32
  ip router isis 1

interface 2ge1
  label-switching
  connect port ge1 service-instance 4ge1
  ip address 10.0.12.1/24
  ip router isis 1
  ldp enable ipv4

vpls-instance INST 2
  member port ge3 service-instance 4ge3
  signaling ldp
  vpls-peer 4.4.4.4
  exit-signaling

vpls-instance INST2 3
  vpls-type tagged svlan 1000 tpid 8100
  member port ge4 service-instance 4ge4
  signaling ldp
  vpls-peer 3.3.3.3
  exit-signaling
```

Аналогичным образом настройте R2, R3 и R4.

В настройках протокола LDP в качестве транспортного адреса должен быть указан интерфейс loopback.0 данного маршрутизатора. В качестве целевых соседей `targeted-peer` необходим указывать всех соседей по VPLS: R3 и R4 — на R1, и R1 — на R3 и R4.



На маршрутизаторах R2, R3 и R4 командой `label-switching` необходимо активировать MPLS на всех интерфейсах, связывающих маршрутизаторы. Одновременно на этих же интерфейсах, а также на loopback-адресах всех маршрутизаторов, необходимо командой `ldp enable ipv4` задействовать протокол LDP, а протокол динамической маршрутизации ISIS командой `ip router isis 1`. Идентификаторы NSAP для ISIS назначаются по порядку: R2 получает `net 10.0000.0000.0002.00`, R3 — `net 10.0000.0000.0003.00`, а R4 — `net 10.0000.0000.0004.00`.

Клиентские устройства настраиваются в соответствии с их принадлежностью к одному из двух сервисов VPLS. В первом сервисе, который объединяет PC1 и PC2 в нетегированный L2-домен, компьютерам назначаются адреса 192.168.0.1/24 и 192.168.0.2/24 соответственно, а в качестве шлюза по умолчанию указываются IP-адреса друг друга. Второй сервис, предназначенный для PC3 и PC4, работает с тегированным трафиком VLAN 120; эти ПК получают те же адреса и также используют адреса друг друга в качестве шлюза по умолчанию. Соответственно, коммутаторы SW1 и SW2 должны быть настроены так, чтобы трафик от PC3 и PC4 помечался тегом VLAN 120. На сервисных интерфейсах 4ge4 на R1 и R3 тег VLAN меняется со 120 на 1000. Экземпляры INST2 3 на R1 и INST2 3 на R3 должны быть настроены для работы с тегированным трафиком с тегом 1000.

Для первого сервиса используется наименование INST и идентификатор 2. Для второго сервиса, работающего с VLAN 120, применяется наименование INST2 и идентификатор 3. Важно, чтобы значения идентификаторов совпадали на всех маршрутизаторах в рамках одного сервиса, в то время как локальные имена экземпляров VPLS могут различаться.

Данные настройки обеспечивают создание двух изолированных виртуальных частных LAN-сервисов поверх MPLS-инфраструктуры.

После применения конфигураций необходимо убедиться в установлении соседских отношений по ISIS и LDP между всеми узлами. Для активации изменений может потребоваться перезапустить протокол управления LDP с помощью команды `clear ldp session *` или перезапустить конкретные интерфейсы с помощью последовательного ввода команд `shutdown` и `no shutdown`.

Для проверки VPLS соединений выполните команду `ping 192.168.0.2` на PC1 и PC3.

При необходимости отладки воспользуйтесь командами:

- `show ldp session` — покажет LDP соседства;
- `show ldp vpls detail` — покажет подробные данные по сессиям VPLS.
- `show vpls-instance detail` — покажет подробные данные по экземпляру VPLS.

26.6.7 VPLS режим Kompella

Для реализации VPLS Kompella необходима комплексная подготовка сетевой инфраструктуры. Основой служит полнофункциональная MPLS-сеть, включающая протокол IGP (OSPF или ISIS) для обеспечения IP-связности loopback-интерфейсов всех PE-маршрутизаторов, а также LDP в качестве транспортного протокола.

Необходима настройка Multiprotocol BGP (MP-BGP) между всеми PE-устройствами. Для этого активируется адресное семейство L2VPN, которое отвечает за обмен управляющими данными о VPLS в формате NLRI. На практике для упрощения управления и масштабирования часто используются Route Reflector'ы, концентрирующие обновления BGP. Протокол должен поддерживать расширенные сообщества BGP, так как именно они несут атрибуты, управляющие логикой VPLS.

Непосредственная конфигурация сервиса на каждом PE-маршрутизаторе включает определение нескольких обязательных параметров. Глобальную идентификацию сервиса VPLS обеспечивает VPN ID. Членство в сервисе определяет атрибут Route Target, обычно имеющий одно значение для импорта и экспорта. Уникальность каждого объявления в общей BGP-таблице гарантирует Route Distinguisher. Каждому маршрутизатору в рамках домена назначается уникальный Ve ID, выполняющий роль порядкового номера сайта, а параметр Label Range резервирует блок меток MPLS, достаточный для подключения всех участников.

Клиентский трафик подключается к виртуальному коммутатору VPLS через физические или логические интерфейсы, такие как Ethernet-порты или VLAN-интерфейсы, которые привязываются к соответствующему экземпляру VPLS. Особое внимание следует уделить планированию нумерации: Ve ID должны быть не только уникальными, но и образовывать непрерывную последовательность, начиная с минимальных значений, чтобы избежать неэффективного расхода пространства меток. Размер Label Range должен закладываться с запасом для будущего расширения сервиса.

Режим Kompella обеспечивает механизм автообнаружения (Autodiscovery) участников VPLS-домена благодаря автоматическому построению полносвязной сети псевдопроводов под управлением BGP.

26.6.7.1 Пример настройки VPLS Kompella

Постройте схему в соответствии с рисунком ниже:





```
enable
configure
hostname R1

router ldp
 transport-address ipv4 1.1.1.1

router ospf 1
 network 1.1.1.1 0.0.0.0 area 0
 network 10.0.12.1 0.0.0.0 area 0

router bgp 2000
 neighbor 3.3.3.3 remote-as 2000
 neighbor 3.3.3.3 update-source 1.1.1.1
 neighbor 4.4.4.4 remote-as 2000
 neighbor 4.4.4.4 update-source 1.1.1.1
 address-family l2vpn vpls
  neighbor 3.3.3.3 activate
  neighbor 4.4.4.4 activate
 exit-address-family

port ge1
 service-instance 4ge1
 encapsulation untagged
```

```
port ge3
  service-instance 4ge3
  encapsulation untagged

port ge4
  mtu 9728
  service-instance 4ge4
  encapsulation dot1q 120
  rewrite translate 1-to-1 1000

interface loopback.0
  ip mtu 1500
  ip address 1.1.1.1/32

interface 2ge1
  ip mtu 1500
  label-switching
  connect port ge1 service-instance 4ge1
  ip address 10.0.12.1/24
  ldp enable ipv4

vpls-instance INST 2
  member port ge3 service-instance 4ge3
  signaling bgp
  ve-id 2001
  ve-range 16
  route-target 2000:2
  rd 2000:1
  exit-signaling

vpls-instance INST2 3
  vpls-type tagged svlan 1000 tpid 8100
  member port ge4 service-instance 4ge4
  signaling bgp
  ve-id 2001
  ve-range 8
  route-target 2000:3
  rd 2000:11
  exit-signaling
```

`vpls-instance INST 2` — на сервисах VPLS R1–R4 и R1–R3 использованы одинаковые имена и идентификаторы VPLS. Это допустимо и является стандартной

практикой, так как имя и идентификатор должны быть уникальными только в пределах одного маршрутизатора.

`member port ge3 service-instance 4ge3` — подключает AC (Access circuit)-порт к экземпляру VPLS.

`ve-id 2001` — идентификатор VE (VPLS Edge) должен быть уникален для каждого маршрутизатора в пределах сервиса VPLS. Для оптимального использования пространства меток рекомендуется назначать `ve-id` в виде непрерывной последовательности, начиная с минимальных значений (например, 1, 2, 3...). Это позволяет минимизировать размер резервируемого блока меток (Label Range) и избежать их неэффективного расхода.

`ve-range 16` — определяет размер резервируемого блока меток MPLS для распределённого сервиса VPLS. Значение должно быть не меньше максимального количества PE-маршрутизаторов (с учётом будущего расширения), которые будут участвовать в данном сервисе. Минимальное значение — 8.

`route-target 2000:2` — определяет членство в распределённом сервисе VPLS, указывая какие локальные экземпляры VPLS на разных маршрутизаторах принадлежат одному и тому же сервису. Использование одинакового значения `route-target` для всех участников в пределах одного сервиса VPLS обеспечивает их корректное соединение друг с другом и изоляцию от других сервисов.

`rd 2000:4` — позволяет различать информацию о экземплярах VPLS разных VPLS-сервисов в общей BGP таблице. Route Distinguisher делает каждый BGP-маршрут уникальным, предотвращая коллизии между одинаковыми адресами из разных распределённых сервисов VPLS. Рекомендуется использовать уникальный RD для каждого локального экземпляра VPLS на маршрутизаторе.

Аналогично R1 настройте R2, R3 и R4.

Задайте наименования устройств.

В настройках протокола LDP в качестве транспортного адреса должен быть указан интерфейс `loopback.0` данного маршрутизатора. Целевых соседей `targeted-peer` в режиме Kompella указывать не нужно.

В настройках `router ospf 1` на каждом маршрутизаторе должны быть указаны адреса интерфейсов, связывающих маршрутизаторы, а также адрес `loopback.0`. Остальные параметры команды `network 1.1.1.1 0.0.0.0 area 0` идентичны для каждого маршрутизатора.

На маршрутизаторах R2 и R3 должна быть настроена маршрутизация BGP. Все маршрутизаторы должны принадлежать одной автономной системе AS 2000. В команде `neighbor A.B.C.D remote-as 2000` в качестве `A.B.C.D` необходимо указать `loopback`-адрес BGP-соседа. Для R2 и R3 BGP-соседом будет R1 с адресом 1.1.1.1. В команде `neighbor 1.1.1.1 update-source A.B.C.D` необходимо также указать BGP-соседа и адрес настраиваемого маршрутизатора в качестве источника `A.B.C.D`. Каждое соседство



должно быть активировано командой `neighbor A.B.C.D activate`. На экземпляре маршрутизатора BGP должно быть включено адресное семейство L2VPN `address-family l2vpn vpls`.

На маршрутизаторах R2, R3 и R4 командой `label-switching` необходимо активировать MPLS на всех интерфейсах, связывающих маршрутизаторы. Одновременно на этих же интерфейсах, а также на интерфейсах `loopback` всех маршрутизаторов, необходимо командой `ldp enable ipv4` задействовать протокол LDP.

Настройте адреса `loopback.0` в соответствии с рисунком.

Сервисный интерфейс `4ge3` на R4 настройте как нетегированный.

На сервисном интерфейсе `4ge4` R3 настройте инкапсуляцию идентичную `4ge4` на R3.

Настройте на R4 экземпляр VPLS следующим образом:

```
vpls-instance INST 2
member port ge3 service-instance 4ge3
signaling bgp
ve-id 2002
ve-range 16
route-target 2000:2
rd 2000:4
exit-signaling
```

Настройте на R3 экземпляр VPLS следующим образом:

```
vpls-instance INST2 3
vpls-type tagged svlan 1000 tpid 8100
member port ge4 service-instance 4ge4
signaling bgp
ve-id 2002
ve-range 8
route-target 2000:3
rd 2000:13
exit-signaling
```

Экземпляр VPLS и маршрутизацию BGP на R2 настраивать не нужно, поскольку он не является граничными PE (Provider Edge) маршрутизаторами сети MPLS.

Настройте адрес и шлюз по умолчанию на PC1, PC2, PC3 и PC4:"

- PC1 и PC3 : 192.168.0.1 255.255.255.0 gateway 192.168.0.2
- PC2 и PC4 : 192.168.0.2 255.255.255.0 gateway 192.168.0.1

Данные настройки обеспечивают создание двух изолированных виртуальных частных LAN-сервисов поверх MPLS-инфраструктуры.

После применения конфигураций необходимо убедиться в установлении соседских отношений по OSPF и LDP между всеми узлами. Для активации изменений может потребоваться перезапустить протокол управления LDP с помощью команды `clear bgp *` или перезапустить конкретные интерфейсы с помощью последовательного ввода команд `shutdown` и `no shutdown`.

Для проверки VPLS соединений выполните команду `ping 192.168.0.2` на PC1 и PC3.

При необходимости отладки воспользуйтесь командами:

- `show ldp session` — покажет LDP соседства;
- `show bgp summary` — покажет BGP соседства;
- `show bgp l2vpn` — покажет запущенные локальные экземпляры VPLS;
- `show vpls-instance detail` — покажет подробные данные по экземплярам VPLS.

26.7 RSVP-TE

Протокол RSVP-TE (Resource Reservation Protocol – Traffic Engineering) работает в режиме распределения меток по запросу от нижележащего узла (downstream-on-demand). Запрос на выделение метки для LSP (Label Switch Path) передаётся в сообщении Path, которое отправляется из входящего (ingress) маршрутизатора (iLER). Сами метки распределяются, начиная с конечной точки (egress маршрутизатора, eLER), с использованием сообщений Resv. Сообщение Resv передаётся последовательно (hop-by-hop) от egress-узла к ingress-узлу. Важно отметить, что сообщение Path может как содержать явно заданный маршрут (explicit route) и запрос на резервирование определенной полосы пропускания, так и не содержать их.

Следует подчеркнуть, что RSVP-TE не является протоколом маршрутизации. Его задача — запрашивать ресурсы сети и метку для однонаправленного потока данных; то есть, он резервирует ресурсы только в одном направлении. Для организации двунаправленного потока требуется создание двух LSP — по одному для передачи трафика в каждом направлении. В описании протокола термины «нижележащий» (downstream) и «вышестоящий» (upstream) обозначают направление потока данных в рамках LSP.



Логический туннель RSVP-TE, представляющий собой сквозное соединение между двумя точками, идентифицируется уникальным Tunnel ID. Однако этот туннель может быть реализован не одним, а несколькими физическими LSP. Каждый из этих путей имеет свой собственный уникальный идентификатор LSP ID, но при этом все они относятся к одному и тому же туннелю, то есть имеют общий Tunnel ID. Такая архитектура обеспечивает повышенную надёжность и гибкость. Например, один LSP может быть основным, а второй — резервным для обеспечения мгновенного восстановления связи в случае сбоя. Новый LSP с уникальным LSP ID автоматически создаётся при каждой перенастройке туннеля или при запуске механизма резервирования, что позволяет туннелю адаптироваться к изменениям в сети, сохраняя при этом своё логическое тождество.

Сообщение Path

При установлении LSP RSVP-TE в сообщении Path указываются требования к узлам на пути. Сообщение Path состоит из нескольких объектов:

- SESSION — однозначно идентифицирует туннель RSVP-TE. Все LSP, принадлежащие одному туннелю, имеют один и тот же объект SESSION. Он содержит IP-адрес назначения туннеля, идентификатор туннеля и расширенный идентификатор туннеля (системный IP-адрес начальной точки туннеля).
- HOP — содержит IP-адрес выходного интерфейса, используемого для отправки сообщения Path на следующий по пути (Next Hop) маршрутизатор.
- TIME_VALUES — ожидаемый интервал обновления сессии RSVP-TE.
- EXPLICIT ROUTE (ERO) — является необязательным и содержит список заданных строго (strict) или нестрого (loose) узлов перехода, которым должен следовать LSP.
- LABEL REQUEST — запрашивает метку для LSP.
- SENDER_TEMPLATE — объекты SESSION и SENDER_TEMPLATE однозначно идентифицируют отдельный LSP. SENDER_TEMPLATE содержит IP-адрес системы-отправителя и идентификатор LSP. Каждый LSP в туннеле RSVP-TE имеет уникальный идентификатор LSP.
- SENDER_TSPEC — содержит параметры управления трафиком для LSP, такие как требуемая полоса пропускания или параметры быстрой перемаршрутизации.
- RECORD ROUTE (RRO) — является необязательным и содержит список переходов, пройденных сообщением Path.

Исходный IP-адрес сообщения Path — это системный адрес головного маршрутизатора. Адрес назначения — системный адрес конечного маршрутизатора. Однако в IP-заголовке используется опция Router Alert. Это указывает маршрутизатору

следующего перехода, что он должен обработать сообщение, даже если он не является конечным пунктом назначения.

RSVP-TE может указать LSP следовать пути выбранному протоколом IGP (аналогично работе LDP), или может наложить некоторые ограничения на маршрут. Эти ограничения перечислены в ERO (объект EXPLICIT_ROUTE) как строгие (strict) или нестрогие (loose) переходы. Если ограничений нет, в сообщении Path отсутствует ERO, и сообщение следует на следующий узел по пути, проложенному IGP в сторону конечного узла.

По ходу следования сообщения Path на каждом маршрутизаторе на пути LSP создаётся объект PSB (Path State Block). PSB содержит критически важную информацию о прямом направлении туннеля, получаемую из сообщений Path. Каждое новое сообщение Path обновляет информацию в PSB. При обработке сообщений Resv, PSB позволяет однозначно определить, какому LSP принадлежит данное сообщение и какому вышестоящему соседу его переслать.

Сообщение Resv

Сообщение Resv используется для присвоения меток и подтверждения резервирования ресурсов на узлах вдоль LSP. Сообщение Resv состоит из следующих объектов:

- **SESSION** — однозначно идентифицирующий туннель RSVP-TE, как и в сообщении Path. Он содержит IP-адрес назначения туннеля, идентификатор туннеля и расширенный идентификатор туннеля (системный адрес начальной точки туннеля).
- **HOP** — содержит IP-адрес выходного интерфейса, через который сообщение Resv отправляется на следующий вышестоящий маршрутизатор.
- **TIME_VALUE** — ожидаемый интервал обновления сессии RSVP-TE.
- **STYLE** — определяет стиль резервирования ресурсов: разделяемое Shared Explicit (SE) или фиксированное Fixed Filter (FF).
- **FLOW_SPEC** — соответствует SENDER_TSPEC из сообщения Path. Содержит параметры управления трафиком, такие как зарезервированная для LSP полоса пропускания.
- **FILTER_SPEC** — соответствует SENDER_TEMPLATE сообщения Path. Содержит IP-адрес системы отправителя и идентификатор LSP.
- **LABEL** — метка LSP, передаваемая этим маршрутизатором маршрутизатору следующего перехода.
- **RECORD ROUTE (RRO)** — является необязательным и содержит список переходов, пройденных сообщением Path. В сообщении Resv поле RRO также содержит метку, назначенную каждому узлу.

Когда конечный узел получает сообщение Path, он создаёт PSB, но ему не нужно выделять ресурсы полосы пропускания, поскольку он является выходом для туннеля. Он выделяет метку и создаёт сообщение Resv для передачи в восходящем направлении.

В отличие от сообщения Path, которое адресовано конечному маршрутизатору, сообщение Resv адресовано следующему маршрутизатору в восходящем направлении. Адрес следующего маршрутизатора берётся из PSB. Он не поступает из RRO (помните, что это необязательное поле в сообщениях Resv и Path). В качестве значения адреса в объекте HOP и исходящего адреса сообщения Resv устанавливается IP-адрес выходного интерфейса.

При генерации сообщения конечный маршрутизатор также создаёт Resv State Block (RSB). RSB хранит сообщение Resv и метку, выделенную для LSP. Сообщение Resv отправляется следующему вышестоящему узлу в том виде, в котором оно хранится в PSB. Сообщение Resv отправляется следующему вышестоящему маршрутизатору с использованием IP-адреса из объекта HOP в PSB.

Получение сообщения Resv вышестоящим маршрутизатором инициируется выделение ресурсов и метки. Для управления резервированием создаётся RSB, а для следующего маршрутизатора — сообщение Resv. Сообщение Resv отправляется на вышестоящий маршрутизатор, используя адрес из объекта HOP, хранящегося в PSB.

Процесс создания RSB и назначения меток продолжается в восходящем направлении до достижения головного маршрутизатора. После чего LSP создан и готов к работе.

Туннели RSVP-TE и LSP

С туннелем RSVP-TE может быть связано несколько LSP. Вторичный (запасной) путь использует отдельный путь LSP, но принадлежит одному и тому же туннелю RSVP-TE. Если в туннеле RSVP-TE несколько LSP, каждый LSP фактически является отдельным LSP и управляется отдельно.

Идентификатор туннеля RSVP-TE содержится в объекте SESSION. Путь LSP определяется идентификатором LSP. В сообщении Path идентификатор LSP содержится в объекте SENDER_TEMPLATE. В сообщении Resv идентификатор LSP содержится в объекте FILTER_SPEC.

Сообщения PathTear и ResvTear

Сообщения PathTear и ResvTear используются для явного сброса параметров резервирования LSP. Сообщение PathTear передаётся вниз по цепочке и маршрутизируется по тому же пути, что и соответствующее сообщение Path. То есть, PathTear адресовано конечному маршрутизатору, и в нём установлена опция Router Alert, что позволяет обрабатывать его на каждом узле. Адрес источника — это адрес головного маршрутизатора, даже если сообщение отправлено промежуточным узлом. ResvTear передаётся вверх по цепочке и маршрутизируется по тому же пути, что и соответствующее сообщение Resv.



Сообщения PathTear или ResvTear также инициируются истечением таймера состояния пути или резервирования на любом маршрутизаторе. Возьмём цепочку маршрутизаторов, объединённых RSVP-TE туннелем: R1—R2—R3—R4. Если между R2 и R3 произойдёт потеря соединения, это приведёт к тому, что R2 не получит сообщения Resv от R3. Состояние Resv истекает по таймеру, и R2 отправляет сообщение ResvTear в восходящем направлении. Аналогично, состояние Path истекает по таймеру на R3, и R3 отправляет сообщение PathTear в нисходящем направлении для сброса состояния LSP. Если сообщения PathTear или ResvTear не будут успешно доставлены другим узлам, состояние LSP в конечном итоге истечёт по таймерам на других узлах.

RSVP-TE может использовать приветственные сообщения для поддержания смежности. Когда маршрутизатор R2 обнаруживает, что смежность MPLS с маршрутизатором R3 нарушена, он отправляет сообщение ResvTear маршрутизатору R1. После получения сообщения ResvTear маршрутизатором R1 LSP операционно выключен.

При выключении туннеля RSVP-TE в результате получения сообщения PathTear или ResvTear соответствующие параметры резервирования на маршрутизаторах также корректируются в соответствии с действительностью. То, как именно будут скорректированы параметры резервирования, зависит от используемого стиля резервирования.

Сообщения об ошибках

Сообщения PathErr и ResvErr сообщают об ошибках RSVP-TE. Эти сообщения содержат объект ERROR, содержащий код ошибки и значение ошибки, указывающие на причину ошибки, а также на узел, где она произошла. Сообщения PathErr передаются вверх по пути к головному узлу и адресуются по каждому узлу. Сообщения ResvErr также адресуются по каждому узлу и передаются вниз по пути к конечному узлу. Важно отметить, что сообщения PathErr и ResvErr не изменяют состояние LSP ни на одном из узлов; они только сообщают о состоянии ошибки.

Если у промежуточного маршрутизатора нет маршрута к следующему узлу, передача сообщения Path не может быть завершена. PathErr отправляется в восходящем направлении к головному узлу от узла, который не может переслать сообщение Path. Головной маршрутизатор устанавливает LSP в состояние операционного сбоя и отправляет сообщение PathTear вниз по LSP, чтобы освободить связанное состояние LSP и, таким образом, освободить все зарезервированные ресурсы на вдоль пути LSP.

26.7.1 Соседства RSVP-TE и приветственные сообщения

В своей первоначальной форме протокол RSVP не использовал концепцию смежности RSVP или соседства RSVP. RSVP интересовался только состоянием своих путей, которые постоянно обновлялись. Однако интервал обновления обычно довольно велик (30 секунд по умолчанию), и может потребоваться относительно много времени,

чтобы обнаружить потерю соседа RSVP или отказ связанных LSP. Сокращение таймера обновления приводит к чрезмерным накладным расходам, поскольку маршрутизатору может потребоваться обновить большое количество LSP.

Для улучшения масштабируемости RSVP-TE и сокращения времени сходимости после сбоя в стандарт RSVP-TE (RFC 3209) были добавлены приветственные сообщения. Приветственные сообщения отслеживают состояние RSVP-TE между двумя соседними маршрутизаторами и, таким образом, косвенно, состояние LSP между ними.

Приветственные сообщения

Сообщение Hello содержит два объекта: объекты REQUEST и ACKNOWLEDGEMENT. Сообщение Hello отправляется с объектом REQUEST, а в ответ ожидается сообщение Hello с объектом ACKNOWLEDGEMENT. Каждая сторона независимо создаёт 4-байтовый идентификатор сессии. Неизменность идентификатора будет говорить о сохранении состояния RSVP-TE. Изменение идентификатора на одной из сторон будет свидетельствовать об утрате данным маршрутизатором актуального состояния RSVP-TE. Эти идентификаторы создаются в момент установления смежности и не меняются на протяжении всей сессии.

Поскольку для поддержания соседства между двумя маршрутизаторами требуется только одна Hello-сессия, которая не требует больших ресурсов (в отличие от множества сессий для LSP), сообщения Hello могут отправляться гораздо чаще, чем сообщения обновления пути. Интервал по умолчанию для сообщений Hello составляет 3 секунды. Смежность между двумя маршрутизаторами RSVP-TE считается нарушенной, когда:

- последовательные сообщения Hello не получены на интерфейсе в течение интервала Hello;
- или получено сообщение Hello, идентификатор сессии которого не соответствует установленному при создании сессии.

При потере смежности с соседним RSVP-TE состояния всех сессий RSVP-TE с соседним устройством очищаются, и сообщения PathTear и ResvTear отправляются по мере необходимости. Головной узел LSP продолжает отправлять сообщения Path, пытаясь восстановить LSP.

Механизм Refresh Reduction

Состояние каждого LSP должно обновляться на каждом маршрутизаторе посредством обмена сообщениями Path и Resv. В сети с сотнями или даже тысячами LSP это может привести к значительным накладным расходам на обмен управляющими данными.

RFC 2961 («Расширения для сокращения накладных расходов на обновление RSVP») определяет расширения RSVP, позволяющие сократить количество сообщений,

необходимых для обновления LSP. В нём определено несколько новых объектов и сообщений RSVP, включая объект MESSAGE_ID и сообщение Summary Refresh.

Когда интерфейс RSVP-TE настроен на сокращение частоты обновлений, он добавляет MESSAGE_ID к сообщениям Path и Resv, отправляемым через этот интерфейс. Маршрутизатор также устанавливает младший бит поля Flags в заголовке RSVP-TE, чтобы сообщить соседу о возможности сокращения частоты обновлений.

Когда интерфейсы на обеих сторонах смежности настроены на сокращение частоты обновлений, маршрутизаторы отправляют сообщение Summary Refresh со списком LSP, которые необходимо обновить, вместо множества сообщений Path и Resv. Отдельные LSP однозначно идентифицируются IP-адресом отправляющего маршрутизатора и значениями идентификатора сообщения.

Когда маршрутизатор получает сообщение Summary Refresh, он ищет в своих PSB и RSB те, для которых получены сообщения Path или Resv с соответствующим идентификатором сообщения. Затем состояние этих LSP обновляется, как если бы было получено обычное сообщение Path или Resv.

Если маршрутизатор не находит сообщение, соответствующее одному из значений идентификатора сообщения, он отправляет своему соседу сообщение Summary Refresh с объектом MESSAGE_ID_NACK, указывающим неизвестный идентификатор сообщения. Затем сосед передаёт обычное сообщение Path или Resv, соответствующее неизвестному идентификатору сообщения.

Таймеры RSVP-TE

Таймер повторных попыток `retry-timer` устанавливается на отдельных LSP и управляет интервалом, с которым маршрутизатор пытается установить LSP. Если работающий LSP выходит из строя и не восстанавливается, маршрутизатор попытается установить LSP с интервалом, заданным таймером повторных попыток.

Значение таймера повтора по умолчанию составляет 30 секунд, с возможным диапазоном от 1 до 600. Параметр `retry-limit` также устанавливается на отдельном LSP и определяет, сколько раз маршрутизатор будет пытаться установить LSP, если он не работает. Значение по умолчанию — ноль, что означает отсутствие ограничений; маршрутизатор будет продолжать попытки установить LSP бесконечно, каждый раз после истечения `retry-timer`.

26.7.2 Выбор пути в RSVP-TE

Одна из основных причин использования RSVP-TE вместо LDP заключается в том, что он предоставляет возможность выбора путей, не соответствующих IGP. Ключевым компонентом выбора пути в RSVP-TE является объект EXPLICIT_ROUTE (ERO), который передаётся в сообщении Path. ERO содержит список IP-адресов, которые определяют



некоторые или все узлы, которые должен пройти LSP.

Методы выбора пути делятся на три основные категории:

- Путь полностью определяется IGP без ограничений.
- Путь определяется IGP, но с учётом того, что он должен проходить через определённые узлы.
- Путь подчиняется ограничениям, связанным с управлением трафиком, таким как количество переходов, административные группы или требования к пропускной способности.

Выбор пути с помощью IGP

В простейшем случае LSP RSVP-TE следует по пути, выбранному IGP. В этом случае в сообщении Path отсутствует объект ERO. Головной маршрутизатор выбирает следующий нижележащий маршрутизатор на основе следующего перехода для FEC из таблицы маршрутизации и отправляет сообщение Path через интерфейс, подключённый к этому маршрутизатору. Следующий нижележащий узел повторяет процесс, и сообщение Path передаётся вниз по пути узел за узлом на конечный маршрутизатор. Обратите внимание, что путь сначала определяется в конструкции `rsvp-path`, а затем упоминается в конфигурации туннеля `rsvp-trunk` LSP с помощью ключевого слова `primary`. Тот факт, что путь пуст, указывает на отсутствие ограничений и определение пути по IGP.

Выбор пути со строгим и нестрогим заданием узлов

LSP может следовать пути, построенному IGP, но быть ограниченным необходимостью прохождения через определённые узлы. Эти узлы задаются как строгие (`strict`) или нестрогие (`loose`) и содержатся в ERO. Путь может содержать разнородный состав строгих и нестрогих переходов. Если переход задан строго, это означает, что сообщение Path должно быть немедленно переслано по указанному адресу. Если запись перехода нестрогая, сообщение Path направляется к указанному переходу с помощью IGP и может пройти через несколько промежуточных узлов.

Ingress маршрутизатор формирует ERO, содержащий полный предписанный маршрут LSP. Маршрутизатор изучает первый элемент в ERO, и если первый элемент совпадает с адресом данного маршрутизатора этот элемент удаляется из ERO, и маршрутизатор изучает следующий элемент, чтобы определить, куда отправить сообщение Path далее.

Если маршрутизатор получает Path, в котором первая запись в объекте ERO является строгой, он пытается перенаправить сообщение Path по адресу из записи, своему непосредственному соседу. Если соседа с таким адресом нет, маршрутизатор возвратит вышестоящему узлу ошибку PathErr. Если же первая запись нестрогая, то сообщение Path направляется на указанный адрес с помощью IGP и по пути может пройти через несколько промежуточных узлов.

Если сообщение Path не достигло конечной точки после прохождения всех заданных переходов, оно направляется на конечную точку с помощью IGP, как если бы это был нестрогий переход.

****Выбор пути с ограничениями управления трафиком**

Помимо ограничений, накладываемых строгими и нестрогими переходами, путь может также подчиняться другим ограничениям, связанным с управлением трафиком. Эти ограничения — количество переходов, административные группы и требования к пропускной способности.

Для LSP с управлением трафиком головной маршрутизатор выполняет расчёт CSPF, чтобы найти путь, соответствующий ограничениям. Если такового нет, LSP остаётся в рабочем состоянии. Если найден путь от головного до конечного маршрутизатора, соответствующий всем ограничениям, этот путь помещается в ERO как список строгих переходов, и сообщение Path отправляется с этим ERO.

Алгоритм CSPF

Маршрутизация на основе ограничений с использованием OSPF и IS-IS подразумевает передачу дополнительной информации, собранной в базе данных управления трафиком (TED, traffic-engineering database). В отличие от обычной информации о состоянии каналов в базе данных о состоянии каналов (LSDB), информация в TED не используется для построения таблицы пересылки IP-адресов. Информация об управлении трафиком используется только при необходимости использования TE-LSP. В этом случае алгоритм поиска кратчайшего пути с ограничениями (CSPF) работает на TED, чтобы найти кратчайший маршрут к месту назначения, соответствующий заданным ограничениям.

Алгоритм CSPF очень похож на обычный алгоритм SPF (shortest path first), за исключением того, что при расчёте CSPF любые соединения в TED, не соответствующие заданным ограничениям, игнорируются. Можно считать, что они отбрасываются, хотя они по-прежнему хранятся в TED, просто не используются при расчёте маршрута на основе ограничений. Ограничения основаны на доступной незарезервированной полосе пропускания, административных группах, количестве переходов или требовании включить или обойти определенный узел или канал.

База данных управления трафиком

Протоколы OSPF и IS-IS были улучшены схожим образом для поддержки маршрутизации на основе ограничений. Усовершенствования OSPF в области управления трафиком описаны в RFC 3630, а IS-IS — в RFC 3784. Эти усовершенствования описывают дополнительную информацию о состоянии канала, передаваемую протоколами маршрутизации. Эта информация собирается в базе данных управления трафиком (TED). Каждый маршрутизатор, поддерживающий управление трафиком, поддерживает две базы данных.



В то время как база данных состояния канала (LSDB) содержит метрику канала, TED содержит дополнительную информацию, связанную с каналом. Как для OSPF, так и для IS-IS, важная информация об управлении трафиком передается в sub-TLV-записях в пакетах состояния соединения:

- Метрика управления трафиком — может совпадать с метрикой состояния соединения или отличаться от неё.
- Максимальная пропускная способность — реальная пропускная способность канала в байтах в секунду.
- Максимальная резервируемая пропускная способность — объем пропускной способности, который может быть зарезервирован для данного канала. Он может превышать максимальную пропускную способность.
- Незарезервированная пропускная способность — ещё не зарезервированная пропускная способность. Пропускная способность может быть зарезервирована с одним из восьми различных уровней приоритета. Объем незарезервированной пропускной способности определяется для каждого уровня приоритета. Начальным значением для всех восьми уровней приоритета является максимальная резервируемая пропускная способность.
- Административная группа — 32-битная маска, назначаемая администратором сети. Когда бит установлен в единицу, это указывает на членство в группе. Младший значащий бит называется группой 0, а старший значащий бит — группой 31. Канал может быть членом нескольких групп (установлено несколько битов). Административная группа также называется классом ресурсов или цветом.

IGP Shortcuts

MPLS — это технология туннелирования, которая может применяться и в других частях IP-сети, где туннелирование уместно. Одна из таких ситуаций — отправка обычного IP-трафика через туннель MPLS вместо использования стандартного механизма пересылки IP-трафика.

IGP Shortcuts в RSVP-TE — это функция интеграции, которая позволяет маршрутизаторам воспринимать настроенные туннели RSVP-TE (LSP) как виртуальные каналы связи в своей обычной таблице маршрутизации IGP. Без этого механизма LSP существует изолированно, и для передачи через него трафика необходимо явно указать это, например, с помощью статического маршрута. IGP Shortcut автоматизирует этот процесс. Когда туннель активируется с функцией IGP Shortcuts, ingress-маршрутизатор туннеля начинает анонсировать его в протокол IGP как дополнительное, прямое соединения с точкой назначения туннеля.

Ключевым элементом настройки является назначение туннелю метрики, понятной для IGP. Метрика может быть задана двумя основными способами: абсолютной (absolute) или относительной (relative). Абсолютная метрика назначает туннелю фиксированное значение (например, 10), которое напрямую сравнивается с метриками других путей. Относительная метрика корректирует стоимость физического пути до цели: например, значение -5 означает, что метрика туннеля будет на 5 единиц меньше, чем лучшая доступная IGP-метрика до точки назначения через обычные интерфейсы. Это позволяет гибко управлять предпочтениями, гарантируя, что туннель будет использоваться (или, наоборот, останется резервным), даже если физическая топология со временем изменится.

26.7.3 LSP в режиме управления трафиком

Управление маршрутом LSP с помощью определения переходов

Простейший подход к управлению маршрутом LSP — использование параметра `path` маршрута LSP. Это обеспечивает механизм, очень похожий на использование статических маршрутов в IP. Маршрут можно настроить так, чтобы он следовал вручную настроенному маршруту. Это обеспечивает высокий уровень контроля, но приводит к потере гибкости. Путь может стать неспособным адаптироваться к изменениям топологии сети и может потребовать ручной перенастройки.

Путь может быть определен как строго, так и нестрого. Указывать выходной узел в пути не обязательно — сообщение «Путь» будет направлено на выход с последнего перехода, указанного в пути. Кроме того, поскольку параметр «путь» фактически представляет собой ограничения LSP, можно использовать один и тот же путь для определения нескольких LSP, даже если они не используют один и тот же выход.

Если путь LSP содержит определённые переходы, головной маршрутизатор создаёт объект EXPLICIT_ROUTE (ERO) в сообщении Path, содержащем ограничения для этого пути.

Нестрогий переход означает, что путь должен проходить через указанный узел, но будет следовать наилучшему маршруту, выбранному IGP для достижения этого пункта. Строгий переход означает, что узел должен быть следующим непосредственным нижележащим переходом. Путь со строгими переходами по всем узлам от входного к выходному фактически не требует IGP, поскольку все переходы указаны. Однако, если строгий переход не является следующим нижестоящим переходом, LSP будет неработоспособен.

26.7.4 Дополнительные способы управления выбором пути

Существуют дополнительные инструменты, которые можно использовать в сочетании с расчётом CSPF для влияния на выбор пути:

- Указание ограничения числа переходов для LSP;
- Использование метрик TE для управления выбором маршрута;
- Использование административных групп.

Для всех этих методов управления трафиком для LSP должен быть включён CSPF. CSPF используется для расчёта маршрута, включаемого в ERO как последовательность строгих переходов. Затем по этому пути должно быть передано сообщение Path.

Указание ограничения числа переходов

Одним из инструментов управления трафиком является возможность указать максимальное количество переходов для пути. Требуемый путь для LSP не обязательно является наилучшим по выбору IGP, но удовлетворяет требованию максимума переходов. Путь можно выбрать, указав ограничение количества переходов в LSP. Ограничение количества переходов фактически указывает количество маршрутизаторов на пути, а входной и выходной LER-маршрутизаторы включаются в количество переходов.

Настройка TE-метрик

Управление трафиком также позволяет настроить отдельную TE-метрику на интерфейсе. Этим можно повлиять на выбор маршрута CSPF. Например, чтобы избежать некоторый маршрут, можно присвоить интерфейсам на пути его следования высокие TE-метрики и CSPF выберет иной маршрут с более низкими метриками.

Использование административных групп

Административные группы (admin groups) обеспечивают дополнительный уровень контроля над выбором сетевых соединений в алгоритме CSPF. Административные группы, также известные как "раскраска соединений", назначаются сетевым администратором и могут использоваться для категоризации сетевых интерфейсов различными способами.

Атрибуты административной группы хранятся в виде 32-битной маски в OSPF Type 10 LSA. Биты в маске устанавливаются в зависимости от того, к какой административной группе относится интерфейс. Интерфейс может быть назначен нескольким административным группам, одной административной группе или ни одной (по умолчанию — ни одной). Хотя этот метод часто называют "раскраской соединений", назначение фактически производится на интерфейс (аналогично метрике TE).

Таким образом, соединение может быть назначено одному набору административных групп на одном конце и другому набору на другом конце.



После назначения интерфейсов административным группам можно определить LSP так, чтобы они включали или исключали одну или несколько групп администраторов. Если LSP включает административную группу, все интерфейсы, используемые для LSP, должны быть назначены этой группе. Если LSP исключает административную группу, ни одно соединение, назначенное этой группе или группам, не может быть использован для LSP.

Необходимо соблюдать осторожность при определении и использовании административных групп. Если не удастся найти путь, соответствующий ограничению, LSP не будет создан.

26.7.5 Настройка RSVP-TE

Настройка RSVP-TE осуществляется:

- в контексте маршрутизации `router rsvp`,
- в контексте отдельных интерфейсов,
- в контексте туннеля RSVP `rsvp-trunk`,
- в конструкции `rsvp-path` задаются явные маршруты.

Для простой настройки RSVP-TE достаточно:

- Включить маршрутизацию RSVP `router rsvp`. При этом можно оставить маршрутизацию без каких-либо настроек.
- Настроить маршрутизацию OSPF.
- На интерфейсах включить: `label-switching` и `enable-rsvp`, задать полосу пропускания `bandwidth 1g`.
- На входном (ingress) и выходном (egress) маршрутизаторах настроить туннель `rsvp-trunk`.
- При необходимости, задать и подключить к туннелю явные маршруты `rsvp-path`.

26.7.5.1 Команды в контекстах `router rsvp` и интерфейсов

Команды сокращения количества обновлений (Refresh Reduction):

Команды вводятся в контексте `router rsvp` и тогда действуют для всех соседей, или вводятся в контексте отдельного интерфейса (с префиксом `rsvp`) и действуют для всех соседей, обнаруженных через этот интерфейс.



- `ack-wait-timeout <1-65535>` — таймер ожидания подтверждения (ACK) получения Summary Refresh сообщений. Значение по умолчанию равно 10 секундам.
- `message-ack` — включает отправку запросов на подтверждение получения для всех сообщений Summary Refresh, отправляемых соседям данного маршрутизатора. Сообщения Summary Refresh отправляются только тем соседям, про которых известно, что они поддерживают механизм Refresh Reduction. Выключена по умолчанию.
- `no refresh-reduction` — выключает механизм Refresh Reduction. Механизм Refresh Reduction включён по умолчанию.
- `bundle-send` — включает отправку пакетных сообщений. Несколько отдельных сообщений RSVP упаковываются в один "контейнер" — Bundle Message. Выключена по умолчанию. Доступна только в контексте `router RSVP`.

Команды сокращения обработки сообщений:

Команды вводятся в контексте `router RSVP`.

- `refresh-path-parsing` / `no-refresh-path-parsing` — включает/отключает анализ сообщений Refresh-Path, полученных от вышестоящих узлов. В выключенном режиме `no-refresh-path-parsing` сообщение Refresh-Path будет переслано нижележащему маршрутизатору без подробного анализа. Это позволит снизить нагрузку на процессор маршрутизатора. Выключить анализ Refresh-Path рекомендуется, когда вы уверены, что конкретному маршрутизатору не нужно анализировать сообщения Refresh-Path для проверки изменений, поскольку LSP, проходящие через этот маршрутизатор, не обязательно должны обновляться одновременно (за один проход сообщения Path) по всей цепочке. Команда включена по умолчанию.
- `refresh-resv-parsing` / `no-refresh-resv-parsing` — Аналогична предыдущей команде. Используйте команду `no-refresh-resv-parsing` чтобы минимизировать обработку Refresh-Resv сообщений, полученных от вышестоящих узлов. Команда включена по умолчанию.

Команды управления приветственными сообщениями:

Команды вводятся как в контексте `router RSVP`, так и в контексте отдельного интерфейса с префиксом `RSVP`. Команды на интерфейсе перекрывают действие команд, заданных глобально.



- `hello-interval <1-65535>` — включает отправку приветственных сообщений и устанавливает значение интервала в секундах между ними. Значение по умолчанию равно трём секундам.
- `hello-receipt` — включает приём приветственных сообщений от RSVP-соседей.
- `hello-timeout <1-65535>` — если LSR не получил приветственное сообщение от соседа в течение заданного этой командой времени (в секундах), все сессии, используемые совместно с этим соседом, сбрасываются. Параметр `hello-timeout` определяет, как долго узел RSVP ожидает приветственного сообщения, прежде чем объявить соседа отключённым. Значение должно быть в 3.5 раза и более выше значения `hello-interval`.
- `neighbor { A.B.C.D | X:X::X:X }` — позволяет назначить адрес соседа, который будет использоваться при обмене приветственными сообщениями. Любое приветственное сообщение от соседа, которое не идентифицировано явно, будет отклонено. Можно задать несколько соседей несколькими командами `neighbor`. Команда вводится только в контексте `router rsdp`.

Команды настройки срока резервирования LSP:

`refresh-time` и `keep-multiplier` работают в связке и определяют, как долго будут существовать LSP в условиях отсутствия обновляющих сообщений и как долго для него будут резервироваться ресурсы. Команды вводятся как в контексте `router rsdp`, так и в контексте отдельного интерфейса с префиксом `rsdp`.

- `refresh-time <1-65535>` — определяет интервал отправки обновляющих сообщений RSVP — как часто (в секундах) входной узел (Ingress) передаёт сообщение Path, а выходной (egress) узел передаёт сообщение Resv. Значение по умолчанию составляет 30 секунд.
- `keep-multiplier <1-255>` — задаёт множитель для расчёта допустимого срока резервирования ресурсов для LSP. Значение по умолчанию равно трём. Используйте следующую формулу для расчёта срока резервирования LSP: $L \geq (K + 0,5) \cdot 1,5 \cdot R$, где $K = \text{keep-multiplier}$, а $R = \text{refresh-time}$. Когда срок резервирования истекает LSP удаляется.

Прочие команды:

Команды вводятся в контексте `router rsdp`.

- `cspf / no-cspf` — включает/отключает работу алгоритма CSPF (Constrained Shortest Path First) для всех путей RSVP. Включена по умолчанию. Алгоритм CSPF вычисляет пути для LSP с учётом каналов связи и различных ограничений, таких как пропускная способность, количество переходов, административные группы,



приоритет и явные маршруты. Отключение CSPF приведёт к тому, что путь будет прокладываться алгоритм SPF, который всегда будет предлагать кратчайший маршрут не зависимо от ограничений. Объект EXPLICIT ROUTE (ERO) удаляется из сообщения Path, заданные ограничения игнорируются.

- **explicit-null** — включает отправку явной нулевой метки напрямую присоединённым выходным узлам вместо реализации режима PHP и отправки выходным узлам пакетов без метки.
- **php / no-php** — включает/отключает работу алгоритма Penultimate Pop Hopping для данного маршрутизатора.
- **from { A.B.C.D | X:X::X:X }** — позволяет вручную задать интерфейс loopback в качестве идентификатора маршрутизатора в LSP.
- **local-protection** — глобально (для всех сессий RSVP) включает функцию локального восстановления (переключения на запасной путь) для тех LSP, для которых данный маршрутизатор является транзитным.
- **loop-detection / no-loop-detection** — включает/отключает механизм обнаружения петель маршрутизации.

26.7.5.2 Команды в контексте туннеля **rsvp-trunk**

- **primary {...}** — конфигурация первичного (primary) пути LSP. Параметры команды будут рассмотрены ниже.
- **secondary {...}** — конфигурация вторичного (secondary) пути LSP. Параметры команды будут рассмотрены ниже.
- **map-route {A.B.C.D | X:X::X:X}** — определяет подсеть, в которую через туннель будет допущен трафик. Возможно задать несколько подсетей несколькими командами.
- **from {A.B.C.D | X:X::X:X}** — задаёт входящий адрес туннеля на входном маршрутизаторе.
- **to {A.B.C.D | X:X::X:X}** — задаёт исходящий адрес туннеля на выходном маршрутизаторе.
- **rsvp-trunk-restart** — перезапускает RSVP туннель.
- **ext-tunnel-id {A.B.C.D | X:X::X:X}** — задаёт расширенный идентификатор туннеля в виде IP-адреса. Идентификатор прописывается в RSVP сообщения в объекте SESSION. Если расширенный идентификатор не задан, используется идентификатор маршрутизатора. Расширенный идентификатор туннеля — это



простой способ идентификации всех LSP, принадлежащих одному и тому же туннелю.

- `update-type {make-before-break | break-before-make}` — задаёт тип переключения между LSP при изменении параметров RSVP-TE:
 - `make-before-break` (по умолчанию) — создаётся новый LSP, и только после этого удаляется старый. В этом случае RSVP должен использовать стиль резервирования Shared Explicit (SE), чтобы оба LSP могли делить одну и ту же полосу.
 - `break-before-make` — старый LSP удаляется (PathTear/ResvTear), трафик прерывается, создаётся новый LSP, трафик возобновляет своё движение по новому пути. Может использоваться любой стиль резервирования.

Команды IGP Shortcut:

- `enable-igp-shortcut` / `disable-igp-shortcut` — включает/отключает использование туннеля RSVP-TE в качестве маршрута в IP-IGP-маршрутизации. При включении, туннели будут участвовать в вычислении протоколом IGP кратчайшего пути наряду с обычными IP-маршрутами.
- `lsp-metric absolute <1-65535>` — задаёт абсолютное значение метрики для RSVP-TE туннеля при вычислении кратчайшего пути IGP. Не зависимо от метрики подлежащего IGP-пути будет установлено заданное значение.
- `lsp-metric relative {<-65535-0> | <1-65535>}` — устанавливает относительное значение метрики для RSVP-TE туннеля при вычислении кратчайшего пути IGP. Значение будет складываться со значением метрики подлежащего IGP-пути. Возможны отрицательные и положительные значения.

Параметры команд `primary` и `secondary` :

- `path <PATHNAME>` — подключает ограничения (обязательные и свободные узлы для прохождения LSP) заданные в контексте команды `rsvp-path`.
- `affinity` / `no-affinity` — включает/отключает отправку объектов атрибутов сессии (SESSION ATTRIBUTE в сообщениях Path) с данными о привязке ресурсов. Параметр включён по умолчанию.
- `exclude-any <ADMIN-GROUP-NAME>` — исключает интерфейсы с указанными в административной группе атрибутами из построения TE-LSP.
- `include-any <ADMIN-GROUP-NAME>` — включает интерфейсы с указанными в административной группе атрибутами в построение TE-LSP.

- **bandwidth <1-10000000000 bits>** — определяет зарезервированную для LSP полосу пропускания. Можно указать более крупные единицы измерения: k, m, g. Значение по умолчанию равно нулю, что позволяет передачу данных, но не резервирует полосу.
- **cspf / no-cspf** — включает/отключает использование алгоритма CSPF для вычисления объекта явного пути (ERO) к выходному маршрутизатору. Сервер CSPF вычисляет пути для LSP с учётом таких ограничений, как пропускная способность, количество переходов, административные группы, приоритет и явные маршруты. CSPF стремится минимизировать перегрузку, разумно балансируя нагрузку на сеть. Если CSPF отключён глобально, его невозможно включить для отдельного LSP.
- **cspf-retry-limit <1-65535>** — определяет количество повторных попыток CSPF вычислить маршрут для данного LSP. Значение по умолчанию равно нулю.
- **cspf-retry-timer <1-600>** — определяет интервал в секундах между попытками CSPF вычислить маршрут для данного LSP. Значение по умолчанию равно нулю.
- **filter {fixed | shared-explicit}** — определяет стиль резервирования ресурсов: разделяемое Shared Explicit (SE) или фиксированное Fixed Filter (FF). При SE для всех LSP выделяется общая полоса пропускания, которую они делят. При FF каждый LSP получает отдельную фиксированную полосу пропускания. Значение по умолчанию — **fixed**.
- **hold-priority <0-7>** — задаёт приоритет удержания ресурсов уже установленным LSP. Вновь устанавливаемые туннели могут вытеснять имеющийся LSP из полосы пропускания. 0 — высший приоритет (почти невозможно вытеснить). 7 — низший приоритет (легко вытеснить). Значение по умолчанию равно нулю.
- **setup-priority <0-7>** — задаёт приоритет установления LSP в случае нехватки полосы пропускания. Определяет способность LSP вытеснять существующие LSP для освобождения полосы пропускания. 0 — высший приоритет (максимальная способность вытеснять). 7 — низший приоритет (не способен вытеснять). Значение по умолчанию равно семи.
- **hop-limit <1-255>** — определяет максимальное количество переходов для LSP, что позволяет ограничить длину пути.
- **local-protection** — включает для данного LSP функцию локального восстановления (переключения на запасной путь) для тех LSP, для которых

данный маршрутизатор является транзитным.

- **record / no-record** — включает/отключает запись пройденных сообщениями Path и Resv узлов. Запись ведётся в объект Route Record Object (RRO) для подтверждения установления резервирования и выявления ошибок. Параметр включён по умолчанию.
- **label-record** — включает механизм записи меток MPLS, которые распределяются между маршрутизаторами в процессе установления LSP. Запись ведётся в объект RRO сообщения Resv. Запись меток может быть полезна при поиске неисправностей в сети.
- **retry-limit <1-65535>** — определяет количество повторных попыток пересоздать LSP, когда входной узел пытается настроить LSP и настройка завершается неудачей из-за получения сообщения об ошибке пути. Значение по умолчанию равно нулю, что означает бесконечные попытки.
- **retry-timer <1-600>** — определяет интервал между попытками пересоздать LSP. Значение по умолчанию составляет 30 секунд.
- **reuse-route-record** — включает использование пути записанного в RRO из входящего Resv-сообщения, в качестве строгого (все переходы будут указаны как strict) пути ERO для следующего сообщения Path. Обеспечивает стабильность пути при отсутствии CSPF.
- **traffic {guaranteed | controlled-load}** — определяет тип сервиса (Service Type). Гарантированный сервис с гарантированной полосой пропускания и задержкой или сервис с контролируемой нагрузкой с низким уровнем потерь но без гарантированно низкой задержки. По умолчанию **controlled-load**.

26.7.5.3 Команды задания явных маршрутов

rsvp-path <PATHNAME> — вводится в конфигурационном режиме и позволяет перейти в контекст задания явных путей.

В контексте **rsvp-path** задаются адреса интерфейсов через которые должен строго (**A.B.C.D strict**) или нестрого (**A.B.C.D loose**) пройти путь LSP. Для правильной работы необходимо задавать входящие интерфейсы маршрутизаторов по ходу следования пути.

Затем по **PATHNAME** заданные ограничения (строгие и нестрогие переходы) можно подключить в параметрах туннеля **rsvp-trunk** к первичному (**primary path <PATHNAME>**) или вторичному (**secondary path <PATHNAME>**) путям LSP.

26.7.6 Конфигурирование RSVP-TE

Постройте схему в соответствии с рисунком ниже:

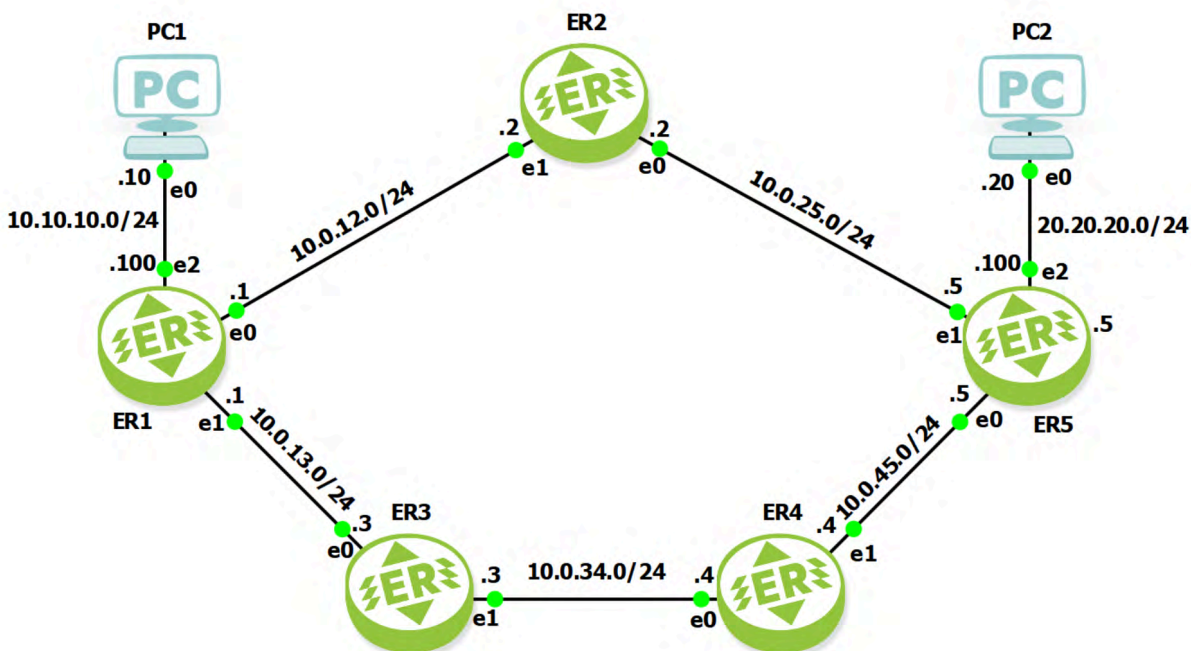


Рисунок 56

Полные конфигурации всех узлов доступны по ссылке <https://docs.ecorouter.ru/Руководство/901-Примеры-конфигураций/RSVP-TE-Strict-path>.

Настройте ER1, как показано ниже:

```
enable
configure
hostname ER1

router rsvp

router ospf 1
 network 1.1.1.1/32 area 0
 network 10.0.12.0/24 area 0
 network 10.0.13.0/24 area 0

# Задание узлов через которые строго (strict) должен следовать путь
rsvp-path RESP1
# Указываются входящие интерфейсы по направлению пути RSVP-TE
10.0.13.3 strict
10.0.34.4 strict
```

```
# Настройка туннеля RSVP-TE
rsvp-trunk TRU1 ipv4
# Подключение явного пути RESP1 в качестве вторичного пути
secondary path RESP1
# Пропускать только трафик следующий в указанную подсеть
map-route 20.20.20.0/24
# Начальная и конечная точки туннеля
from 1.1.1.1
to 5.5.5.5

port ge0
service-instance si0
encapsulation untagged

port ge1
service-instance si1
encapsulation untagged

port ge2
service-instance si2
encapsulation untagged

interface loopback.0
ip address 1.1.1.1/32

interface in0
# Включение коммутации по меткам
label-switching
connect port ge0 service-instance si0
ip address 10.0.12.1/24
# Установка пропускной способности
bandwidth 1g
# Тип интерфейса point-to-point позволяет быстрее устанавливать соседство
ip ospf network point-to-point
enable-rsvp
rsvp refresh-time 5

interface in1
label-switching
connect port ge1 service-instance si1
ip address 10.0.13.1/24
bandwidth 1g
ip ospf network point-to-point
```




```
enable-rsvp
```

```
interface in2
```

```
connect port ge2 service-instance si2
```

```
ip address 10.10.10.100/24
```

По аналогии с ER1 зеркально настройте ER5.

ER2, ER3 и ER4 настраиваются аналогично но туннель и `rsvp-path` на них настраивать не нужно.

Настройте IP-адрес и шлюз на PC1 и PC2.

Проверьте связь между PC1 и PC2 дав на PC1 команду `ping 20.20.20.20`.

Для проверки настроек RSVP воспользуйтесь командой `show rsvp`

Для проверки состояния RSVP сессии воспользуйтесь командой `show rsvp session`.

Для просмотра базы данных RSVP-TE воспользуйтесь командой `show ip ospf te-database`.

Командой `show mpls forwarding-table` проверьте установился ли туннель и удостоверьтесь, что маршрут 20.20.20.0/24 использует этот туннель.

```
ER1#show mpls forwarding-table
```

```
Codes: > - installed FTN, * - selected FTN, p - stale FTN,
```

```
       B - BGP FTN, K - CLI FTN,
```

```
       L - LDP FTN, R - RSVP-TE FTN, S - SNMP FTN, I - IGP-Shortcut,
```

```
       U - unknown FTN
```

Code	FEC	FTN-ID	Tunnel-id	Pri	LSP-Type	Out-Label	Out-Intf
Nexthop							
R>	2.2.2.2/32	1	5001	Yes	LSP_DEFAULT	52480	in1
10.0.13.3							
R>	2.2.2.2/32	3	5001	No	LSP_DEFAULT	52482	in1
10.0.13.3							
R>	20.20.20.0/24	2	5001	Yes	LSP_DEFAULT	52480	in1
10.0.13.3							

27 BFD

27.1 Протокол BFD

Bidirectional Forwarding Detection (BFD) — это протокол, созданный для быстрого обнаружения разрыва соединений 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").

27.1.1 Команды для настройки BFD

Команды вводимые в контекстном режиме интерфейса:

`bfd disable` Команда выключает все BFD-сессии на интерфейсе, сессии переводятся в состояние **Admin-Down**. Значение по умолчанию: **enabled**.

`bfd session {A.B.C.D E.F.G.H | X:X::X:X Y:Y::Y:Y}` Команда включает на интерфейсе BFD-сессию для конкретной пары адресов источника и назначения. Адреса могут быть как в формате IPv4 так и в формате IPv6. Таким образом может быть задано произвольное количество BFD-сессий.

`bfd interval <25-999> minrx <25-999> multiplier <3-50>` 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 разорванной. Значения по умолчанию: 250/250/3.

Команды вводимые в конфигурационном режиме:

`bfd data-plane {enable | disable}` Команда помогает избежать разрыва сессий BFD при высокой (более 80%) загрузке процессора маршрутизатора, которую можно отследить с помощью команды `show platform cpu`. Команда влияет только на вновь запущенные сессии BFD, поэтому уже активные сессии необходимо предварительно отключить, а затем включить заново после её применения.

`bfd slow-timer <1000-30000>` Команда устанавливает временной промежуток в миллисекундах между BFD-сообщениями, которые отправляются в случае разрыва BFD-сессии для возможности её восстановления.




```
bfd multihop-peer {A.B.C.D | X:X::X:X} {interval <50-999>} {minrx <50-999>}  
{multiplier <3-50>}
```

 Команда определяет соседа по multihop-сессии BFD и устанавливает таймеры аналогично команде `bfd interval`. Получаемые приветственные сообщения должны иметь TTL не более 254. В случае TTL 255 сообщение будет отброшено.

`bfd all-interfaces` Команда вводится в контекстном режиме настройки маршрутизации (`config-router`). Команда устанавливает bfd-сессии со всеми соседями в рамках соответствующего процесса маршрутизации.

`neighbor A.B.C.D fall-over bfd multihop` Команда вводится в контекстном режиме настройки маршрутизации BGP (`config-router`). Команда включает для конкретного BGP-соседа работу протокола BFD-multihop.

Начиная с версии 3.2.6.1.16715 в протоколе BFD режим **echo** **не поддерживается!**

Команды просмотра для протокола BFD на EcoRouter приведены ниже.

Показать информацию о глобальных настройках BFD:

```
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
```

- Start Time — время старта процесса oamd;
- BFD Admin State — административное состояние протокола на устройстве;
- Number of Sessions — количество активных сессий;
- Slow Timer — значение slow таймера;
- Image type — тип обработки hello-пакетов (монолитный — производится одним процессом, распределённый — 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.

Показать информацию о настройках BFD на всех интерфейсах, на которых включён этот протокол:

```
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
```

- 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 прерванной.

Показать информацию обо всех активных bfd-сессиях:

```
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
```

- Sess-Idx — локальный id сессии;
- Remote-Disc — id сессии на удалённом устройстве;
- Lower-Layer — инкапсулирующий протокол;
- Sess-Type — тип сессии (single/multi);
- Sess-State — состояние сессии;
- UP-Time — up-time сессии;
- Remote-Addr — адрес интерфейса удалённого маршрутизатора, с которым установлена сессия;
- Number of Sessions — количество активных сессий.

Показать детальную информацию обо всех активных bfd-сессиях.

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
```

- 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 удалённом 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 удалённого маршрутизатора, с которым установлена сессия;
- Remote Port — удалённый UDP-порт;
- Min Tx/Neg Tx — локальный/удалённый интервал отправки bfd-control сообщений;
- Min Rx/Neg Rx — локальный/удалённый ожидаемый интервал приёма bfd-control сообщений;
- Multiplier/Neg detect multi — количество пропущенных сообщений, после которого сессия считается прерванной. Значения на локальном и удалённом роутерах;
- Min echo Tx/Min echo Rx — локальный/удалённый интервал отправки echo-сообщений;
- Sess down time — время прерывания сессии;
- Sess discontinue time — время, на протяжении которого сессия была прервана;
- Bfd GTSM — состояние функции GTSM;
- Bfd Authentication — состоянии функции аутентификации;
- Pkt In — количество пришедших BFD-пакетов;
- Pkt Out — количество отправленных BFD-пакетов;
- Echo Out — количество отправленных echo-пакетов;
- UPTIME — up-time сессии;
- Protocol Client Info — информация о протоколе, посредством которого установлена сессия;
- Number of Sessions — количество активных сессий.

Показать информацию о сессии между конкретным локальным интерфейсом с указанием его ip-адреса и конкретным удалённым интерфейсом с указанием его ip-адреса:

```
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
```

- Session Interface Index — системный номер локального интерфейса;
- Lower Layer — инкапсулирующий протокол;
- Session State — состояние сессии;
- Session Index — локальный id сессии;
- Session Type — тип сессии (single/multi);
- Local Discriminator — локальный id сессии;
- Local Address — адрес интерфейса локального маршрутизатора, на котором установлена сессия;
- Local Port — локальный UDP-порт;
- Remote Discriminator — id сессии на удалённом устройстве;
- Remote Address — адрес интерфейса удалённого маршрутизатора, с которым установлена сессия;
- Remote Port — удалённый UDP-порт;
- Min Tx — локальный интервал отправки bfd-control сообщений;
- Min Rx — локальный ожидаемый интервал приёма bfd-control сообщений;
- Multiplier — количество пропущенных сообщений, после которого сессия считается прерванной;
- UPTIME — up-time сессии.

27.2 Пример настройки single-hop BFD-OSPF

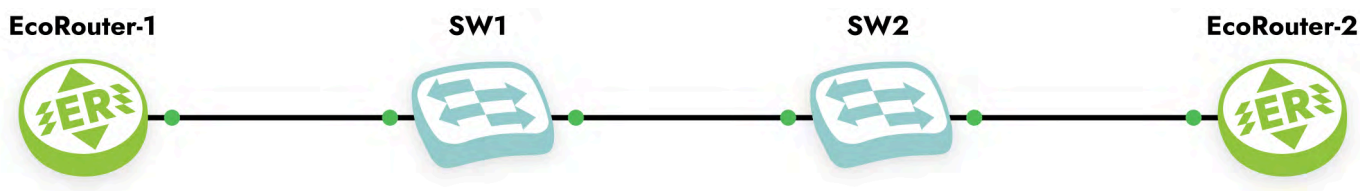


Рисунок 57

Конфигурация EcoRouter1:

Настройка интерфейсов и портов:

```
enable
configure
hostname EC01
port ge0
    service-instance 4ge0
    encapsulation untagged
    exit
interface loopback.0
    ip address 1.1.1.1/32
interface 2ge0
    ip address 10.1.1.1/24
    connect port ge0 service-instance 4ge0
    exit
```

Настройка OSPF и включение BFD:

```
router ospf 100
    ospf router-id 1.1.1.1
    network 1.1.1.1/32 area 0.0.0.1
    network 10.1.1.0/24 area 0.0.0.1
    bfd all-interfaces
```

Конфигурация EcoRouter2:

Настройка интерфейсов и портов:

```
enable
configure
hostname EC02
port ge0
    service-instance 4ge0
    encapsulation untagged
    exit
interface loopback.0
    ip address 2.2.2.2/32
interface 2ge0
    ip address 10.1.1.2/24
    connect port ge0 service-instance 4ge0
    exit
```

Настройка OSPF и включение BFD:

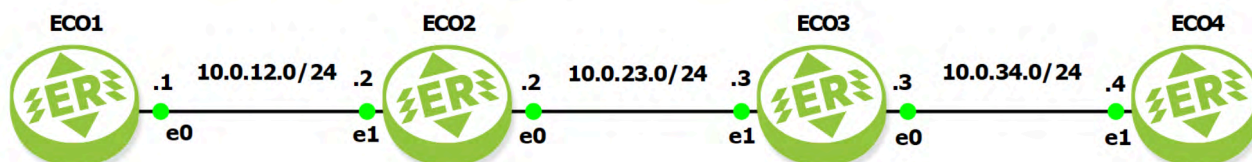
```
router ospf 100
    ospf router-id 2.2.2.2
    network 2.2.2.2/32 area 0.0.0.1
```



```
network 10.1.1.0/24 area 0.0.0.1  
bfd all-interfaces
```

27.3 Настройка Multi-Hop BFD по BGP

Постройте схему в соответствии с рисунком ниже:



Полные конфигурации всех узлов доступны по ссылке <https://docs.ecorouter.ru/Руководство/901-Примеры-конфигураций/Multi-Hop-BFD-по-BGP>.

На всех маршрутизаторах настроены порты и интерфейсы. Между соседями настроены общие подсети.

На каждом маршрутизаторе задан loopback адрес вида 1.1.1.1/32 соответственно порядковому номеру маршрутизатора.

На каждом маршрутизаторе запущен протокол ISIS 1. Каждый интерфейс каждого маршрутизатора привязан к ISIS 1. Обеспечена IP-связность между всеми интерфейсами всех маршрутизаторов.

На ECO1 и ECO4 запущен протокол BGP, между ECO1 и ECO4 установлено BGP-соседство.

Настройте ECO1 как показано ниже:

```
hostname ECO1  
  
router isis 1  
  net 49.0000.0000.0001.00  
  passive-interface loopback.0  
  
router bgp 1000  
  neighbor 4.4.4.4 remote-as 1000  
  neighbor 4.4.4.4 update-source 1.1.1.1  
  neighbor 4.4.4.4 fall-over bfd multihop  
  
port ge0  
  service-instance sige0  
  encapsulation untagged
```



```
interface loopback.0
  ip address 1.1.1.1/32

interface inge0
  connect port ge0 service-instance sige0
  ip address 10.0.12.1/24
  ip router isis 1

bfd multihop-peer 4.4.4.4 interval 100 minrx 100 multiplier 3
```

Аналогично настройте остальные маршрутизаторы.

На маршрутизаторах ECO2 и ECO3 не настраивается BGP и BFD.

BGP на ECO1 и ECO4 настраивается зеркально.

`router isis 1 > net 49.0000.0000.0001.00` : ECO2 — 0002, ECO3 — 0003, ECO4 — 0004.

Адреса loopback на каждом маршрутизаторе привязываются в настройках `router isis 1` как пассивные интерфейсы.

Ключевые команды для настройки BFD вводятся на :

- Команда `bfd multihop-peer` определяет соседа по multihop-сессии таймеры. Команда описана в первом подразделе данного раздела.
- Команда `router bgp 1000 > neighbor 4.4.4.4 fall-over bfd multihop` включает BGP-multihop соседство.

Проверить работу Multi-Hop BFD можно с помощью программы перехвата трафика.

Перехватите трафик между ECO3 и ECO4. Выберите одно из сообщений BFD Control. Перейдите в Internet Protocol Version 4 и удостоверьтесь что TTL пакета = 253. TTL = 253 говорит о том, что пакет сделал два перехода через ECO2 и ECO3 перед тем как попасть в перехват.

28 Маршрутизация Multicast

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/приёма 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 определённого сервера при IGMPv2;
- IGMP проху 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.

28.1 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 с первой по третью версии протокола.

Таблица 96 — Список команд для настройки протокола IGMP

Команда	Режим	Описание
---------	-------	----------

Команда	Режим	Описание
<code>ip igmp access-group <номер списка доступа></code>	(config-if)#	Фильтрация доступа к определённым мультикаст-группам с помощью списков доступа
<code>ip igmp immediate-leave group-list <номер списка фильтров></code>	(config-if)#	Команда сокращения времени отписки последнего клиента от группы/групп, заданных в списке фильтрации
<code>ip igmp join-group <ip-адрес></code>	(config-if)#	Команда добавления интерфейса маршрутизатора в мультикаст-группу
<code>ip igmp last-member-query-count <2-7></code>	(config-if)#	Настройка количества IGMP query сообщений, отправляемых в ответ на сообщение типа leave. По умолчанию 2
<code>ip igmp last-member-query-interval <1000-25500></code>	(config-if)#	Настройка интервала отправки IGMP query сообщений. По умолчанию 1000 мс
<code>ip igmp limit <1-2097152></code>	(config)#	Настройка ограничений количества мультикаст-маршрутов
<code>ip igmp mroute-proxy <имя интерфейса></code>	(config-if)#	Включение проксирования для мультикаст маршрутов на другой интерфейс
<code>ip igmp proxy unsolicited-report-interval <1000-25500></code>	(config-if)#	Задание значения задержки между двумя IGMP join сообщениями. По умолчанию 1000 мс
<code>ip igmp proxy-service</code>	(config-if)#	Включение режима IGMP proxy
<code>ip igmp querier-timeout <60-300></code>	(config-if)#	Задание времени до перевыборов querier маршрутизатора в сегменте в секундах
<code>ip igmp query-interval <1-18000></code>	(config-if)#	Задание частоты отправки General Query. По умолчанию 125 с

Команда	Режим	Описание
<code>ip igmp query-max-response-time <1-240></code>	(config-if)#	Задание максимального значения времени ответа на IGMP query в секундах. По умолчанию 10 с
<code>ip igmp robustness-variable <2-7></code>	(config-if)#	Задание числа для тонкой настройки IGMP сообщений. По умолчанию 2
<code>ip igmp startup-query-count <2-10></code>	(config-if)#	Задание количества query сообщений. По умолчанию 2
<code>ip igmp startup-query-interval <1-18000></code>	(config-if)#	Настройка интервала отправки IGMP query сообщений. По умолчанию 31 с
<code>ip igmp static-group <ip-адрес></code>	(config-if)#	Назначение интерфейса устройства на прослушивания определённой мультикаст-группы
<code>ip igmp version <1-3></code>	(config-if)#	Выставление версии IGMP
<code>ip igmp ssm-map {enable static <номер списка доступа>}</code>	(config)#	Включение SSM-картирования. Задание статического SSM с помощью списка доступа
<code>ip igmp tos-check</code>	(config)#	Проверка значения поля TOS. Включена по умолчанию
<code>ip igmp vrf <имя виртуального маршрутизатора> {limit <1-2097152> ssm-map enable ssm-map static <номер списка доступа>}</code>	(config)#	Команды настройки для выполнения в виртуальном маршрутизаторе
<code>ip igmp ra-option</code>	(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 нисходящем 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
```

28.2 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 приведён пример настройки:

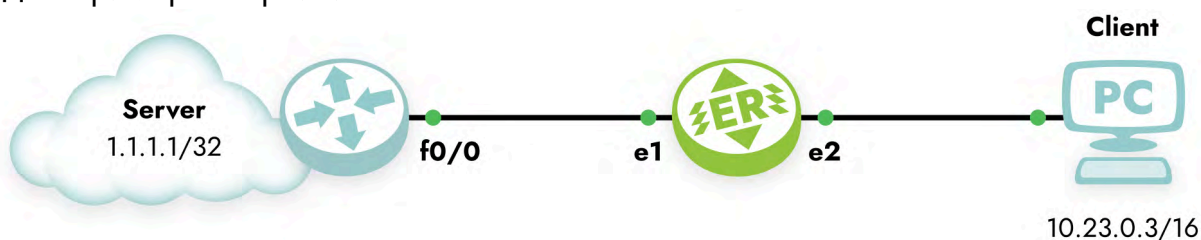


Рисунок 58

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 определённой группы.

```
ecorouter(config)#policy-filter-list 2 permit 235.7.7.7
```

Шаг 3. Включение SSM-mapping для определённой 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 мappингу `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
```

28.3 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 передаёт 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 приведён пример топологии и конфигурирования IGMP Proxy сервиса в EcoRouterOS.

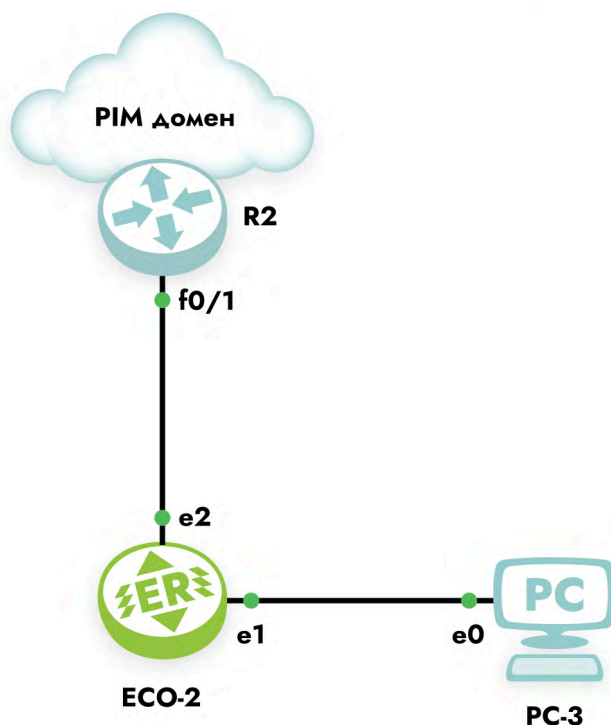


Рисунок 59

28.3.1 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».

28.4 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 вводе 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 определённой мультикастовой группе, а параметр **override** повышает приоритет статической записи о RP по сравнению с полученной динамическим путём. Динамический путь описан ниже.

Шаг 4. Добавление возможности переключения на более короткий маршрут до источника при помощи команды **ip pim spt-treshold [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 отдаётся 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 97 — Параметры команды `ip pim rp-candidate`

Параметр	Описание
<code><название интерфейса></code>	Интерфейс, назначаемый кандидатом. Интерфейс должен быть предварительно создан в системе
<code>priority</code>	Приоритет, при задании нескольких кандидатов. Чем меньше значение данного параметра, тем выше приоритет кандидата. Допустимые значения от 0 до 255. Значение по умолчанию 192
<code>group-list <POLICY-FILTER-LIST></code>	Группы, которым рассылается реклама о кандидате
<code>interval</code>	Интервал рассылки сообщений в секундах. Допустимые значения от 1 до 16383

Далее необходимо сконфигурировать рекламных агентов, которые будут рассылать 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 98 — Параметры команды `ip pim bsr-candidate`

Параметр	Описание
<code><название интерфейса></code>	Интерфейс, назначаемый рекламным агентом (BSR). Интерфейс должен быть предварительно создан в системе
<code><0-32></code>	Длина хэш-маски для расчёта хэш-значения RP. Допустимые значения от 0 до 32. Значение по умолчанию 10
<code><0-255></code>	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 приведён 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, произойдёт SPT switchover — маршрут поменяется на ECO-3 — ECO-4 — PC1.

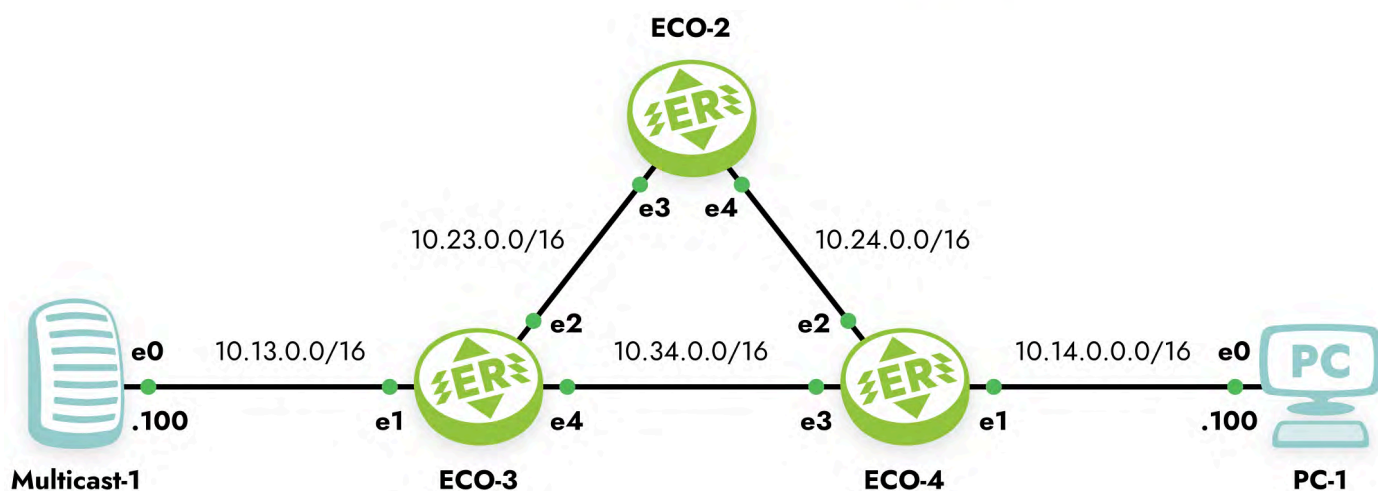


Рисунок 60

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-treshold
```

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-treshold
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.

28.4.1 Дополнительные команды конфигурирования

Таблица 99 — Дополнительные команды конфигурирования

Команда	Режим	Описание
<code>ip pim accept-register <policy-filter-list></code>	(conf)#	Указывает RP принимать Register сообщения от определённых источников
<code>ip pim cisco-register-checksum</code>	(conf)#	Опция для расчёта checksum в Register сообщениях. Для совместимости с более старыми версиями Cisco IOS

Команда	Режим	Описание
<code>ip pim ignore-rp-set-priority</code>	(conf)#	Используется для игнорирования приоритета RP, чтобы полагаться только на хеш-алгоритм
<code>ip pim jp-timer <1-65535></code>	(conf)#	Тайминг для отправки сообщений Join и Prune
<code>ip pim register-rate-limit <1-65535></code>	(conf)#	Управление количеством отправляемых Register сообщений
<code>ip pim register-rp-reachability</code>	(conf)#	Включение проверки RP доступности на маршрутизаторе (по умолчанию в конфигурации)
<code>ip pim register-source <адрес></code>	(conf)#	Задание адреса в Register сообщениях
<code>ip pim register-suppression <1-65535></code>	(conf)#	Изменение RP-keepalive-timer, если команда ip pim rp-register-kat не задана
<code>ip pim rp-register-kat <1-65535></code>	(conf)#	Изменение таймеров для мониторинга Register сообщений
<code>ip pim dr-priority</code>	(conf-int)#	Приоритет маршрутизатора для выбора DR
<code>ip pim bsr-border</code>	(conf-int)#	Пометить интерфейс как пограничный, для отмены передачи/приёма bootstrap
<code>ip pim exclude-genid</code>	(conf-int)#	Исключение опции generated ID
<code>ip pim hello-holdtime <1-65535></code>	(conf-int)#	Установка таймера holdtime для сообщений hello
<code>ip pim hello-interval <1-18724></code>	(conf-int)#	Установка таймера interval для сообщений hello
<code>ip pim neighbor-filter <policy-filter-list></code>	(conf-int)#	Установка соседств с конкретными маршрутизаторами
<code>ip pim propagation-delay <1000-5000></code>	(conf-int)#	Установка задержки распространения сообщений
<code>ip pim unicast-bsm</code>	(conf-int)#	Включение unicast bootstrap сообщений. Для совместимости с более старыми версиями Cisco IOS
<code>ip pim sparse-mode passive</code>	(conf-int)#	Включение пассивного режима
<code>ip multicast ttl-threshold <1-255></code>	(conf-int)#	Включение TTL-score мультикастового домена

Команда	Режим	Описание
<code>ip mroute <адрес подсети> <rpf сосед></code>	(conf)#	Статическая запись о подсети, в которой находится источник мультикаста

28.4.2 Команды просмотра

Таблица 100 — Команды просмотра

Команда	Описание
<code>show ip mroute</code>	Таблица мультикастовой маршрутизации
<code>show ip mvif</code>	Информация о созданных виртуальных интерфейсах, которые поддерживают мультикаст
<code>show ip rpf <адрес источника></code>	Отображение RPF информации о источнике
<code>show ip pim bsr-router</code>	Информация об BSR маршрутизаторах в домене
<code>show ip pim interface</code>	Информация об интерфейсах, на которых включена мультикастовая маршрутизация
<code>show ip pim local-members</code>	Локальная информация о запрошенных группах
<code>show ip pim mroute [detail]</code>	Детальная информация по мультикастовой маршрутизации
<code>show ip pim neighbor</code>	Информация о соседских отношениях
<code>show ip pim nexthop</code>	Информация о RP, источниках многоадресной рассылки, интерфейсах через которые получены данные
<code>show ip pim rp mapping</code>	Информация об RP в домене
<code>show ip pim rp-hash <адрес группы></code>	Информация об RP для конкретной группы
<code>show ip mroute count</code>	Вывод статистической информации

28.4.3 Команды сброса данных

```
clear ip mroute statistics <*/адрес группы>
clear ip mroute <*/адрес группы>
clear ip pim sparse-mode bsr rp-set *
```

28.5 PIM-DM и смешанный режим Sparse-Dense

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 определённых групп можно настроить обработку трафика исключительно PIM-DM логикой. Для этого используется команда `ip pim dense-group <адрес группы>`.

29 BRAS

Одним из центральных элементов сети интернет-провайдера является BRAS (Broadband Remote Access Server) — сервер широкополосного удалённого 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 удалённого доступа являются следующие:

- назначение и применение сетевых настроек на клиентском оборудовании;
- аутентификация, авторизация и выделение индивидуальных атрибутов для абонентов;
- учёт, 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 Tacsacs- или Radius-серверам. Устройства EcoRouter позволяют при запуске виртуальных сервисов на маршрутизаторе использовать как удалённые, так и локальные серверы AAA (запущенные непосредственно на самом маршрутизаторе).

Для предоставления интернет-услуг используется несколько протоколов. До недавнего времени наиболее распространённым 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 всё чаще, так как требует минимум конфигурации конечного оборудования. Технология Q-in-Q, которая является расширением стандарта IEEE 802.1Q считается наиболее безопасной. При её использовании изначально каждое конечное устройство находится в выделенном VLAN, чем гарантируется изоляция подписчиков друг от друга.

EcoRouterOS версии 3.2 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. (Подробнее о сервисных интерфейсах читайте в соответствующем разделе



документации). Для работы с IPoE и PPPoE абонентами в CLI устройства предусмотрен интерфейс со специальным наименованием `bmi` (broadband multiple instances).

29.1 IPoE абоненты

Для управления абонентскими сессиями предусмотрены конфигурируемые карты абонентов (`subscriber-map`). Под управлением понимается создание/удаление сессий, установка правил аутентификации, авторизации и аккаунтинга (AAA), настройка специфических таймеров для сессий. По правилам аутентификации, использующимся в абонентских картах, абонентские сессии различаются на статические и динамические.

Статические правила не требуют работы протокола DHCP для выдачи IP-адресов, а также настроек аутентификации и авторизации через RADIUS-сервер. По этим сессиям осуществляется только аккаунтинг на удалённых серверах. Статическая конфигурация предоставляет все необходимые настройки для выхода статических IPoE-абонентов в Интернет. Абоненты могут получать адреса по DHCP (зарезервированные адреса сконфигурированы на DHCP-сервере для определённых устройств), но при этом иметь специальные статические правила в картах. Такие типы сессий называются статическими. При срабатывании статических правил сессия моментально инициализируется и создаётся на устройстве.

Правила считаются **динамическими** в случае, если это правила тегирования (802.1Q) и настройки IP-адресов для абонентов, сформированные в процессе получения адреса по DHCP или при передаче первого пакета от абонента. Для динамических клиентов функционал аутентификации и авторизации доступен как через локальную конфигурацию на маршрутизаторе, так и через удалённый RADIUS-сервер. Такие типы сессий называются динамическими. Способ создания динамической сессии зависит от параметров команды `session-trigger` в настройках BMI интерфейса (как показано в таблице ниже).

Таблица 101 — Параметры команды `session-trigger`

Параметр команды <code>session-trigger</code>	Способ заведения динамической сессии
<code>dhcp</code>	По первому пакету DHCP Discovery от абонента (настройка по умолчанию)
<code>ip</code>	По первому IP-пакету от абонента

Таким образом, абонентская IPoE сессия может быть создана статически, по первому IP-пакету от абонента или по сообщению DHCP discover.

Каждая карта абонентов состоит из одной или нескольких последовательностей правил (**sequence**). В свою очередь, последовательность содержит одно или несколько правил (**match**) и действий (**set**). Для сопоставления абонентов и их AAA-правил используются команды `match static`, `match dynamic` и `set`. Ключевые слова **static** и **dynamic** внутри карты абонента определяют статический и динамический характер карты. Совместно с префиксными списками (**prefix-list**) карты абонентов обеспечивают удобный интерфейс и расширенную логику работы с IP-подсетями абонентов.

Для настройки карт абонентов используется команда конфигурационного режима `subscriber-map <NAME> <NUMBER>`. Где **NAME** может быть любым наименованием (до 15 символов). Рекомендуемый формат имени — все буквы прописные. **NUMBER** — номер последовательности правил (приоритет) обработки правил карты (задаётся числовыми значениями от 1 до 65535). В первую очередь будет обработана последовательность правил с номером 1, затем 2, 3 и т. д.. Последней будет обрабатываться последовательность правил с максимальным порядковым номером.

Для привязки карты абонента к интерфейсу необходимо ввести в контекстном режиме конфигурирования интерфейса BMI команду `subscriber-map <NAME>`, где **NAME** соответствует ранее созданной карте абонента.

Пример создания карты **"TEST"** с несколькими последовательностями правил внутри и ее привязки к BMI интерфейсу:

```
subscriber-map TEST 10
?
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
  exit
subscriber-map TEST 20
  exit
subscriber-map TEST 30
  exit
interface bmi.100
  subscriber-map TEST
```

В карте TEST первой будет обрабатываться последовательность правил с номером 10, затем 20, далее 30.

При создании карты, пользователь переходит в режим её конфигурирования. Доступны команды `match` и `set`, с помощью которых можно настроить соответствие



абонентской сессии (команда `match` ссылается на адрес абонента) с локальным или удалённым типом сервиса (команда `set` ссылается либо на имя локального сервиса, либо на AAA группу удалённых серверов).

Для сопоставления со всеми абонентами и сервисами используется **неявная** карта абонента с правилами (аналог **match ANY**), блокирующая весь трафик от абонентов. Синтаксис команд, определяющих правила (**match**) и действия (**set**) представлен ниже в разделах **Статические абоненты** и **Динамические абоненты** соответственно.

29.1.1 Статические абоненты

Статические абоненты — это абоненты, которые попадают под правила статических сессий. Статическое правило создаётся командой:

```
match static prefix-list <NAME> {untagged | svlan <значение> cvlan <значение> | cvlan <значение>}
```

Параметры команды описаны в таблице ниже.

Таблица 102 — Параметры команды `match static prefix-list`

Параметр	Описание
NAME	Имя префиксного списка (prefix-list). Префиксный список должен быть создан заранее
untagged	Нетегированный трафик
svlan	Сервисный VLAN
cvlan	Клиентский VLAN
значение	Значение VLAN ID

Ключевое слово **static** создаёт статическое правило, которое будет сопоставляться с одним конкретным IP-адресом, указанным через префиксный список (prefix-list). Значение параметра **NAME** должно соответствовать заранее сконфигурированному префиксному списку (prefix-list) с правилом **permit**. Дополнительную информацию по префиксным спискам можно найти в соответствующем разделе (см. "Списки доступа"). С помощью опций **svlan** и **cvlan** настраивается точное соответствие: IP-адрес — VLAN-теги абонента (802.1q и 802.1ad).

Если указаны слова **static** и имя префиксного списка, то требуется указать правила тегирования при отправке трафика в сторону абонента. Если в трафике отсутствуют теги, то следует указать ключевое слово **untagged**.

Например, в случае QinQ в LAN-сегменте абонента команда для создания статического правила для одного IP-адреса (одного устройства), трафик которого должен иметь внешний тег 10, а внутренний 20, будет выглядеть так `match static prefix-list`

`TEST svlan 10 cvlan 20`. Таким образом данные из маршрутизатора в сторону абонента будут выходить с двумя тегами в заголовке 802.1ad.

Абонент, удовлетворяющий статическому правилу `match`, по умолчанию считается локально аутентифицированным!

При создании статического правила, абонентская сессия появляется в глобальной таблице абонентов (вывод таблицы доступен по команде `show subscribers <NAME>`, где **NAME** — имя интерфейса BMI).

В версии ОС 3.2 при конфигурации префиксных списков в BRAS параметры **ge**, **le**, **eq** не учитываются.

Отсутствие статического правила в карте абонента

Если в одной из последовательностей правил карты абонента отсутствует правило **match**, то под эту последовательность попадают все IP-адреса абонентов. Это соответствует ситуации, если бы в последовательности правил карты абонента было правило **match dynamic prefix-list ALL**, где в **prefix-list ALL** было бы правило **permit 0.0.0.0/0 le 32**.

29.1.2 Динамические абоненты

Динамические абоненты — это абоненты, которые попадают под правила динамических сессий.

Динамическое правило создаётся командой:

`match dynamic prefix-list <NAME>`, где **NAME** соответствует заранее сконфигурированному префиксному списку (**prefix-list**) с правилом **permit**.

Ключевое слово **dynamic** создаёт динамическое правило, которое будет сопоставляться с одним или множеством IP-адресов с помощью префиксных списков (**prefix-list**). При указании динамического правила предполагается, что абонент или устройство получает настройки IP через DHCP или по приходу первого IP-пакета (параметр команды **session-trigger**). Во время прохождения DHCP-пакетов Discover, Offer, Request или Ack, маршрутизатор автоматически применяет правила тегирования для абонентов. Подобное поведение наблюдается при приёме первого IP-пакета от абонента. Поэтому команды для указания VLAN (**svlan**, **cvlan**, **untagged**) в динамических правилах не требуются.

Абоненты, IP-адреса которых удовлетворяют динамическому правилу **match**, считаются локально аутентифицированными. Однако аутентификация через удалённый AAA-сервер имеет наибольший приоритет, поэтому если в карте абонента присутствует правило **set** с ссылкой на удалённые AAA-сервера, то правило **match** не аутентифицирует абонентов локально, а указывает с каких устройств (для каких IP-адресов) должны идти AAA запросы на удалённые RADIUS-серверы.



Пользователь может получить доступ в Интернет только при успешной аутентификации. В случае отказа в аутентификации от AAA-сервера, время работы сессии составляет 5 мин. Это означает, что сессия будет автоматически удалена из глобальной таблицы абонентов через 5 мин (подробнее о абонентских таймерах читайте ниже).

Инициализация динамической 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 версии ОС 3.2 при конфигурации префиксных списков в BRAS параметры **ge**, **le**, **eq** не учитываются.

Отсутствие динамического правила в карте абонента

Если в одной из последовательностей правил карты абонента отсутствует правило **match**, то под эту последовательность попадают все IP-адреса абонентов. Это соответствует ситуации, если бы в последовательности правил карты абонента было правило **match dynamic prefix-list ALL**, где в **prefix-list ALL** было бы правило **permit 0.0.0.0/0 le 32**.

29.1.3 Пример настройки карты абонента с использованием статического префиксного списка

В следующем примере описывается процесс настройки статической карты абонента для абонента с адресом 192.168.0.1 из VLAN 100 с сервисной политикой в 10 Мб и блокирующей политикой для других абонентов.

Таблица 103 — Процесс настройки статической карты абонента

Консоль	Комментарий
<code>ecorouter(config)#ip prefix-list client_A permit 192.168.0.1/32</code>	Создание префиксного списка с именем client_A . Список содержит один разрешенный IP-адрес — 192.168.0.1/32
<code>ecorouter(config)#service-policy BANDWIDTH</code>	Создание сервисной политики с именем BANDWIDTH и переход в ее контекстный конфигурационный режим.
<code>ecorouter(config-policy)#bandwidth mbps 10</code>	Задание полосы пропускания для данной политики в 10 Мбит/с.
<code>ecorouter(config-policy)#exit</code>	Выход из контекстного конфигурационного режима

Консоль	Комментарий
<pre>ecorouter(config)#subscriber- service TEST</pre> <pre>ecorouter(config-service)#service- policy BANDWIDTH upstream</pre> <pre>ecorouter(config-service)#service- policy BANDWIDTH downstream</pre> <pre>ecorouter(config-service)#exit</pre>	<p>Создание сервиса абонента с именем TEST и переход в его контекстный конфигурационный режим.</p> <p>Применение сервисной политики с именем BANDWIDTH для ограничения трафика в направлениях upstream и downstream для данного сервиса абонента.</p> <p>Выход из контекстного конфигурационного режима</p>
<pre>ecorouter(config)#subscriber-map IPoE 5</pre> <pre>ecorouter(config-subscriber- map)#match static prefix-list client_A cvlan 100</pre> <pre>ecorouter(config-subscriber- map)#set subscriber-service TEST</pre> <pre>ecorouter(config-subscriber- map)#exit</pre>	<p>Создание карты абонента с именем IPoE и внутри нее последовательности с порядковым номером 5 и переход в ее контекстный конфигурационный режим.</p> <p>Создание статического правила, которое проверяет трафик на соответствие префиксному списку с именем client_A. При этом дополнительно проверяется наличие метки cvlan 100.</p> <p>Назначение сервиса с именем TEST для абонентов, попадающих под условие правила match.</p> <p>Выход из контекстного конфигурационного режима</p>
<pre>ecorouter(config)#interface bmi.100</pre> <pre>ecorouter(config-if)#subscriber-map IPoE</pre> <pre>ecorouter(config-if)#exit</pre>	<p>Создание интерфейса с именем bmi.100 и переход в его контекстный конфигурационный режим.</p> <p>Назначение на данный интерфейс карты абонента с именем IPoE.</p> <p>Выход из контекстного конфигурационного режима</p>

Консоль	Комментарий
<pre>ecorouter(config)#show run begin subscriber</pre>	Отображение текущей конфигурации начиная с фрагмента, начинающегося текстом "subscriber".
<pre>subscriber-service TEST service-policy BANDWIDTH upstream service-policy BANDWIDTH downstream</pre>	При просмотре отображаются успешно сделанные изменения в конфигурации.
<pre>subscriber-map IPoE 5 match static prefix-list client_A session-timeout 1440 idle-timeout 5 set service TEST</pre>	При получении первого пакета от клиента с адресом 192.168.0.1 из VLAN 100 сессия будет статически аутентифицирована, и ей будет доступна 10 Мбит/с полоса пропускания канала связи в обоих направлениях
[...]	

29.1.4 Пример настройки карты абонента с использованием динамического префиксного списка

В следующем примере описывается процесс настройки карты абонента с использованием статических и динамических префиксных списков, удаленного RADIUS-сервера, с примером конфигурирования L2-портов и L3-интерфейсов.

Таблица 104 — Процесс настройки карты абонента с использованием статических и динамических префиксных списков

Консоль	Комментарий
<pre>ecorouter(config)# ip prefix-list client_A permit 192.168.0.1/32</pre>	Создание префиксных списков: список с именем client_A содержит один разрешенный IP-адрес — 192.168.0.1/32;
<pre>ecorouter(config)# ip prefix-list clients_network_B permit 192.168.0.0/25</pre>	список с именем client_network_B содержит группу разрешенных IP-адресов — 192.168.0.0/25 ;
<pre>ecorouter(config)# ip prefix-list clients_network_C permit 192.168.0.128/25</pre>	список с именем client_network_C содержит группу разрешенных IP-адресов — 192.168.0.128/25

Консоль	Комментарий
<pre>ecorouter(config)# dhcp-profile 1</pre> <pre>ecorouter(config-dhcp)# mode relay</pre> <pre>ecorouter(config-dhcp)# server 192.168.1.2</pre> <pre>ecorouter(config-dhcp)# exit</pre>	<p>Создание DHCP-профиля с именем 1 и переход в его контекстный конфигурационный режим.</p> <p>Задание режима работы — relay.</p> <p>Задание адреса DHCP-сервера — 192.168.1.2.</p> <p>Выход из контекстного конфигурационного режима</p>
<pre>ecorouter(config)# radius-group NEW_RADIUS</pre> <pre>ecorouter(config-radius-group)# mode active-standby</pre> <pre>ecorouter(config-radius-group)# server 192.168.1.3 priority 10 secret pass1234</pre> <pre>ecorouter(config-radius-group)# exit</pre>	<p>Создание группы RADIUS-серверов с именем NEW_RADIUS и переход в ее контекстный конфигурационный режим.</p> <p>Задание режима работы — active-standby.</p> <p>Добавление в группу RADIUS-сервера с адресом 192.168.1.3, приоритетом 10 и секретным словом pass1234.</p> <p>Выход из контекстного конфигурационного режима</p>
<pre>ecorouter(config)# subscriber-aaa GROUP_C</pre> <pre>ecorouter(config-sub-aaa)# authentication radius NEW_RADIUS</pre> <pre>ecorouter(config-sub-aaa)# accounting radius NEW_RADIUS</pre> <pre>ecorouter(config-sub-aaa)# exit</pre>	<p>Создание абонентского AAA-профиля с именем GROUP_C и переход в его контекстный конфигурационный режим.</p> <p>Назначение аутентификации при помощи группы RADIUS-серверов с именем NEW_RADIUS.</p> <p>Назначение аккаунтинга при помощи группы RADIUS-серверов с именем NEW_RADIUS.</p> <p>Выход из контекстного конфигурационного режима</p>
<pre>ecorouter(config)# service-policy BANDWIDTH_A</pre>	<p>Создание сервисной политики с именем BANDWIDTH_A и переход в ее</p>

Консоль	Комментарий
<pre>ecorouter(config-policy)# bandwidth mbps 10 ecorouter(config-policy)# exit</pre>	<p>контекстный конфигурационный режим.</p> <p>Задание полосы пропускания для данной политики в 10 Мбит/с.</p> <p>Выход из контекстного конфигурационного режима</p>
<pre>ecorouter(config)# service-policy BANDWIDTH_B ecorouter(config-policy)# bandwidth kbps 512 ecorouter(config-policy)# exit</pre>	<p>Создание сервисной политики с именем BANDWIDTH_B и переход в ее контекстный конфигурационный режим.</p> <p>Задание полосы пропускания для данной политики в 512 кбит/с.</p> <p>Выход из контекстного конфигурационного режима</p>
<pre>ecorouter(config)# service-policy BANDWIDTH_C ecorouter(config-policy)# bandwidth mbps 5 ecorouter(config-policy)# exit</pre>	<p>Создание сервисной политики с именем BANDWIDTH_C и переход в ее контекстный конфигурационный режим.</p> <p>Задание полосы пропускания для данной политики в 5 Мбит/с.</p> <p>Выход из контекстного конфигурационного режима</p>
<pre>ecorouter(config)# subscriber-service TEST_A ecorouter(config-service)# service- policy BANDWIDTH_A upstream ecorouter(config-service)# service- policy BANDWIDTH_A downstream</pre>	<p>Создание сервиса абонента с именем TEST_A и переход в его контекстный конфигурационный режим.</p> <p>Применение сервисной политики с именем BANDWIDTH_A для ограничения трафика в направлениях upstream и downstream для данного сервиса абонента.</p>

Консоль	Комментарий
<code>ecorouter(config-service)# exit</code>	Выход из контекстного конфигурационного режима
<code>ecorouter(config)# subscriber-service TEST_B</code> <code>ecorouter(config-service)# service-policy BANDWIDTH_B upstream</code> <code>ecorouter(config-service)# service-policy BANDWIDTH_B downstream</code> <code>ecorouter(config-service)# exit</code>	<p>Создание сервиса абонента с именем TEST_B и переход в его контекстный конфигурационный режим.</p> <p>Применение сервисной политики с именем BANDWIDTH_B для ограничения трафика в направлениях upstream и downstream для данного сервиса абонента.</p> <p>Выход из контекстного конфигурационного режима</p>
<code>ecorouter(config)# subscriber-service TEST_C</code> <code>ecorouter(config-service)# service-policy BANDWIDTH_C upstream</code> <code>ecorouter(config-service)# service-policy BANDWIDTH_C downstream</code> <code>ecorouter(config-service)# exit</code>	<p>Создание сервиса абонента с именем TEST_C и переход в его контекстный конфигурационный режим.</p> <p>Применение сервисной политики с именем BANDWIDTH_C для ограничения трафика в направлениях upstream и downstream для данного сервиса абонента.</p> <p>Выход из контекстного конфигурационного режима</p>
<code>ecorouter(config)# subscriber-map IPoE 5</code> <code>ecorouter(config-subscriber)# match static prefix-list client_A cvlan 100</code>	<p>Создание карты абонента с именем IPoE и внутри нее последовательности с порядковым номером 5 и переход в ее контекстный конфигурационный режим.</p> <p>Создание статического правила, которое проверяет трафик на соответствие префиксному списку с именем client_A. При этом дополнительно проверяется</p>

Консоль	Комментарий
<pre>ecorouter(config-subscriber)# set subscriber-service TEST_A ecorouter(config-subscriber)# exit</pre>	<p>наличие метки cvlan 100.</p> <p>Назначение сервиса с именем TEST_A для абонентов, попадающих под условия правила match данной последовательности.</p> <p>Выход из контекстного конфигурационного режима</p>
<pre>ecorouter(config)# subscriber-map IPoE 10 ecorouter(config-subscriber)# match dynamic prefix-list clients_network_B ecorouter(config-subscriber)# set subscriber-service TEST_B ecorouter(config-subscriber)# exit</pre>	<p>Создание в карте абонента с именем IPoE последовательности с порядковым номером 10 и переход в ее контекстный конфигурационный режим.</p> <p>Создание динамического правила, которое проверяет трафик на соответствие префиксному списку с именем clients_network_B.</p> <p>Назначение сервиса с именем TEST_B для абонентов, попадающих под условие правила match данной последовательности.</p> <p>Выход из контекстного конфигурационного режима</p>
<pre>ecorouter(config)# subscriber-map IPoE 15 ecorouter(config-subscriber)# match dynamic prefix-list clients_network_C</pre>	<p>Создание в карте абонента с именем IPoE последовательности с порядковым номером 15 и переход в ее контекстный конфигурационный режим.</p> <p>Создание динамического правила, которое проверяет трафик на соответствие префиксному списку с именем clients_network_C.</p>



Консоль	Комментарий
<pre>ecorouter(config-subscriber)# set aaa GROUP_C</pre> <pre>ecorouter(config-subscriber)# set subscriber-service TEST_B</pre> <pre>ecorouter(config-subscriber)# exit</pre>	<p>Назначение абонентского AAA-профиля с именем GROUP_C для удалённой аутентификации.</p> <p>Назначение сервиса с именем TEST_C для абонентов, попадающих под условие правила match данной последовательности.</p> <p>Выход из контекстного конфигурационного режима</p>
<pre>ecorouter(config)# interface bmi.100</pre> <pre>ecorouter(config-if)# ip address 192.168.0.100/24</pre> <pre>ecorouter(config-if)# subscriber-map IPoE</pre> <pre>ecorouter(config-if)# dhcp-profile 1</pre> <pre>ecorouter(config-if)# exit</pre>	<p>Создание интерфейса с именем bmi.100 и переход в его контекстный конфигурационный режим. Данный интерфейс обрабатывает трафик между маршрутизатором и абонентом.</p> <p>Назначение на данный интерфейс группы IP-адресов 192.168.0.100/24.</p> <p>Назначение на данный интерфейс карты абонента с именем IPoE.</p> <p>Назначение на данный интерфейс DHCP-профиля с именем 1.</p> <p>Выход из контекстного конфигурационного режима</p>
<pre>ecorouter(config)# interface eth1</pre>	<p>Создание интерфейса с именем eth1 и переход в его контекстный конфигурационный режим. Данный интерфейс обрабатывает трафик между маршрутизатором и сетью Интернет.</p> <p>Назначение на данный интерфейс группы IP-</p>

Консоль	Комментарий
<pre>ecorouter(config-if)# ip address 77.77.0.1/30 ecorouter(config-if)# exit</pre>	<p>адресов 77.77.0.1/30.</p> <p>Выход из контекстного конфигурационного режима</p>
<pre>ecorouter(config)# port te0 ecorouter(config-port)# description LAN1 ecorouter(config-port)# service-instance A ecorouter(config-service- instance)# encapsulation dot1q 100 exact ecorouter(config-service- instance)# connect ip interface bmi.100 ecorouter(config-service- instance)# exit ecorouter(config-port)# service-instance B ecorouter(config-service- instance)# encapsulation dot1q 20 exact ecorouter(config-service- instance)# connect ip interface bmi.100 ecorouter(config-service- instance)# exit ecorouter(config-port)# exit</pre>	<p>Создание порта с именем te0 и переход в его контекстный конфигурационный режим.</p> <p>Задание описания для данного порта LAN1.</p> <p>Создание сервисного интерфейса с именем A и переход в его контекстный конфигурационный режим.</p> <p>Задание требования точного (exact) совпадения по значению инкапсуляции dot1q 100.</p> <p>Установка соединения между данным портом и интерфейсом с именем bmi.100.</p> <p>Выход из контекстного конфигурационного режима.</p> <p>Создание сервисного интерфейса с именем B и переход в его контекстный конфигурационный режим.</p> <p>Задание требования точного (exact) совпадения по значению инкапсуляции dot1q 20.</p> <p>Установка соединения между данным портом и интерфейсом с именем bmi.100.</p> <p>Выход из контекстного конфигурационного режима.</p>

Консоль	Комментарий
	Выход из контекстного конфигурационного режима
<pre>ecorouter(config)# port te1 ecorouter(config-port)# description LAN2 ecorouter(config-port)# service-instance C ecorouter(config-service-instance)# encapsulation dot1q 30-50 second-dot1q 10 ecorouter(config-service-instance)# connect ip interface bmi.100 ecorouter(config-service-instance)# exit ecorouter(config-port)# exit</pre>	<p>Создание порта с именем te1 и переход в его контекстный конфигурационный режим.</p> <p>Задание описания для данного порта LAN2.</p> <p>Создание сервисного интерфейса с именем C и переход в его контекстный конфигурационный режим.</p> <p>Задание требования совпадения по значению инкапсуляции dot1q 30-50 second-dot1q 100.</p> <p>Установка соединения между данным сервисным интерфейсом и интерфейсом с именем bmi.100.</p> <p>Выход из контекстного конфигурационного режима.</p> <p>Выход из контекстного конфигурационного режима</p>
<pre>ecorouter(config)# port te2 ecorouter(config-port)# description WAN ecorouter(config-port)# service-instance TEST ecorouter(config-service-instance)# encapsulation untagged</pre>	<p>Создание порта с именем te2 и переход в его контекстный конфигурационный режим.</p> <p>Задание описания для данного порта WAN.</p> <p>Создание сервисного интерфейса с именем TEST и переход в его контекстный конфигурационный режим.</p> <p>Задание требования совпадения по значению инкапсуляции untagged.</p> <p>Установка соединения между данным сервисным интерфейсом и интерфейсом с именем eth1.</p>

Консоль	Комментарий
<pre>ecorouter(config-service- instance)# connect ip interface eth1 ecorouter(config-service- instance)# exit</pre>	Выход из контекстного конфигурационного режима
<pre>ecorouter(config)# show run begin subscriber ! subscriber-aaa GROUP_C authentication radius NEW_RADIUS accounting radius NEW_RADIUS ! subscriber-service TEST_A service-policy BANDWIDTH_A upstream service-policy BANDWIDTH_A downstream ! subscriber-service TEST_B service-policy BANDWIDTH_B upstream service-policy BANDWIDTH_B downstream ! subscriber-service TEST_C service-policy BANDWIDTH_C upstream service-policy BANDWIDTH_C downstream ! subscriber-map IPoE 5 match static prefix-list client_A cvlan 100 session-timeout 1440 idle-timeout 5 set service TEST_A ! subscriber-map IPoE 10 match dynamic prefix-list clients_network_B session-timeout 1440 idle-timeout 5 set service TEST_B ! subscriber-map IPoE 10 match dynamic prefix-list clients_network_C session-timeout 1440 idle-timeout 5 set service TEST_B set aaa GROUP_C</pre>	<p>Отображение текущей конфигурации начиная с фрагмента, начинающегося текстом "subscriber".</p> <p>При просмотре отображаются успешно сделанные изменения в конфигурации.</p>



29.2 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 создаётся 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").

```
ecorouter(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 105 — Команды контекстного режима `ecorouter(config-pppoe)#`

Команда	Описание
<code>dns</code>	Задать IP-адрес DNS. Допускается создание одной (primary) или двух (primary и secondary) записей. Подробнее см. пример ниже

Команда	Описание
<code>gateway</code>	Задать IP-адрес шлюза
<code>pado-timeout <0-65535></code>	Задать величину задержки между получением PADI и ответом PADO в миллисекундах. Диапазон значений 0—65535
<code>pool</code>	Задать пул IP-адресов (см. раздел "Пул IP-адресов для PPPoE клиентов")
<code>ppp</code>	Команды для настройки Point-to-Point Protocol (см. раздел "Point-to-Point Protocol")
<code>set</code>	Команды для задания политик (см. раздел "Команды set для конфигурирования PPPoE")
<code>tag-ac-name <ACNAME></code>	Задать значение тега PPPoE AC-name, которое будет отображаться в ответном PADO пакете
<code>tag-service-name <SRVNAME></code>	Задать значение тега PPPoE service-name, которое будет отображаться в ответном PADO пакете. При задании команды <code>tag-service-name any</code> , сервер будет принимать от абонентов любое значение поля service-name, включая пустое

Пример создания, конфигурации и просмотра PPPoE-профиля:

```
pppoe-profile 1
dns ipv4 192.168.10.100
dns ipv4 192.168.10.200 secondary
pado-timeout 50
tag-ac-name ER-1
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 счётчиков 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` приведён 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
authentication status: accepted(L)
type: PPPoE
encapsulation: untagged
wan pkts: 1
lan pkts: 1
wan bytes: 98
lan bytes: 106
```

29.2.1 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:

- Задать адрес на bmi-интерфейсе из PPPoE-подсети и включить интерфейс bmi в протокол динамической маршрутизации так же, как и обычный IP-интерфейс.
- 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.

29.2.2 Параметры PPPoE при аутентификации через RADIUS-сервер

29.2.2.1 Протокол PAP (Password Authentication Protocol)

При аутентификации PPPoE-абонента через RADIUS-сервер с использованием протокола PAP маршрутизатор 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, ставший триггером для отправки запроса на RADIUS-сервер (пакет-триггер). 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 — 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. В текущей реализации — **PPP**;
- **NAS-Port** — c-vlan — внутренняя метка VLAN из заголовка пакета-триггера.



29.2.2.2 Протокол CHAP (Challenge Handshake Authentication Protocol)

При аутентификации PPPoE абонента через RADIUS-сервер с использованием протокола CHAP маршрутизатор 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.

29.2.3 Параметры IPoE при аутентификации через RADIUS-сервер

При аутентификации абонента через RADIUS-сервер маршрутизатор 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>:<с-vlan>:<с-vlan>` — порт и интерфейс указываются те, на которые пришёл пакет, ставший триггером для отправки запроса на RADIUS-сервер (пакет-триггер). 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;
- **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;
- **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;

- **NAS-Port** — c-vlan — внутренняя метка VLAN из заголовка пакета-триггера.

При аутентификации абонента через RADIUS-сервер маршрутизатор 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 применён на сессию. Сервис будет применён на сессию при условии, что он создан на маршрутизаторе при помощи команды `subscriber-service <service_name>`.

29.2.4 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. В текущей реализации — **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 счётчика 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).

29.2.5 Аутентификация PPPoE

Настройка PPPoE-аутентификация абонентов в ОС версии 3.2.

Для выбора протоколов аутентификации необходимо выполнить следующие шаги:

- Перейти в контекстный режим конфигурирования PPPoE-профиля.
- Включить аутентификацию через PPPoE.
- Указать группу RADIUS-серверов, которые будут использованы для удалённой 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.

```
ecorouter(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 (`pppoe-profile`)). Подробнее о группах RADIUS-серверов читайте в соответствующем разделе ("Авторизация в системе").

ВНИМАНИЕ: аутентификация производится только при помощи RADIUS-серверов, локальная аутентификация не поддерживается.

29.2.6 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:

```
ecorouter(config-pppoe)#ppp
authentication      Authentication
auth-req-limit      Auth request limit
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 106 — Параметры настройки параметров PPP



Параметр с диапазоном значений	Описание
authentication	Настройка аутентификации (подробнее см. в разделе "Аутентификация PPPoE")
auth-req-limit <1-100>	Максимальное количество запросов Auth Request от абонента при выполнении процедуры аутентификации на удалённом сервере (по-умолчанию 10)
max-configure <1-20>	Максимальное количество запросов Configure-Request перед получением ответа (значение по умолчанию 10)
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)

29.2.7 Пул IP-адресов

Создание пула IP-адресов для выдачи их PPPoE-абонентам.

Создание пула IP-адресов производится с помощью команды конфигурационного режима **ip pool <NAME>**, где **NAME** — имя пула. Команда переводит CLI в контекстный режим настройки пула, где командой **range <A.B.C.D-E.F.G.H>** через дефис задаются диапазоны адресов. Единичный адрес задаётся командой **range <A.B.C.D>**, но в конфигурации он будет записан в виде диапазона. Всего можно задать до 15 диапазонов в одном пуле.

По умолчанию каждый диапазон включён в работу командой **activate**. При необходимости диапазон можно исключить из работы командой **deactivate** в контексте диапазона.

В контексте настройки пула можно задать описание пула командой **description <LINE>**, где **LINE** — текстовая строка (допустимы пробелы).

Пример:



```
ip pool POOL_NAME
  range 1.1.1.1-1.1.1.10
  range 2.2.2.2-3.3.3.3
```

Для удаления пула IP-адресов используется команда конфигурационного режима `no ip pool <NAME>`.

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
```

Index	Pool	Begin	End	Free	In use	State
1	0	192.168.10.2	192.168.10.254	1	252	activate
	0	192.168.12.2	192.168.12.2	10	243	deactivate

Для просмотра информации по выбранному пулу используется команда `show ip pool <NAME>`.

```
ecorouter#show ip pool POOL_NAME
```

Index	Pool	Begin	End	Free	In use	State
2	POOL_NAME	1.1.1.1	1.1.1.10	1	0	activate
		2.2.2.2	3.3.3.3	16843010	0	activate

Для назначения пула выделяемых по умолчанию IP-адресов используется команда `pool ipv4 <NAME>` контекстного режима конфигурации `pppoe-profile` и `config-ppp`, где **NAME** — наименование пула.

Для отмены назначения пула выделяемых по умолчанию IP-адресов используется команда `no pool ipv4 <NAME>`.

29.2.8 Команды set для конфигурирования PPPoE

Для настройки некоторых параметров PPPoE используется команда `set` в контекстном режиме конфигурирования `(config-pppoe)#`. Параметры, доступные для настройки, перечислены в таблице.

Таблица 107 — Параметры команды в `set` контекстном режиме `(config-pppoe)#`

Параметр	Описание
----------	----------

Параметр	Описание
aaa SUBSCRIBER_AAA	Назначить заранее созданный AAA-профиль абонента
idle-timeout <0-1440>	Задать idle-timeout в минутах. Значение по умолчанию — 30 минут. Значение 0 минут означает бесконечно большое значение параметра
subscriber-service SERVICE_NAME	Назначить заранее созданный сервис абонента
session-timeout <0-527040>	Задать session-timeout в минутах. Значение по умолчанию — 1440 минут. Значение 0 минут означает бесконечно большое значение параметра
update-interval <5-1440>	Задать интервал аккаунтинга в минутах

Пример:

```
ecorouter(config)#subscriber-aaa SUB_AAA
ecorouter(config-sub-aaa)#ex
ecorouter(config)#pppoe-profile 111
ecorouter(config-pppoe)#set subscriber-service SUB_SERV
ecorouter(config)#pppoe-profile PPPOE_PROFILE
ecorouter(config-pppoe)#set aaa
    SUBSCRIBER_AAA Subscriber AAA profile name
ecorouter(config-pppoe)#set aaa SUB_AAA
ecorouter(config-pppoe)#ex
ecorouter(config)#ex
ecorouter#show pppoe-profile PPPOE_PROFILE
pppoe-profile PPPOE_PROFILE
AAA profile: SUB_AAA
Service: SUB_SERV
PPP options
Authentication: no
Configure-Request limit: 10
Configure-Nak limit: 5
Terminate-Request limit: 1
Echo-Request limit: 5
Auth request limit: 10
Retry timeout: 3
Echo timeout: 10
Gateway address:
Primary DNS address:
```

29.3 Аутентификация, авторизация и аккаунтинг

29.3.1 Локальная аутентификация

IPoE-абонент считается локально аутентифицированным, если IP-адрес абонента соответствует статическому или динамическому правилу в последовательности **subscriber-map**, в которой отсутствует команда **set aaa** с указанием имени группы удалённых AAA RADIUS-серверов.

Для PPPoE абонентов возможность локальной аутентификации отсутствует, однако можно полностью отключить аутентификацию абонентов в PPPoE профайле с помощью команды **no authentication**. В этом случае любая попытка абонентского PPP подключения будет считаться успешной.

29.3.2 Локальная авторизация

Под авторизацией подразумевается конфигурация для абонентов определённых сервисов (с какой скоростью осуществляется передача данных для абонента в разных направлениях). Существует возможность использования локально сконфигурированного сервиса, а также полученного через удалённый RADIUS-сервер. Приведённые 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 PPPoE-профилю или к картам абонентов IPoE вручную или получен с RADIUS-сервера:

```
ecorouter(config)#subscriber-service ?  
SUBSCRIBER_SERVICE  Subscriber service name
```

Для **subscriber-service** следует назначить **subscriber-policy**.

```
ecorouter(config-sub-service)#set ?  
policy  Set policy  
ecorouter(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>`  
ecorouter(config-sub-policy)#bandwidth ?  
in  Upstream packets
```



```
out Downstream packets
ecorouter(config-sub-policy)#bandwidth in
  kbps Bandwidth value in kbps
ecorouter(config-sub-policy)#bandwidth in kbps ?
  <64-10000000> Kbits per second
ecorouter(config-sub-policy)#set filter-map ?
  in Upstream packets
  out Downstream packets
ecorouter(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.

```
ecorouter(config)#filter-map policy ipv4 ?
  FILTER_MAP_POLICY_IPV4 Filter map name
ecorouter(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 PPPoE-профиле и карте абонентов IPoE:

```
ecorouter(config-pppoe)#set subscriber-service ?
  SUBSCRIBER_SERVICE Specify subscriber service name
```

Ниже приведён пример полной настройки для PPPoE.

- Настройка **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
```

- Настройка **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
```

- Настройка **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.

- После установки соединения состояние сервиса можно посмотреть командой **show subscribers <interface bmi> <ip addr>** .

5.1 В случае задания **subscriber-service** вручную после названия 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)
...
```

Локальная авторизация для IPoE-абонентов конфигурируется аналогичным образом, установкой нужного **subscriber-service** в последовательности **subscriber-map**. По умолчанию авторизация через RADIUS имеет наибольший приоритет, ключевое слово **strict** в команде **set subscriber-service <NAME>** позволяет сделать локальную авторизацию приоритетной.

29.3.2.1 Отсутствие сервиса в карте абонента

Если в одной из последовательностей карты абонента отсутствует правило `set`, то в этой последовательности все абоненты, попавшие под правило **match** (отсутствие правила **match** соответствует всем IP-адресам), попадают под неявное правило DROP. Весь трафик от этих абонентов блокируется, а сервис считается недействительным. Время жизни для таких сессий устанавливается 5 мин, то есть, сессия будет удалена автоматически из глобальной таблицы абонентов через 5 мин.

29.3.3 Удалённая аутентификация, авторизация и аккаунтинг при помощи RADIUS

Для аутентификации, авторизации 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 произведён 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.

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)#
```

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.

В данный момент для установки сервиса от AAA-сервера требуется выполнение следующих условий:

- Наличие сконфигурированного абонентского сервиса (subscriber-service) на маршрутизаторе.
- Конфигурация группы AAA-серверов для абонентов с помощью `subscriber-aaa`.
- Полное соответствие имени абонентского сервиса и имени сервиса в сообщении от AAA-сервера.

При соблюдении вышеуказанных требований, установить сервис от RADIUS-сервера можно с помощью команды `set aaa <NAME>`, где **NAME** соответствует заранее сконфигурированной группе AAA-серверов для абонентов. Напомним, что при наличии этой команды в карте абонента аутентификация и авторизация меняются с локальной на удалённую для этой последовательности в `subscriber-map`.

Если от AAA-сервера приходит сервис, имя которого не найдено в конфигурации маршрутизатора, и локальных сервисов для этих абонентов не предусмотрено в **subscriber-map**, то сервис для клиентов считается недействительным и трафик от абонентов блокируется.



Для использования настроенного профиля в PPPoE необходимо перейти в контекстный конфигурационный режим PPPoE профиля (config-pppoe) и выполнить аналогичную команду `set aaa <SUBSCRIBER_AAA>`.

29.3.4 Группы RADIUS-серверов

Для удалённой авторизации/аутентификации и аккаунтинга на EcoRouter поддерживается использование групп RADIUS-серверов. Данная 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 её создания в результате выполнения команды будет автоматически произведён 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 (config-radius-group)# можно отредактировать или удалить описание группы, настроить режим её работы, изменить параметры выбранного RADIUS-сервера или удалить выбранный RADIUS-сервер из группы. Данные команды и параметры описаны в таблице ниже.

Таблица 108 — Команды контекстного режима (config-radius-group)#

Команда/ параметр	Описание
<code>description <TEXT></code>	Задание описания группы RADIUS-серверов.
<code>no description</code>	Удаление описания группы RADIUS-серверов
<code>mode <MODE></code>	Настройка режима работы группы RADIUS-серверов. Допустимые значения режима работы группы RADIUS-серверов — MODE : - 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

Команда/ параметр	Описание
	<p>RADIUS-сервер с наибольшим приоритетом перестает отвечать на запросы, то запросы начинают поступать на следующий по приоритету сервер. По истечении определённого периода времени производится попытка повторить отправку запросов на наиболее приоритетный сервер. Если такая попытка удачна, то он снова становится активным.</p> <p>- 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.</p> <p>Значение по умолчанию — active-standby</p>
transmission-rate threads <NUMBER> packets <NUMBER>	<p>Количество одновременно отправляемых запросов на RADIUS-сервер. Задаётся двумя параметрами:</p> <p>- threads — максимальное количество одновременных потоков. Диапазон значений: от 1 до 12. Значение по умолчанию: 4.</p> <p>- packets — максимальное количество пакетов на поток. Диапазон значений: от 64 до 256. Значение по умолчанию: 256.</p> <p>Общее количество запросов — произведение threads × packets</p>
Настройка таймеров	
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-</p>

Команда/ параметр	Описание
	<p>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. 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 формата атрибута Calling-Station-Id	
<code>attribute mac default</code>	Использовать формат по умолчанию. Имеет вид — XXXX.XXXX.XXXX
<code>attribute mac ietf</code>	Использовать формат IETF. Имеет вид — XX-XX-XX-XX-XX-XX
<code>attribute mac unformatted</code>	Использовать формат без разделителей. Имеет вид — XXXXXXXXXXXXXX
Настройка формата атрибута Nas-Port	
<code>attribute nas-port default</code>	Использовать комбинацию VLAN: сервисного и клиентского
<code>attribute nas-port session-id</code>	Использовать идентификатор сессии
Настройка формата атрибута username	



Команда/ параметр	Описание
<code>attribute username format <></code>	<p>Формат атрибута username. Возможные значения:</p> <ul style="list-style-type: none"> - default — по умолчанию используется username = mac address, - любая комбинация из полей: cvlan, interface, ip, mac, svlan. <p>Разделитель для этих полей — символ '.'.</p> <p>Атрибут модифицируется только для IPoE абонентов</p>
Настройка подсчета трафика по сессии	
<code>attribute accounting direction port</code>	Направление трафика относительно порта маршрутизатора
<code>attribute accounting direction subscriber</code>	Направление трафика относительно пользователя

Настройка параметров отдельного сервера в группе

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

```
radius-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>]
```

IP-адрес и секретный ключ — обязательные параметры. Остальные параметры не являются обязательными и могут быть заданы в любом порядке. Если при вызове команды указан IP-адрес существующего RADIUS-сервера, то будут изменены его параметры. Иначе будет создан RADIUS-сервер с указанным IP-адресом.

Таблица 109 — Параметры команды `server`

Параметр	Описание
<code>server A.B.C.D</code>	IP-адрес RADIUS-сервера
<code>secret <WORD></code>	Значение атрибута secret (по умолчанию не задан)
<code>priority <0-65535></code>	Приоритет RADIUS-сервера (актуально для режима active/standby). Чем меньше значение, тем выше приоритет
<code>vrf <VRF></code>	Имя VRF, в котором задан IP-адрес RADIUS-сервера (значение по умолчанию — VRF текущего виртуального маршрутизатора)

Параметр	Описание
<code>source A.B.C.D</code>	IP-адрес, который будет указан в качестве адреса источника в пакете запроса (по умолчанию — адрес интерфейса, с которого уходит запрос)
<code>auth-port <1-65535></code>	Порт для запросов аутентификации (значение по умолчанию 1812)
<code>acct-port <1-65535></code>	Порт для запросов аккаунтинга (значение по умолчанию 1813)
<code>coa-listen-port <1-65535></code>	Порт, на основе которого на BRAS будет открыт сокет для обработки coa 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
ecorouter(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 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
!
```

29.4 Фильтрация и HTTP перенаправление

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 её параметров.

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

Таблица 110 — Параметры команды `subscriber-policy`

Параметр	Описание
<code><BANDWIDTH></code>	Ширина полосы пропускания в Мбит/сек от 1 до 200
<code><DESCRIPTION></code>	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
```

29.4.1 Создание и настройка filter-map policy

Создание filter-map policy производится при помощи команды конфигурационного режима `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 её 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):

4. 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.
5. 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.
6. 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.

Таблица 111 — Общие параметры команды `filter-map policy`



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 от 0 до 65535. Если значение не задано, то параметр для созданного filter-map ethernet автоматически получит последующее свободное значение с шагом 10
PROTOCOL	<p>Значение поля protocol. Может быть указано значение поля в диапазоне от 0 до 255 или одно из следующих обозначений:</p> <ul style="list-style-type: none"> - ipinip; - icmp; - gre; - igmp; - pim; - 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	<p>IP-адрес источника, задается в одном из следующих форматов:</p> <ul style="list-style-type: none"> - 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	<p>IP-адрес назначения, задаётся в одном из следующих форматов:</p> <ul style="list-style-type: none"> - 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>	

Параметр	Описание
<code>set accept</code>	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
<code>set discard</code>	Запретить без отправки ICMP-уведомления
<code>set nexthop <A.B.C.D></code>	Указать IP-адрес next hop. Пакеты, попавшие под действие правила, отсылаются на адрес next-hop с учётом существующих маршрутов в RIB
<code>set redirect <REDIRECTNAME></code>	Перенаправить HTTP GET на указанный REDIRECTNAME , где REDIRECTNAME — имя заранее заданного URL (адрес для перенаправления должен начинаться с http://). Пример настройки перенаправления приведён ниже.
<code>set reject</code>	Запретить с отправкой ICMP-уведомления
<code>set vrf <VRF_NAME> [<A.B.C.D>]</code>	Для пакетов, попавших под действие правила, будет использоваться таблица маршрутизации vrf, где VRF_NAME — имя необходимого vrf. Для данного vrf можно при необходимости указать IP-адрес next hop

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>] **.`

Таблица 112 — Дополнительные параметры команды `filter-map policy` при указании **udp**

Параметр	Описание
<code>PORT_CONDITION</code>	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	
<code>eq</code>	Номер порта равен
<code>gt</code>	Номер порта больше, чем
<code>lt</code>	Номер порта меньше, чем
<code>tftp</code>	UDP(69)
<code>bootp</code>	UDP(67)

Параметр	Описание
<code><0-65535></code>	Точный номер порта, любое значение из указанного диапазона
<code>range <0-65535> <0-65535></code>	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>]`.

Таблица 113 — Дополнительные параметры команды `filter-map policy` при указании **tcp**

Параметр	Описание
PORT_CONDITION	Условие для значения порта. Может быть указано одно из следующих значений: <code>{{eq gt lt} {ftp ssh telnet www <0-65535>} range <0-65535> <0-65535>}</code>
FLAG	<p>Значения флага, по которым может производиться обработка пакетов. Может быть указано одно из следующих значений (префикс <code>not</code> — означает, что указанный флаг не установлен): ack not-ack fin not-fin psh not-psh rst not-rst syn not-syn urg not-urg</p> <ul style="list-style-type: none"> - 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. <p>Можно перечислить несколько флагов через пробел. При этом правило сработает, если в пакете будут установлены все перечисленные флаги. Например, правило <code>not-rst syn ack</code> сработает, если пакет содержит флаги SYN и ACK, но не содержит RST</p>

Параметр	Описание
Значения PORT_CONDITION	
<code>eq</code>	Номер порта равен
<code>gt</code>	Номер порта больше, чем
<code>lt</code>	Номер порта меньше, чем
<code>ftp</code>	TCP(21)
<code>ssh</code>	TCP(22)
<code>telnet</code>	TCP(23)
<code>www</code>	TCP(HTTP-80)
<code><0-65535></code>	Точный номер порта, любое значение из указанного диапазона
<code>range <0-65535> <0-65535></code>	Номер порта входит в диапазон

29.4.1.1 Задание адреса для перенаправления

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

29.4.1.2 Пример настроек для обработки трафика в абонентской сессии

В данном примере настроен статический IPoE.

В результате выполнения приведённых 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
```

29.5 Удалённая аутентификация, авторизация и аккаунтинг

29.5.1 Удалённая аутентификация, авторизация и аккаунтинг при помощи RADIUS

Для аутентификации, авторизации 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 произведён 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.

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)#
```

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.

В данный момент для установки сервиса от AAA-сервера требуется выполнение следующих условий:

7) Наличие сконфигурированного абонентского сервиса (**subscriber-service**) на



маршрутизаторе.

8) Конфигурация группы AAA-серверов для абонентов с помощью **subscriber-aaa**.

9) Полное соответствие имени абонентского сервиса и имени сервиса в сообщении от AAA-сервера.

При соблюдении вышеуказанных требований, установить сервис от RADIUS-сервера можно с помощью команды `set aaa <NAME>`, где **NAME** соответствует заранее сконфигурированной группе AAA-серверов для абонентов. Напомним, что при наличии этой команды в карте абонента аутентификация и авторизация меняются с локальной на удалённую для этой последовательности в **subscriber-map**.

Если от AAA-сервера приходит сервис, имя которого не найдено в конфигурации маршрутизатора, и локальных сервисов для этих абонентов не предусмотрено в **subscriber-map**, то сервис для клиентов считается недействительным и трафик от абонентов блокируется.

Для использования настроенного профиля в PPPoE необходимо перейти в контекстный конфигурационный режим PPPoE профиля (`pppoe-profile`)# и выполнить аналогичную команду `set aaa <SUBSCRIBER_AAA>`.

29.5.2 Параметры PPPoE при аутентификации через RADIUS-сервер

29.5.2.1 Протокол PAP (Password Authentication Protocol)

При аутентификации PPPoE-абонента через RADIUS-сервер с использованием протокола PAP маршрутизатор 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, ставший триггером для отправки запроса на RADIUS-сервер (пакет-триггер). 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 — 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. В текущей реализации — **PPP**;
- **NAS-Port** — c-vlan — внутренняя метка VLAN из заголовка пакета-триггера.

29.5.2.2 Протокол CHAP (Challenge Handshake Authentication Protocol)

При аутентификации PPPoE абонента через RADIUS-сервер с использованием протокола CHAP маршрутизатор 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.

29.5.3 Параметры IPoE при аутентификации через RADIUS-сервер

При аутентификации абонента через RADIUS-сервер маршрутизатор 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, ставший триггером для отправки запроса на RADIUS-сервер (пакет-триггер). 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;
- **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;
- **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;
- **NAS-Port** — c-vlan — внутренняя метка VLAN из заголовка пакета-триггера.

При аутентификации абонента через RADIUS-сервер маршрутизатор 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 применён на сессию. Сервис будет применён на сессию при условии, что он создан на маршрутизаторе при помощи команды **subscriber-service <service_name>**.

29.5.4 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. В текущей реализации — **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 счётчика 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).

29.5.5 Функция Authentication Failover

Если по какой-либо причине удалённый AAA-сервер недоступен, BRAS может автоматически применять локальные политики аутентификации и авторизации. Для этого предусмотрена функция Authentication Failover. Благодаря этой функции абоненты смогут получить доступ в Интернет и даже не заметят сбоя в сети оператора. По умолчанию данная функция выключена. Для её использования должны быть выполнены два условия:

10. Функция должна быть включена командой `authentication-failover` в режиме конфигурации интерфейса **bmi**.

11. В **subscriber-map** или в **pppoe-profile**, в зависимости от типа абонентов сети, должен быть сконфигурирован локальный сервис.

Ниже представлен пример настройки функции **authentication-failover** для локального сервиса с именем **2mbps** и недоступного RADIUS-сервера из группы с именем **NEW_RADIUS**.

```
...
interface bmi.1
  connect port te0 service-instance clients
  dhcp-profile 1
  subscriber-map clients
  session-trigger dhcp
  authentication-failover
  ip address 10.1.1.1/24
subscriber-map clients 1
  set idle-timeout 30
  set session-timeout 1440
  match dynamic prefix-list PERMITANY
  set subscriber-service 2mbps
  set aaa radius
subscriber-aaa radius
  authentication radius NEW_RADIUS
  accounting radius NEW_RADIUS
radius-group NEW_RADIUS
  radius-server 192.168.255.2 secret pass1234 vrf management priority 10
subscriber-policy 2mbps
  bandwidth in mbps 2
  bandwidth out mbps 2
  set filter-map in default
  set filter-map out default
subscriber-service 2mbps
```




```
set policy 2mbps
...
```

После аутентификации и авторизации абонента с помощью функции **authentication-failover** его сессия в таблице IPoE обозначается меткой "F", а также записью "auth. failed" в столбце Status, которая говорит о невозможности связаться с удалённым AAA-сервером.

```
ecorouter#show subscribers bmi.1
VRF: default
Total subscribers: 2
Accepted: 2, Rejected: 0, Authenticating: 0, DHCP conversation: 0
Codes:
    l - local authentication (prefix-list), r - remote authentication
(subscriber-aaa)
    L - local authorization (subscriber-service), R - remote authorization
(radius attribute SERVICE_NAME)
    B - blocked by IP Source Guard, F - local auth during Radius unavailable
(authentication-failover)
    U - unknown (internal error), N - not specified
IP Address      MAC Address      Port      S-tag  C-tag  Status      Type
-----
F> 10.1.1.3      0050.7966.6801    te0       ----- 10    auth. failed  IPoE L2
```

При вводе команды **authentication-failover** можно задать тайм-аут для автоматического сброса абонентских сессий с меткой F:

authentication-failover <0-65535>, где число — это время в минутах, по истечении которого произойдёт сброс всех абонентских сессий с меткой F (0 означает бесконечность).

Задание тайм-аута для **authentication-failover** позволит абонентским устройствам автоматически пересоздавать сессии в BRAS, и сетевому администратору не придётся вручную закрывать все необходимые сессии командой **clear**.

Применение тайм-аута **authentication-failover** происходит следующим образом. При аутентификации абонента параметру **session-timeout** в настройках соответствующей **subscriber-map** присваивается значение, указанное в команде **authentication-failover**. При восстановлении связи с удалённым AAA-сервером тайм-аут **authentication-failover** продолжает действовать. BRAS сможет инициировать новую сессию для абонентского устройства через запрос к удалённому AAA-серверу только по истечении тайм-аута **authentication-failover** или после принудительного закрытия текущей сессии командой **clear**.

Следует также помнить, что при использовании функции **authentication-failover** значение параметра **idle-timeout** для абонентских сессий не изменяется и остаётся

равным значению из соответствующей **subscriber-map**. Поэтому сброс абонентской сессии может произойти до истечения тайм-аута **session-timeout**.

29.6 Таймеры абонентских сессий

Для абонентских сессий действуют следующие таймеры:

- **session-timeout <5-45000>** время жизни активной сессии (1440 мин. по умолчанию);
- **idle-timeout <1-10>** время жизни неактивной сессии (5 мин. по умолчанию).

Таймеры создаются автоматически при создании новой последовательности карты или PPPoE профиля. При желании пользователь может изменить поведение по умолчанию, настроив специфичные опции для конкретных сессий с помощью команд **set session-timeout** и **set idle-timeout** соответственно.

29.7 Команды группы show для BRAS

29.7.1 Команда просмотра состояния PPPoE сессии

Состояние PPPoE сессии можно посмотреть с помощью команды **show subscribers bmi.1 pppoe detail**:

```
ecorouter#show subscribers bmi.1 pppoe ?
detail  PPPoE subscribers detail info
|       Output modifiers
<cr>
```

В результате выполнения команды отображается таблица с основными характеристиками состояния сессии, в том числе и для ещё не установившейся (пояснения вывода см. в таблицах ниже):

```
ecorouter#show subscribers bmi.1 pppoe detail
VRF: default
```

MAC Address	S-tag	C-tag	MRU	Port	ID	Service	PPP-State
PPP-Auth	User		IP Address				

cafe.da70.7300	70	73	1492	ge2	0xe307	PPPoE	network



```

chap      pppoe-3      192.168.250.4
cafe.da70.7100 70      71      1492    ge2      0x7405  PPPoE      network
chap      pppoe-1      192.168.250.5
cafe.da70.7700 70      77      1492    ge2      0xbf02  PPPoE      network
chap      pppoe-7      192.168.250.6

```

Таблица 114 — Параметры вывода команды `show interface bmi.0 pppoe clients`

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

Таблица 115 — Значения параметра **PPP-State**

Значение	Пояснение
down	Physical-layer not ready
establish	Link Establishment Phase
authenticate	Authentication Phase
network	Network-Layer Protocol Phase
terminate	Link Termination Phase

Таблица 116 — Значения параметра **PPP-Auth**

Значение	Пояснение
none	Без аутентификации
pap	Аутентификации по протоколу PAP
chap	Аутентификации по протоколу CHAP
ms-chap-v1	Аутентификации по протоколу MS-CHAPv1
ms-chap-v2	Аутентификации по протоколу MS-CHAPv2

29.7.2 Команды просмотра карт абонентов и сервисов абонентов

Подробную информацию по определённой 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 приведён вывод при отсутствии применённой карты абонента на интерфейсе (subscriber-map не была применена на BMI-интерфейсе).

```
Subscriber-map "clients" is applied for:
Interface      IP-Address
<empty>        <empty>
```

Если при вызове команды `show subscriber-map` имя карты отсутствует, то отображается краткая информация по всем картам абонентов.

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 счётчиков 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 счётчиков 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	te1	----	----	accepted(L)	IPoE
20.20.20.3	7e6e.5221.bf2a	te1	----	----	accepted(L)	IPoE
20.20.20.5	0000.0000.0000	te1	----	----	accepted(L)	static
20.20.20.6	8e5e.5223.e212	te1	----	----	accepted(L)	PPPoE

Таблица 117 — Параметры вывода команды `show subscribers <INAME>`

Параметр	Описание
IP Address	IP-адрес абонента

Параметр	Описание
MAC Address	MAC-адрес абонента
Port	Порт, через который подключён абонент
S-tag, C-tag	VLAN-теги абонентского трафика
Status	Статус данного абонента
Type	Тип подключения: - static — абонент задан через CLI в subscriber-map; - IPoE — IPoE сессия; - PPPoE — PPPoE сессия; - dhcp — абонент находится на стадии получения IP-адреса с DHCP-сервера
Статусы	
accepted	Абонент успешно аутентифицировался на RADIUS-сервере
rejected	Абонент заблокирован
in progress	Отправлен запрос на RADIUS-сервер
Статусы при типе подключения DHCP	
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 ОС версии 3.2 есть возможность вручную сбросить абонентскую сессию или счётчики пакетов и байтов по сессии. Для сброса сессии в административном режиме необходимо выполнить команду: `clear subscribers IFNAME ip|mac|all`.

Для сброса счётчиков по сессии в административном режиме необходимо выполнить команду: `clear counters subscribers IFNAME ip|mac|all`.

Абонентскую сессию или счётчики по ней можно сбросить по IP-адресу или по MAC-адресу — в случае, когда у абонента ещё нет IP-адреса. Также можно сбросить все сессии (или счётчики по всем сессиям) на определённом BMI-интерфейсе. Сброс счётчиков по сессии инициирует отправку **Interim-Update** accounting сообщения с обновлёнными атрибутами **Acct-Input-Octets**, **Acct-Output-Octets**, **Acct-Input-Packets**, **Acct-Output-Packets**, **Acct-Input-Gigawords** и **Acct-Output-Gigawords**.

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   te1   ----  ----  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   te1   ----  ----  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 te1  ----  ----  accepted(L) IPoE
20.20.20.3 7e6e.5221.bf2a te1  ----  ----  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: te1
service: ddff
session timeout: 3 min
session time remaining: 0 min
idle timeout: 3 min
idle time remaining: 0 min
authentication 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 информация по subscriber-policy, service-policy, а также выведен список карт абонентов, на которых применен указанный сервис.

Для просмотра счётчиков по CoA и Disconnect запросам служит команда `show counters subscribers coa-messages`.

Пример:

```
ecorouter#show counters subscribers coa-messages
```

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	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.

29.8 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 выключён. 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:

```
ecorouter# 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
```

29.9 Рекомендации и тонкости настройки

29.9.1 IPoE

Последовательности правил в карте абонента проверяются в порядке возрастания их номера. В конфигурации присутствует неявная карта с максимальным номером (больше, чем у любой карты абонента, созданной пользователем), которая сопоставляется со всеми устройствами на интерфейсе BMI (все IP-адреса абонентов) и сервисом, который блокирует весь трафик от клиентов (правило **implicit drop**). Абоненты, попавшие под действие правила **implicit drop**, не будут отображены в глобальной таблице абонентов. Это экономит место в самой таблице, а также защищает от атак на переполнение таблицы. Поэтому настоятельно НЕ рекомендуется создавать пустую последовательность (без команды **match**) в **subscriber-map** вида:

```
subscriber-map TEST 30
set idle-timeout 30
set session-timeout 1440
set service 2Mb
```



В таком случае попытки аутентификации всех абонентов будут успешными и информация о каждом клиенте появится в глобальной таблице!!!

Так называемый сервис по умолчанию, когда существует общее правило для большинства сессий, от провайдера к провайдеру сильно отличается. Гибкость карты абонента позволяет сетевым администраторам использовать широкий спектр сценариев для обслуживания абонентских сессий и настройки поведения по умолчанию.

При наличии правила **match** в последовательности карты абонентов с префиксным списком, не существующим в маршрутизаторе, последовательность игнорируется.

В нескольких последовательностях карт абонентов может быть несколько правил **match**, в таком случае в последовательности работает логическое правило «ИЛИ». Обратите внимание, что префиксные списки могут меняться и дополняться отдельно от карт абонентов. Изменения в префиксных списках, применённых в карте абонентов, могут вызывать изменения логики действия карты, будьте аккуратны.

Приведём пример неаккуратного! изменения правила в последовательности.

```
ecorouter(config)#subscriber-map TEST 10
ecorouter(config-subscriber-map)#no match dynamic prefix-list A
ecorouter(config-subscriber-map)#match dynamic prefix-list B
```

Это вызовет пересчёт всей логики в карте **TEST**, т.к. при введении срабатывает неявное правило **match**.

Правильный вариант:

```
ecorouter(config)#subscriber-map TEST 10
ecorouter(config-subscriber-map)#match dynamic prefix-list B
ecorouter(config-subscriber-map)#no match dynamic prefix-list A
```

29.9.2 PPPoE

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.

12) Задать адрес на **bmi**-интерфейсе из PPPoE-подсети и включить интерфейс **bmi** в протокол динамической маршрутизации так же, как и обычный IP-интерфейс.

13) 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.

29.10 Логирование абонентских сессий

Для отслеживания установления абонентской сессии служит команда режима администрирования **debug subscriber**.

Таблица 118 — Параметры команды **debug subscriber**

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 recieved 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 recieved 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 recieved for client 00:50:79:66:68:01
[data-plane] [PETROV] DHCP-REQUEST message recieved from client
00:50:79:66:68:01
[data-plane] [PETROV] DHCP-ACKNOWLEDGE message recieved 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
[debug] [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 recieved 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 recieved for client 00:50:79:66:68:01
[data-plane]      [VLAN10]  DHCP-REQUEST message recieved from client
00:50:79:66:68:01
[data-plane]      [VLAN10]  DHCP-ACKNOWLEDGE message recieved for client
00:50:79:66:68:01
[data-plane]      [VLAN10]  DHCP-DISCOVER message recieved from client
00:50:79:66:68:02
[data-plane] [VLAN10] DHCP-OFFER message recieved for client 00:50:79:66:68:02
[data-plane]      [VLAN10]  DHCP-REQUEST message recieved from client
00:50:79:66:68:02
[data-plane]      [VLAN10]  DHCP-ACKNOWLEDGE message recieved 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.

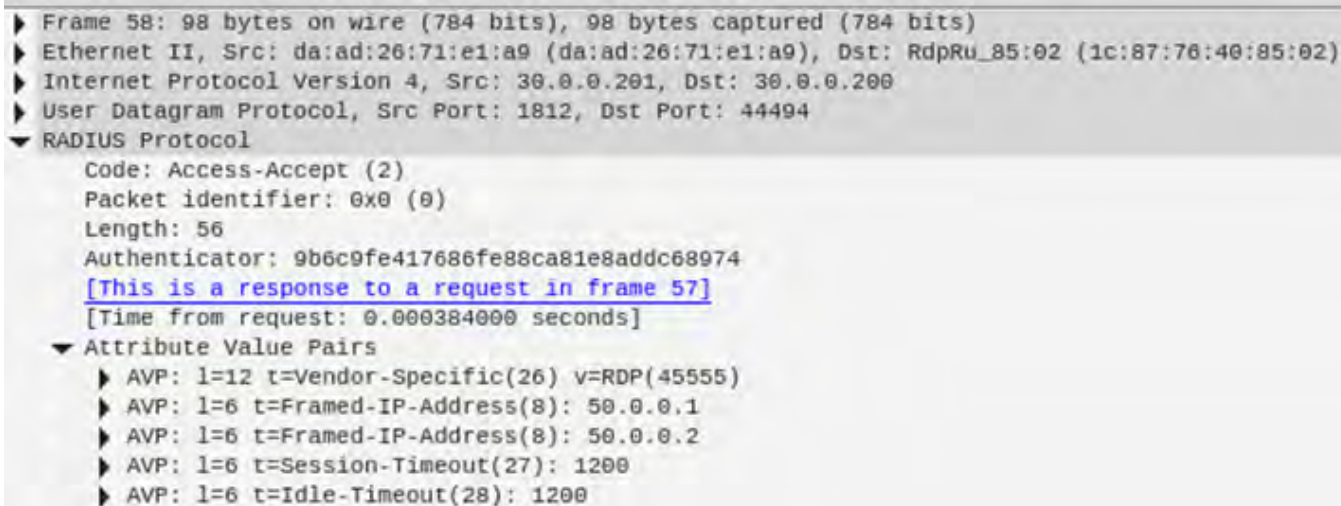


29.11 Общие сервисы

Настройка общего сервиса (Shared Contract) для нескольких абонентов, где общая полоса пропускания делится между абонентами, доступна для типов подключения IPoE L2/L3 и PPPoE. Для включения общего сервиса в IPoE используется команда в режиме конфигурирования **subscriber-map**:

```
eco(config-sub-map)#shared-service key ?
agent-option      DHCP opt.82 or PPPoE IA as key for shared subscriber-service
framed-ip         Creating sessions from list of Framed-IP and key for shared
                  subscriber-service
radius-attribute   Radius vendor-specific attribute 251 as key for shared
                  subscriber-service
vlan              VLAN as key for shared subscriber-service
```

Ключом для создания общего контракта может быть одинаковый VLAN, в котором располагаются абоненты, DHCP-опция 82 при передаче сообщений DHCP discover от абонентов, список атрибутов Framed-IP-Address с IP-адресами абонентских устройств в сообщении RADIUS Access-Accept, а также дополнительный 251 RADIUS-атрибут **Named_Master_of_SLA_ER**. Ниже приведён пример сообщения Access-accept от RADIUS-сервера со списком Framed-IP-Address.



```
▶ Frame 58: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
▶ Ethernet II, Src: da:ad:26:71:e1:a9 (da:ad:26:71:e1:a9), Dst: RdpRu_85:02 (1c:87:76:40:85:02)
▶ Internet Protocol Version 4, Src: 30.0.0.201, Dst: 30.0.0.200
▶ User Datagram Protocol, Src Port: 1812, Dst Port: 44494
▼ RADIUS Protocol
  Code: Access-Accept (2)
  Packet identifier: 0x0 (0)
  Length: 56
  Authenticator: 9b6c9fe417686fe88ca81e8addc68974
  [This is a response to a request in frame 57]
  [Time from request: 0.000384000 seconds]
  ▼ Attribute Value Pairs
    ▶ AVP: 1=12 t=Vendor-Specific(26) v=RDP(45555)
    ▶ AVP: 1=6 t=Framed-IP-Address(8): 50.0.0.1
    ▶ AVP: 1=6 t=Framed-IP-Address(8): 50.0.0.2
    ▶ AVP: 1=6 t=Session-Timeout(27): 1200
    ▶ AVP: 1=6 t=Idle-Timeout(28): 1200
```

Рисунок 61

Общий сервис возможен только для абонентов, авторизованных исключительно через удалённый RADIUS-сервер. При использовании локальных функции AAA на BRAS общий сервис не применится. При использовании ключа **framed-ip** для корректной работы процедуры RADIUS Change of Authorization, сообщения RADIUS CoA от RADIUS-клиента

должны содержать тот же список атрибутов Framed-IP-Address, что и Access-Accept сообщение.

Для PPPoE команда настройки общего сервиса выглядит аналогично, только в режиме конфигурации PPPoE профайла.

Абонентские сессии с общим сервисом в глобальной абонентской таблице отображаются с флажками «**SR>**» (**R** — remote authorization (radius attribute **SERVICE_NAME**), **S** — shared subscriber-service between subscribers), **>** — active and valid session).

Более детальную информацию по сервисам у абонентов можно получить с помощью команды `show subscribers bmi.X service`, где **bmi.X** — имя и номер BRAS-интерфейса bmi. У абонентов с общим сервисом Service-ID в выводе команды должен быть одинаковым.

Принцип работы приоритетов в **subscriber-map** (номера seq) позволяет гибко выделять IP-подсеть — абонентов, для которых разрешен или запрещен общий сервис.

Специфичный 251 RADIUS-атрибут **Nameld_Master_of_SLA_ER** даёт некоторые расширенные возможности и удобства при работе с одним сервисом для нескольких абонентов. Помимо того, что этот 251 атрибут (тип строка) может быть ключом для создания общего сервиса, как и упомянутые ранее VLAN, Framed-IP-Address и DHCP-опция 82, он же может использоваться в качестве дополнительного описания для общего сервиса.

Например, если выбрать в качестве ключа для общего сервиса Framed-IP-Address и включить в сообщения от RADIUS-сервера специфичный 251 атрибут (например, номер договора), то на BRAS для общего сервиса, помимо его имени и ID, появится дополнительное описание в командах группы **show** (значение поля **Sharing Description**).

```
ecorouter#sh subscribers bmi.2 service
VRF: default
Total subscribers: 2
Accepted: 2, Rejected: 0, Authenticating: 0, DHCP conversation: 0
Codes:
> - active and valid session
B - blocked by IP Source Guard
F - authentication during Radius unavailable
L - local authorization (subscriber-service)
N - not specified
R - remote authorization (radius attribute SERVICE_NAME)
S - shared subscriber-service between subscribers
U - unknown (internal error)
l - local authentication (prefix-list)
r - remote authentication (subscriber-aaa)
```



```
s - single subscriber for shared subscriber-service
Keys for sharing service:
RA - Radius Attribute 251
FIP - List of Framed IP Address attributes
VLAN - C-VLAN and S-VLAN number
OPT82 - DHCP option 82
```

IP Address	MAC Address	Service	Shared Key	Sharing Description	Service ID

SR> 50.0.0.1	0050.7966.6805	coa_test	FIP	dogovor	#1703
0x00000037					
SR> 50.0.0.2	0050.7966.6800	coa_test	FIP	dogovor	#1703
0x00000037					

Для того, чтобы отсортировать абонентов с одинаковым описанием (**Sharing Description**) введите команду: `show subscribers bmi.2 service description LINE`, где **LINE** — точное совпадение строки в 251 атрибуте (например, **dogovor #1703**) или воспользуйтесь функциями **grep**.

Например:

`show subscribers bmi.2 service | grep PATTERN`, где **PATTERN** — шаблон для поиска в выводе.

29.12 Удалённые абонентские сети в среде MPLS

В EcoRouterOS есть возможность подключить через BRAS удалённых абонентов. Рассматривается сценарий, когда удалённая абонентская сеть доступна через MPLS облако.

В роли транспорта будет выступать pseudowire соединение. Таким образом от удалённой сети будет приходить оригинальный L2 трафик. Т. е. наряду с IP трафиком для IPoE сессий будут возможны подключения DHCP, PPPoE.

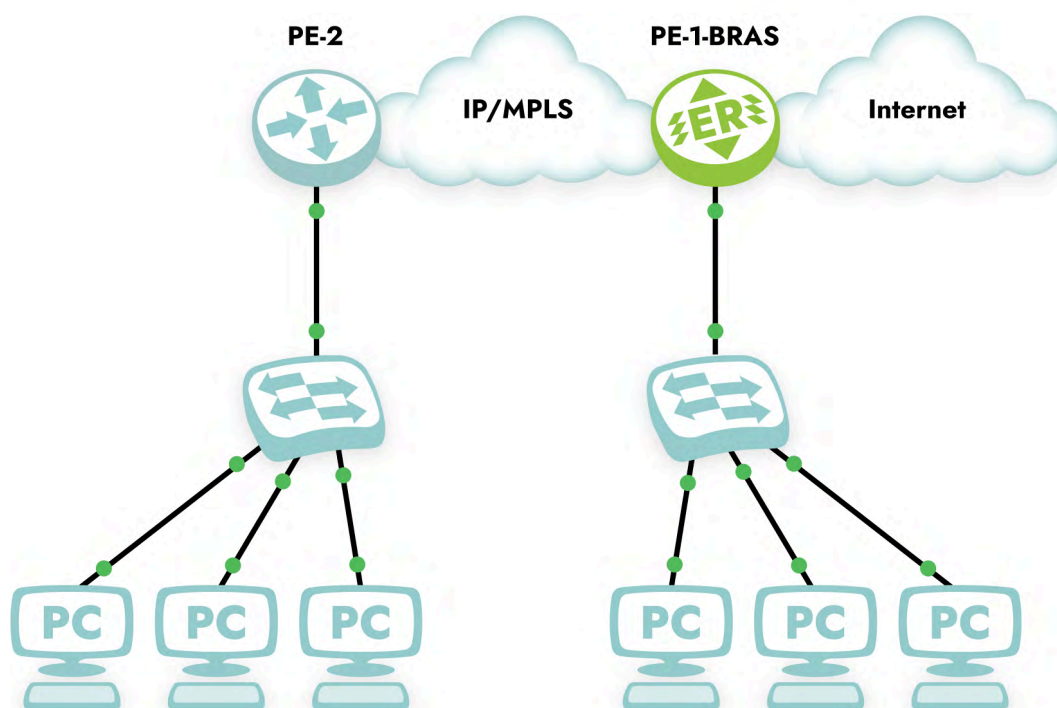


Рисунок 62

Пример такой топологии представлен на рисунке.

Удалённые абоненты подключены к PE2. Создание сессий будет происходить на PE1-BRAS.

У PE1-BRAS так же есть локальная сеть с абонентами.

Поддержка такой топологии на EcoRouterOS возможна благодаря наличию специальных виртуальных портов. Порты создаются парами и непосредственно соединяются друг с другом.

На PE2 настраивается классический pseudowire на порте, который подключен к абонентской сети. (см. Настройка L2-circuit)

На PE1-BRAS настраивается IPoE/PPPoE сервер широкополосного доступа (см.)

Для подключения удалённых абонентов необходимо установить pseudowire к PE2 до удалённой сети. В качестве локального порта будет служить один порт из виртуальной пары портов.

```
mpls l2-circuit vc1 1 2.2.2.2
!
router ldp
pw-status-tlv
targeted-peer ipv4 2.2.2.2
exit-targeted-peer-mode
transport-address ipv4 1.1.1.1
!
```

```
port virt.0
virtual-network pair virt.1
service-instance vc1
encapsulation untagged
mpls-l2-circuit vc1 primary
```

Второй порт из виртуальной пары присоединяется к интерфейсу bmi

```
port virt.1
virtual-network pair virt.0
service-instance bmi
encapsulation untagged
!
interface bmi.1
connect port virt.1 service-instance bmi
```

В итоге на bmi интерфейсе заведутся удалённые абонентские сессии через виртуальный порт

IP Address	MAC Address	Port	S-tag	C-tag	Status	Type

L> 192.168.1.2	0050.7966.6800	virt.1	--	--	accepted(1)	IPoE L2

30 L2TP

Протокол туннелирования уровня 2 (L2TP) — обычно используется для передачи сессий PPP от инициатора, известного как концентратор доступа L2TP (L2TP Access Concentrator, LAC), к серверу L2TP (L2TP Network Server, LNS). L2TP обычно используется для массового предоставления услуг широкополосного доступа абонентам. В этом сценарии LAC находится в сети провайдера и имеет подключение уровня 2 к мультиплексору доступа. LAC играет свою роль на этапе обнаружения (если используется PPPoE) и во время PPP согласования протокола управления соединением (Link Control Protocol, LCP). LAC также выполняет первоначальную аутентификацию абонента. Успешная аутентификация, обычно от RADIUS, указывает LAC, что кадры PPP от этого абонента должны быть туннелированы на LNS по указанному IP-адресу. Затем LAC туннелирует кадры PPP от этого абонента через туннель L2TP на LNS, где сессия PPP фактически завершается. Другими словами, для переноса сессий PPP требуются туннели L2TP.

L2TP использует два типа сообщений: управляющие сообщения и сообщения с данными. Управляющие сообщения используются для установления, обслуживания и разрыва туннелей и сессий. Для обеспечения расширяемости и максимальной совместимости полезная нагрузка управляющих сообщений кодируется с использованием пар атрибутов и значений (Attribute Value Pairs, AVP), некоторые из которых применимы ко всем управляющим сообщениям, а некоторые — к конкретным управляющим сообщениям. Заголовок L2TP содержит поля порядковых номеров, которые должны присутствовать в управляющих сообщениях для обеспечения надёжного канала управления L2TP, гарантирующего доставку. Сообщения с данными используются для инкапсуляции кадров PPP, передаваемых по туннелю. Сообщения с данными не передаются повторно в случае потери пакетов.

L2TP имеет общий фиксированный формат заголовка как для управляющих сообщений, так и для сообщений с данными, а бит типа (Т) в заголовке используется для указания того, является ли пакет управляющим (1) или сообщением с данными (0). Пакет L2TP затем передаётся в транспортном протоколе, и хотя спецификации позволяют напрямую инкапсулировать L2TP через Frame Relay, ATM и UDP/IP, именно последний вариант используется практически повсеместно.

30.1 Установление управляющего туннеля L2TP

Перед передачей каких-либо пользовательских данных, LAC (инициатор) и LNS (терминатор) должны договориться о параметрах логического канала, который будет использоваться для управления. Этот процесс начинается с того, что LAC отправляет LNS пакет **SCCRQ** (Start-Control-Connection-Request) по **UDP-порту 1701**. В этом запросе



содержатся базовые сведения об отправителе, такие как его имя хоста, а также список поддерживаемых атрибутов протокола, версия L2TP и предлагаемый идентификатор туннеля. Получив запрос, LNS формирует ответ **SCCRP** (Start-Control-Connection-Reply), в котором подтверждает или корректирует параметры и указывает собственный идентификатор туннеля. После обмена этими сообщениями LAC отправляет финальное подтверждение **SCCCN** (Start-Control-Connection-Connected), что завершает так называемое "рукопожатие" — туннель считается установленным.

Это управляющее соединение носит долгосрочный характер и существует отдельно от пользовательских сессий. Оно служит исключительно для обмена служебными сообщениями: поддержания связи (keepalive), уведомлений об ошибках и, что самое важное, для управления жизненным циклом пользовательских сессий внутри этого туннеля. Для поддержания активности туннеля узлы периодически асинхронно обмениваются служебными пакетами **Hello** и **ZLB (Zero-Length Body)**.

30.2 Подготовка пользовательской сессии PPP

После создания туннеля L2TP становится возможным начать процесс установления сеанса PPP. Для установления сеанса в туннеле L2TP используется трёхсторонний обмен управляющими сообщениями, состоящий из входящего вызова **ICRQ** (Incoming Call Request), ответа на входящий вызов **ICRP** (Incoming Call Reply) и сообщения о соединении **ICCN** (Incoming Call Connected). Поскольку это управляющие сообщения, все они явно подтверждаются с помощью встроенных подтверждений или подтверждений ZLB.

Сообщение ICRQ отправляется от LAC к LNS, чтобы сообщить о получении входящего вызова (сеанса PPP) и о необходимости установления сеанса между двумя узлами в рамках этого вызова. Сообщение ICRQ предоставляет LNS достаточно информации о вызове, чтобы принять решение об ответе на него. Сообщение ICRQ содержит AVP типа сообщения **Message Type** и назначенного идентификатора сессии **AVP Assigned Session ID**, а также AVP серийного номера вызова **Call Serial Number**, которая может использоваться как на LAC, так и на LNS в качестве удобного указателя на вызов для устранения неполадок. ICRQ также может переносить дополнительные AVP, включая AVP номера вызывающего абонента **Calling Number** и информации о линии доступа (смотри RFC 5515), такие как идентификатор канала, удалённый идентификатор, фактическая скорость передачи данных в восходящем направлении и фактическая скорость передачи данных в нисходящем направлении.

ICRP отправляется от LNS к LAC в ответ на ICRQ, чтобы указать, что параметры в ICRQ были приемлемы и что LAC может продолжить вызов. ICRP содержит только два AVP: тип сообщения **Message Type** и присвоенный идентификатор сессии **Assigned Session ID**. Значения присвоенного идентификатора сессии являются локальными для каждого абонента и не являются согласованными или утверждёнными.



Последним сообщением в трехстороннем обмене данными, используемом для установления сеансов внутри туннеля, является ICCN. Оно отправляется от LAC к LNS, чтобы указать, что вызов был принят, и сеанс L2TP переходит в состояние "установлен" (established). Оно также предоставляет дополнительную информацию о параметрах, которые были использованы для ответа на вызов и которые могли быть недоступны на момент отправки ICRQ (хотя, в большинстве случаев они будут доступны). Как минимум, ICCN должен содержать AVP типа сообщения Message Type, типа кадрирования Framing Type и скорости соединения TX Connect Speed. TX Connect Speed определяет скорость в битах в секунду с точки зрения трафика, поступающего от LAC к абоненту (т.е. скорость нисходящего потока LAC) и, для максимальной точности, может быть получена LAC из тестов PPP Broadband Forum Access Line Characteristic, вставленных узлом доступа. Скорость соединения TX может быть полезна для косвенной установки агрегированной скорости иерархического QoS (H-QoS). Это косвенный процесс, поскольку LNS не может напрямую определить и установить суммарную скорость на основе параметра TX Connect Speed AVP, а вместо этого скорость TX Connect Speed передаётся на RADIUS-сервер (используя параметр include-radius-attribute access-loop-option в политике аутентификации), который, в свою очередь, может передать суммарную скорость на LNS в VSA (Vendor Specific Attributes) с переопределением QoS.

Также в сообщении ICCN может присутствовать ряд дополнительных параметров AVP, предоставляющих информацию из процесса согласования протокола управления соединением LCP (Link Control Protocol) между LAC и абонентом. К ним относятся Initial Receive, Last Transmit и Last Receive LCP Config Requests, вместе с Proxy Authentication Type, Name, Challenge, и Response. Эти параметры позволяют LNS либо принудительно пересогласовать LCP, либо продолжить сеанс PPP и перейти к фазе IPCP. Последний параметр AVP, присутствующий в ICCN, — это параметр RX Connect Speed AVP, который противоположен параметру TX Connect Speed и определяет скорость в битах в секунду с точки зрения трафика, поступающего от абонента к LAC.

После завершения трехстороннего обмена управляющими сообщениями, необходимого для установления сессии, LNS аутентифицирует пользователя. RADIUS возвращает стандартные и специфичные для поставщика атрибуты, ранее определённые в файле пользователей. Успешная аутентификация позволяет LNS перейти к фазе IPCP (IP Control Protocol, RFC 1332) с абонентом. После завершения фазы IPCP создаётся экземпляр абонента, и сеанс L2TP становится активным. Параметры Tunnel-ID и Session-ID представляют собой локально сгенерированные числа, которые передаются в управляющих сообщениях L2TP.

30.3 Пользовательская сессия PPP

Сессия PPP состоит из шести фаз.

30.3.1 "Мёртвая" фаза (Dead Phase)

Физический уровень недоступен во время "мёртвой" фазы. PPP-соединение начинается и заканчивается этой фазой.

Когда два устройства обнаруживают, что физическое соединение между ними активировано, например, когда на физическом канале обнаруживаются несущие сигналы, оба устройства переходят из "мёртвой" фазы в фазу установления соединения.

После завершения PPP-соединения оба устройства снова переходят в "мёртвую" фазу.

30.3.2 Фаза установления соединения

В фазе установления соединения два устройства выполняют согласование LCP для определения режима работы SP (Single-link PPP) или MP (Multilink PPP), MRU, режима аутентификации и "волшебного числа" (Magic Number). После завершения согласования LCP (Link Control Protocol) два устройства переходят к следующему этапу. На этапе установления соединения статус LCP изменяется следующим образом:

- если канал недоступен (в "мёртвой" фазе), LCP находится в "Начальном" состоянии или состоянии "Запуск". Когда физический уровень обнаруживает доступность канала, он отправляет событие "Up" на канальный уровень. После получения события канальный уровень изменяет статус LCP на "Запрос отправлен". Затем устройства на обоих концах отправляют друг другу пакеты **Configuration Request** для настройки канала передачи данных.
- Если локальное устройство сначала получает пакет **Configure-Ack** от другого устройства, статус LCP изменяется с "Запрос отправлен" на "Подтверждение получено". После того, как локальное устройство отправляет пакет **Configuration Ack** другому устройству, статус LCP изменяется с "Подтверждение получено" на Open.
- Если локальное устройство сначала отправляет пакет Configure-Ack удалённому устройству, статус LCP изменяется с Request-Sent на Ack-Sent. После того, как локальное устройство получает пакет Configure-Ack от удалённого устройства, статус LCP изменяется с Ack-Sent на Open. После перехода LCP в состояние

Open начинается следующая фаза. Следующая фаза — это фаза аутентификации или сетевой фазы, в зависимости от того, требуется ли аутентификация.

Наличие и тип (PAP/CHAP) аутентификации согласуются в сообщениях **Configuration Request** и **Configuration Ack**.

30.3.3 Фаза аутентификации

Фаза аутентификации является необязательной. По умолчанию PPP не выполняет аутентификацию во время установления PPP-соединения. Если аутентификация требуется, протокол аутентификации должен быть указан в фазе установления соединения.

PPP предоставляет два режима аутентификации по паролю: аутентификация PAP и аутентификация CHAP.

Доступны два метода аутентификации: односторонняя и двусторонняя. При односторонней аутентификации устройство на одном конце выступает в роли аутентифицирующего устройства, а устройство на другом конце — в роли аутентифицируемого устройства. При двусторонней аутентификации каждое устройство выступает одновременно и в роли аутентифицирующего, и в роли аутентифицируемого устройства. На практике используется только односторонняя аутентификация.

PAP аутентификация

PAP (Password Authentication Protocol) — это протокол двухшаговой аутентификации, передающий пароли в простом текстовом виде. Аутентифицируемое устройство отправляет локальное имя пользователя и пароль аутентифицирующему устройству. Аутентифицирующее устройство проверяет, находится ли полученное имя пользователя в списке пользователей. Если полученное имя пользователя присутствует в списке пользователей, аутентифицирующее устройство проверяет правильность полученного пароля.

Если пароль правильный, аутентификация проходит успешно. Если пароль неправильный, аутентификация завершается неудачно. Если полученное имя пользователя отсутствует в списке пользователей, аутентификация завершается неудачно.

Пакет PAP инкапсулирован в поле Information пакета PPP со значением поля Protocol 0xC023.

Пакет PAP содержит следующие поля:

- В поле Code значение 0x01 означает сообщение **Authenticate-Request**, 0x02 означает сообщение **Authenticate-Ack**, 0x03 означает сообщение **Authenticate-Nak**.
- Поле identifier — порядковый номер сообщения для его идентификации.

- Поле Length — длина PAP-пакета, включая длины полей Code, Identifier, Length и Data.
- Поле Data — содержит передаваемые данные в зависимости от поля Code.

CHAP аутентификация

CHAP (Challenge Handshake Authentication Protocol) — это протокол трёхшаговой аутентификации, когда в открытом виде передаются только имена пользователей, но не пароли, что обеспечивает бóльшую безопасность.

Однонаправленная аутентификация CHAP применяется в двух сценариях (рекомендуется первый сценарий, чтобы аутентифицированное устройство могло проверить имя пользователя аутентифицирующего устройства).

Первый сценарий, когда на аутентифицирующем устройстве настроено имя пользователя.

В этом сценарии аутентифицирующее устройство инициирует запрос на аутентификацию, отправляя аутентифицируемому устройству пакет Challenge, содержащий случайное число и локальное имя пользователя.

После получения пакета Challenge через интерфейс аутентифицируемое устройство проверяет, настроен ли пароль CHAP на интерфейсе. Если пароль настроен, аутентифицируемое устройство использует алгоритм хеширования для вычисления хеш-значения на основе идентификатора пакета, пароля CHAP и случайного числа в пакете, а затем отправляет аутентифицирующему устройству пакет Response, содержащий хеш-значение и локальное имя пользователя.

Если пароль не задан, аутентифицируемое устройство ищет пароль, соответствующий имени пользователя аутентифицирующего устройства в полученном пакете, использует алгоритм хеширования для вычисления хеш-значения на основе идентификатора пакета, пароля, соответствующего имени пользователя, и случайного числа в пакете, а затем отправляет Response-пакет, содержащий хеш-значение и локальное имя пользователя, аутентифицирующему устройству.

Аутентифицирующее устройство использует алгоритм хеширования для вычисления хеш-значения на основе идентификатора пакета, локально сохраненного пароля аутентифицируемого устройства и случайного числа в пакете Challenge, а затем сравнивает хеш-значение с хеш-значением в пакете Response. Если два хеш-значения совпадают, аутентификация проходит успешно. В противном случае аутентификация завершается неудачей.

Если на аутентифицирующем устройстве не известно имя пользователя, аутентифицирующее устройство инициирует запрос на аутентификацию, отправляя аутентифицируемому устройству пакет Challenge, содержащий случайное число.

После получения пакета запроса (Challenge packet) аутентифицируемое устройство использует алгоритм хеширования для вычисления хеш-значения на основе идентификатора пакета, пароля CHAP и случайного числа в пакете, а затем отправляет ответный пакет (Response packet), содержащий хеш-значение и имя пользователя, аутентифицирующему устройству.

Аутентифицирующее устройство использует алгоритм хеширования для вычисления хеш-значения на основе идентификатора пакета, локально сохранённого пароля аутентифицируемого устройства и случайного числа в пакете запроса, а затем сравнивает хеш-значение с хеш-значением в пакете ответа. Если оба хеш-значения совпадают, аутентификация проходит успешно. В противном случае аутентификация завершается неудачно.

Пакет CHAP содержит следующие поля:

- В поле Code значение 0x01 означает сообщение **Challenge**, 0x02 означает сообщение **Response**, 0x03 означает сообщение **Success**, 0x04 означает сообщение **Failure**.
- Identifier — порядковый номер сообщения для сопоставления **Challenge** и **Response** (какой запрос к какому ответу относится).
- Length — длина пакета CHAP, включая длину полей Code, Identifier, Length и Data.
- Data — содержит передаваемые данные в зависимости от поля Code.

30.3.4 Сетевая фаза

Фаза Network Layer Protocol, или сетевая фаза протокола PPP, является заключительным этапом установления соединения перед началом передачи пользовательских данных. Её основная цель — независимо настроить параметры для каждого сетевого протокола, который должен работать поверх уже созданного и аутентифицированного PPP-канала. Для этого используются специализированные подпротоколы, объединённые общим названием NCP (Network Control Protocol). Каждый сетевой протокол имеет свой NCP: например, IPCP для IPv4, IPv6CP для IPv6 или IPXCP для устаревшего IPX.

Процесс согласования для каждого NCP следует чёткой последовательности, аналогичной фазе установления канала LCP. Инициатор, которым часто выступает клиентское оборудование, отправляет пакет **Configure-Request**, содержащий список опций для настройки. В случае с IPCP это обычно запрос на назначение IP-адреса (часто обозначаемый как 0.0.0.0), адресов DNS-серверов и, возможно, параметров сжатия заголовков. Удалённая сторона анализирует запрос и может ответить несколькими способами: **Configure-Ack** для подтверждения приемлемой конфигурации, **Configure-**

Nak для предложения альтернативных значений, или **Configure-Reject** для отклонения неподдерживаемых опций. Сетевая фаза не является строго одноразовым событием. В течение всей жизни соединения NCP может быть повторно активирован для динамического обновления параметров, например, при получении новых адресов DNS.

Завершение использования конкретного протокола также управляется через NCP с помощью обмена пакетами **Terminate-Request** и **Terminate-Ack**, что не обязательно приводит к разрыву всего PPP-соединения. На практике успешное завершение сетевой фазы, особенно для IPCP, является тем моментом, когда на интерфейсе маршрутизатора появляется IP-адрес и устанавливаются маршруты, делая PPP-канал полностью работоспособным для передачи данных.

30.3.5 Завершающая (Termination) фаза

Завершающая фаза (Termination) протокола PPP — это контролируемый процесс разрыва установленного соединения. Её инициирует любая из сторон, когда необходимо завершить сессию — будь то плановое отключение, потеря физической связи, тайм-аут или неудачная аутентификация. Ключевым протоколом на этом этапе снова выступает LCP (Link Control Protocol), который обеспечивает упорядоченный демонтаж канала.

Устройство, инициирующее разрыв, отправляет пакет **LCP Terminate-Request**. Получив его, противоположная сторона обязана ответить пакетом **LCP Terminate-Ack**, подтверждая получение запроса на разрыв. Этот обмен гарантирует, что обе стороны синхронно понимают факт завершения сессии, предотвращая состояния «полуразорванного» соединения.

После успешного обмена Terminate-Request и Terminate-Ack протокол PPP выполняет ряд важных финальных действий. Он уведомляет все активные NCP (Network Control Protocol), такие как IPCP, о необходимости завершить свою работу, что приводит к удалению сетевых настроек (например, IP-адресов). Затем соединение официально возвращается в исходное состояние — в "мёртвую" фазу, ожидая нового сигнала физического уровня для возможного последующего установления связи.

30.4 Передача данных

С момента установления пользовательской сессии PPP-кадры будут передаваться внутри L2TP-туннеля. LAC выполняет инкапсуляцию: он "оборачивает" исходный PPP-кадр (содержащий, например, IP-пакет) в заголовок L2TP. Этот заголовок содержит два ключевых поля: **Tunnel ID** и **Session ID**. Tunnel ID позволяет маршрутизаторам в сети правильно направить пакет к нужному LNS, а на самом LNS именно Session ID указывает, какому конкретному пользователю и виртуальному интерфейсу принадлежат данные.



Полученный L2TP-пакет затем помещается в стандартную UDP-дейтаграмму и передаётся по IP-сети. LNS выполняет обратную операцию: принимает UDP-пакет, по Tunnel ID находит нужный туннель, а по Session ID — конкретную сессию, после чего извлекает исходный PPP-кадр, обрабатывает его и направляет "чистый" IP-трафик во внутреннюю сеть.

30.5 Настройка L2TP

Для простой настройки RSVP-TE на EcoRouter достаточно:

- Настроить IP связность между LNS и LAC.
- Настроить IP связность с сервером AAA.
- Настроить профили абонентов и параметры в них.
- Настроить опции PPP.
- Настроить профиль L2TP.
- Настроить VMI интерфейс.

30.5.1 Настройка опций PPP

Для обработки PPP-сессий абонентов необходимо настроить набор опций PPP с помощью команды конфигурационного режима `ppp-options <NAME>`, где **NAME** — локальное наименование профиля.

Команды в контексте `ppp-options` :

- `description <LINE>` — задаёт строку описания для данного набора опций PPP.
- `set aaa <NAME>` — подключает профиль абонента внешнего сервера AAA (Authentication Authorization Accounting), созданный по команде `subscriber-aaa`. В свою очередь, `subscriber-aaa` ссылается на настройки `radius-group`.
- `set subscriber-service <NAME>` — подключает абонентский сервис, который будет применён в случае отсутствия сервиса от сервера AAA.
- `lcp renegotiation` — включает функцию повторного согласования параметров LCP напрямую с клиентом.
- `authentication {pap | chap | ms-chap | ms-chap-v2}` — задаёт набор (можно указать несколько протоколов) поддерживаемых способов аутентификации.

- **pool ipv4 <NAME>** — подключает созданный с помощью команды **ip pool <NAME>** пул IP-адресов и переводит в контекст настройки пула. В случае отсутствия со стороны сервера AAA соответствующих параметров могут быть заданы параметры по умолчанию:
 - **gateway <A.B.C.D>** — передаёт абонентам адрес шлюза по умолчанию.
 - **dns <A.B.C.D>** — передаёт абонентам DNS IP-адрес
- **auth-req-limit <1-100>** — задаёт количество полученных от абонента запросов на аутентификацию, после которого на сервер AAA будет передан повторный запрос на аутентификацию. До достижения указанного лимита повторных запросов, система будет ожидать ответ от сервера AAA на предыдущий запрос абонента. Значение по умолчанию равно 10.
- **idle-timeout <0-1440>** — задаёт максимальное время простоя сессии в минутах (отсутствие какого-либо трафика со стороны абонента), после которого сессия будет завершена. Значение по умолчанию равно 30 минутам.
- **max-configure <1-20>** — задаёт максимальное количество запросов параметров сессии (Configuration Request) в адрес абонента, после которого сессия будет завершена. Значение по умолчанию равно 10.
- **max-failure <1-10>** — задаёт максимально допустимое количество отказов (Nak) в получении параметров сессии, которое система может отправить, после чего будет отправлено сообщение Reject. Значение по умолчанию равно пяти. Отказ (Nak) отправляется абоненту, если поле запроса поддерживается, но его значение не может быть принято. Запрос параметров сессии может быть полностью отклонён (Reject) если поле в запросе не поддерживается или же если количество запросов с неудовлетворяющими значениями превысит количество, указанное в **max-failure**.
- **max-terminate <1-10>** — задаёт максимальное количество запросов на завершение сессии (Termination Request) без положительных ответов, после которого сессия будет завершена в одностороннем порядке. Значение по умолчанию равно единице.
- **mru-alignment** — включает работу функции выравнивая размера MRU (Maximum Receive Unit). Функция применяет наименьший MRU и выравнивает его размер в байтах так, чтобы он был кратен восьми. По умолчанию функция выключена.
- **new-session-min-interval <0-60>** — задаёт минимальный интервал времени в секундах до момента, когда абонент сможет повторно инициировать сессию. Значение по умолчанию равно 10 секундам.



- `session-timeout <0-71582780>` — время работы сессии в минутах, по истечению которого сессия будет завершена. Значение по умолчанию равно 1440 минутам.
- `shared-service key {agent-option | framed-ip | radius-attribute | vlan}` — устанавливает ключевой признак, по которому ряд пользовательских сессий будут обработаны общим сервисом (subscriber-service):
 - `agent-option` — общая DHCP опция 82 (Circuit ID + Remote ID) (для IPoE абонентов) или теги PPPoE IA (для PPP абонентов);
 - `framed-ip` — общий список адресов (Framed-IP-Address), назначенный сервером AAA;
 - `radius-attribute` — общий специфический для РДП Инновации RADIUS-атрибут 251 (Shared Service).
 - `vlan` — общий VLAN тег или комбинация sVLAN и cVLAN.
- `timeout-echo <1-10>` — задаёт промежуток времени в секундах между отправками Echo-сообщений. Значение по умолчанию равно 10.
- `max-echo <0-100>` — задаёт максимальное допустимое количество запросов Echo-Request без ответов, после которого сессия будет завершена. Значение по умолчанию равно пяти.
- `timeout-retry <1-10>` — задаёт максимально допустимое время ожидания ответа абонента на управляющие сообщения (кроме Echo-Request) в секундах, после которого последует повторная отправка управляющего сообщения. Значение по умолчанию равно трём секундам.
- `update-interval <0-1440>` — задаёт промежуток времени в минутах между отправками отчётных (Accounting) сообщений на сервер AAA. Значение по умолчанию равно пяти минутам.

30.5.2 Настройка профиля L2TP

Для работы L2TP-туннеля на маршрутизаторе необходимо создать и настроить соответствующий профиль. Это делается в конфигурационном режиме командой `l2tp-profile <NAME>`, где **NAME** — локальное наименование профиля.

Команды в контексте `l2tp-profile`:

- `description <LINE>` — задаёт строку описания для данного профиля L2TP.
- `set ppp-options <PPP>` — подключает набор опций PPP, созданный по команде `ppp-options <PPP>` к L2TP. Процесс установления соединения PPP, в

данном случае, является стадией процесса L2TP.

- `set allowed-ip <PREFIX_LIST>` — с помощью префиксного списка задаёт допустимые исходящие адреса принимаемых L2TP туннелей.
- `ipv4 address {A.B.C.D | any}` — задаёт IPv4-адрес назначения, с которым допускается принимать L2TP туннели. Значение **any** — позволяет принимать туннели с любым адресом назначения.
- `hostname <NAME>` — задаёт наименование маршрутизатора. Включается в управляющие сообщения L2TP. Служит для идентификации человеком при журналировании и отладке. Может также быть использовано для аутентификации и авторизации серверами AAA.
- `vendor-name <NAME>` — задаёт наименование производителя. Включается в управляющие сообщения L2TP. Служит для идентификации производителя оборудования. Служит для идентификации человеком при журналировании и отладке. При настройке мультивендорных сетей помогает правильно интерпретировать поведение туннеля.
- `secret <WORD>` — пароль для L2TP туннеля. Для работы туннеля на обоих его концах должен быть задан одинаковый пароль.
- `hello-interval <1-250>` — задаёт интервал отправки приветственных сообщений в секундах. Обеспечивает механизм поддержания жизнеспособности туннеля (keepalive). Значение по умолчанию равно 60 секундам.
- `max-retransmit count <1-250>` — задаёт максимальное количество повторных попыток передать управляющее сообщение. При неполучении подтверждающих сообщений управляющее сообщение отправляется повторно через время, определяемое `ack-timeout`. После исчерпания количества попыток сессия будет завершена. Значение по умолчанию равно пяти.
- `ack-timeout <1-250>` — задаёт время ожидания подтверждающего сообщения (ZLB) в секундах, по истечении которого произойдёт повторная отправка управляющего сообщения. Заданное время ожидания увеличивается вдвое после каждой повторной отправки управляющего сообщения. Значение по умолчанию равно одной секунде.
- `max-ack-timeout <8-250>` — задаёт предельное время ожидания подтверждающих сообщений ZLB (Zero-Length Body) в секундах. Когда время ожидания `ack-timeout` превысит значение заданное `max-ack-timeout`, управляющие сообщения продолжат отправляться с периодичностью `max-ack-`

`timeout` до исчерпания `max-retransmit count`. Значение по умолчанию равно 16 секундам.

- `receive-window-size <1-32768>` — задаёт размер окна приёма для управляющих сообщений. Удалённый узел может отправить указанное количество управляющих сообщений, после чего ему придётся ждать подтверждения для отправки следующих. Значение по умолчанию равно 16.
- `tunnel-negotiation wait-timeout <1-250>` — максимальное время на установления L2TP туннеля. Значение по умолчанию равно 60 секундам.

30.6 Конфигурирование L2TP LNS

Постройте схему в соответствии с рисунком ниже:

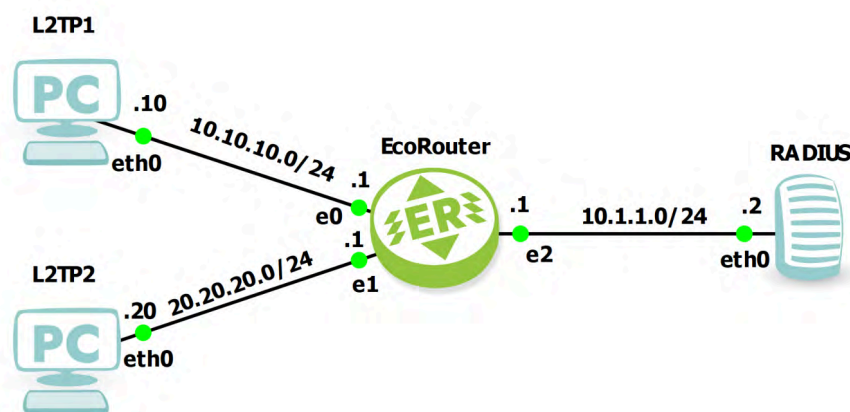


Рисунок 63

В данной схеме EcoRouter выполняет роль LNS (L2TP Network Server). LAC (L2TP Access Concentrator) отсутствует в схеме, его функции берут на себя абонентские устройства L2TP1 и L2TP2 выстраивая L2TP туннели до BMI-интерфейса на EcoRouter.

Настройте EcoRouter в качестве LNS

```
enable
configure

hostname EcoRouter

ip prefix-list ANY permit any

# Настройка параметров группы RADIUS-серверов
radius-group test
```

```
# Пароль pass1234 – идентичен настроенному на RADIUS
radius-server 10.1.1.2 secret pass1234 priority 10

subscriber-aaa test
  authentication radius test

ip pool POOL
  # Назначение диапазона IP-адресов для абонентов
  range 192.168.1.10-192.168.1.20

# Карта фильтрации разрешающая любой IPv4 трафик
filter-map policy ipv4 test 10
  match any any any
  set accept

# Настройка политики для абонентов
subscriber-policy test
  bandwidth in mbps 10
  bandwidth out mbps 10
  set filter-map in test
  set filter-map out test

# Настройка сервиса абонентов
subscriber-service test
  set policy test

ppp-options PPP
  # Подключить AAA-профиль
  set aaa test
  # Подключить сервис абонентов
  set subscriber-service test
  # Выбрать возможные способы аутентификации
  authentication pap chap
  # Подключить пул адресов абонентов
  pool ipv4 POOL 1
  # Задать шлюз по умолчанию
  gateway 192.168.1.1

l2tp-profile L2TP
  # Подключить профиль PPP
  set ppp-options PPP
  # Принимать любые адреса отправителя (source)
  set allowed-ip ANY
```



```
# Наименование производителя
vendor-name EcoRouter

port ge0
  service-instance bmi
  encapsulation untagged

port ge1
  service-instance e1
  encapsulation untagged

port ge2
  mtu 9728
  service-instance rad
  encapsulation untagged

interface bmi.1
  connect port ge0 service-instance bmi
  ip address 10.10.10.1/24
  # Подключение профиля L2TP
  set l2tp-profile L2TP

interface e1
  ip mtu 1500
  connect port ge1 service-instance e1
  ip address 20.20.20.1/24

interface rad
  ip mtu 1500
  connect port ge2 service-instance rad
  ip address 10.1.1.1/24
```

Настройте RADIUS

Сервер RADIUS настраивается так чтобы принимать любые запросы на авторизацию пользователей и присваивать им абонентский сервис test настроенный на EcoRouter.

Настройте интерфейс командой `ip a a 10.1.1.2/24 dev eth0` .
`/etc/raddb/clients.conf`

```
client router {
  ipaddr = 10.1.1.1
  secret = pass1234
}
```



/etc/raddb/dictionary

```
VENDOR RDP 45555
BEGIN-VENDOR RDP
ATTRIBUTE SERVICE_NAME 250 string
END-VENDOR RDP
```

/etc/raddb/users

```
DEFAULT      Auth-Type := Accept
              SERVICE_NAME += "test"
```

Настройте L2TP1

Настройте интерфейс командой `ip a a 10.10.10.10/24 dev eth0`.

/etc/ppp/pap-secrets

```
# Secrets for authentication using PAP
# client      server  secret IP addresses
"admin10" * "admin10" *
```

/etc/xl2tpd/xl2tpd.conf

```
[global]
auth file = /etc/ppp/pap-secrets
ipsec saref = no
force userspace = no
access control = no
debug avp = yes
debug network = yes
debug packet = yes
debug state = yes
debug tunnel = yes

[lac admin10]
lns = 10.10.10.1
redial = yes
redial timeout = 15
refuse chap = yes
require pap = yes
require authentication = yes
name = admin10
pppoptfile = /etc/ppp/options.l2tpd
autodial = yes
ppp debug = yes
```



/etc/ppp/options.l2tpd

```
persist
maxfail 0
nopcomp
noaccomp
defaultroute
replacedefaultroute
debug
noproxyarp
require-pap
refuse-eap
refuse-chap
refuse-mschap
refuse-mschap-v2
name admin10
noauth
```

Настройте L2TP2

Настройте интерфейс командой `ip a a 20.20.20.20/24 dev eth0`.

Настройте маршрут к интерфейсу VMI на EcoRouter командой `ip route add 10.10.10.1/32 via 20.20.20.1`.

/etc/ppp/pap-secrets

```
# Secrets for authentication using PAP
# client      server  secret IP addresses
"admin20" * "admin20" *
```

/etc/xl2tpd/xl2tpd.conf

```
[global]
auth file = /etc/ppp/pap-secrets
ipsec saref = no
force userspace = no
access control = no
debug avp = yes
debug network = yes
debug packet = yes
debug state = yes
debug tunnel = yes

[lac admin20]
```




```
lns = 10.10.10.1
redial = yes
redial timeout = 15
refuse chap = yes
require pap = yes
require authentication = yes
name = admin20
pppoptfile = /etc/ppp/options.l2tpd
autodial = yes
ppp debug = yes
```

/etc/ppp/options.l2tpd

```
persist
maxfail 0
nopcomp
noaccomp
defaultroute
replacedefaultroute
debug
noproxyarp
require-pap
refuse-eap
refuse-chap
refuse-mschap
refuse-mschap-v2
name admin20
noauth
```

Для запуска L2TP используйте команду `xl2tpd`.

Для перезапуска L2TP используйте последовательно команды `kill xl2tpd` и `xl2tpd`.

Проверьте работу построенной схемы

На L2TP2 для проверки соединения с LNS выполните `ping 10.10.10.1`.

На EcoRouter для проверки соединения с RADIUS выполните `ping 10.1.1.2`.

На L2TP1 и L2TP2 командой `ip a` проверьте появился ли PPP интерфейс.

На EcoRouter командой `show subscribers` проверьте наличие двух присоединённых абонентов: admin10 и admin20.



30.7 Конфигурирование L2TP LNS + LAC 1

Постройте схему в соответствии с рисунком ниже:

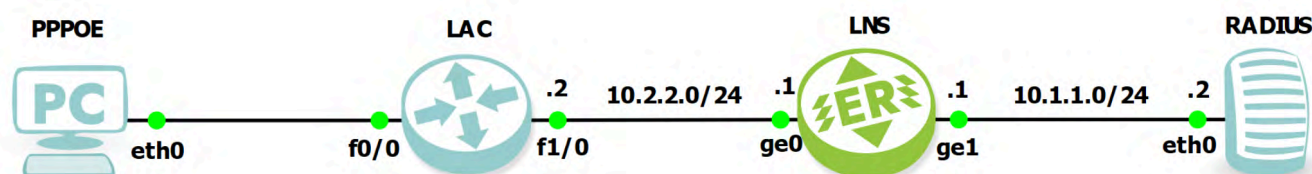


Рисунок 64

Настройте EcoRouter LNS

```
enable
configure

hostname LNS

radius-group RAD
# Пароль 1234 – идентичен настроенному на RADIUS
radius-server 10.1.1.2 secret 1234 priority 20

# Создание абонентского AAA-профиля
subscriber-aaa test
# Аутентификация и аккаунтинг будут проводиться
# с помощью серверов указанных в RADIUS-группе RAD
authentication radius RAD
accounting radius RAD

ip pool pppoe 1
# Назначение диапазона IP-адресов для абонентов
range 192.168.11.100-192.168.11.254

# Карта фильтрации разрешающая любой IPv4 трафик
filter-map policy ipv4 LAN 10
match any any any
set accept

# Префиксные списки разрешающие любые адреса
ip prefix-list ANY seq 5 permit any
ip prefix-list any seq 5 permit any
```

```
# Настройка политики для абонентов
subscriber-policy LAN 1
    bandwidth in mbps 100
    bandwidth out mbps 100
    set filter-map in LAN
    set filter-map out LAN

# Настройка сервиса абонентов
subscriber-service LAN
    set policy LAN

subscriber-map test 10
    match dynamic prefix-list any
    set subscriber-service LAN
    set aaa test

ppp-options PPP
    # Подключить AAA-профиль
    set aaa test
    # Подключить сервис абонентов
    set subscriber-service LAN
    # Включить повторное согласование LCP
    lcp renegotiation
    # Выбрать возможные способы аутентификации
    authentication pap chap
    # Подключить пул адресов абонентов
    pool ipv4 pppoe 1
        # Задать шлюз по умолчанию
        gateway 192.168.11.1

l2tp-profile L2TP
    # Подключить профиль PPP
    set ppp-options PPP
    # Принимать любые адреса отправителя (source)
    set allowed-ip ANY
    # Принимать любые адреса назначения (destination)
    ipv4 address any
    # Наименование маршрутизатора
    host-name LNS
    # Наименование производителя
    vendor-name LNS
    # Пароль для соединения с LAC
    secret tunnelpass
```



```
port ge0
  service-instance LAN
  encapsulation untagged

port ge1
  service-instance radius
  encapsulation untagged

interface bmi.1
  connect port ge0 service-instance LAN
  subscriber-map test
  ip address 10.2.2.1/24
  # Подключение профиля L2TP
  set l2tp-profile L2TP

interface radius
  connect port ge1 service-instance radius
  ip address 10.1.1.1/24
```

Когда на LNS в опциях PPP `ppp-options PPP` включено повторное согласование LCP `lcp renegotiation`, LNS игнорирует любые заранее согласованные параметры LCP и повторно согласовывает как параметры LCP, так и PPP-аутентификацию с PPP-клиентом.

Настройте RADIUS-сервер

На интерфейсе eth0 RADIUS-сервера установите IP-адрес 10.1.1.2 и маску 255.255.255.0.

`/etc/raddb/clients.conf`

```
client ecorouter {
  ipaddr = 10.1.1.1
  secret = 1234
}
```

Пароль в строке `secret = 1234` должен соответствовать указанному строке `radius-group RAD > radius-server` на LNS.

`/etc/raddb/dictionary`

```
VENDOR RDP 45555
BEGIN-VENDOR RDP
ATTRIBUTE SUBSCRIBER_POLICY 242 string
ATTRIBUTE FILTER_MAP_POLICY 247 string
ATTRIBUTE SERVICE_NAME      250 string
```



```
END-VENDOR RDP
ATTRIBUTE Class 25 octets
```

```
/etc/raddb/users
```

```
DEFAULT Auth-Type := Accept
SERVICE_NAME += "LAN", Framed-Pool += "pppoe"
```

Запустите сервер командой `radiusd -X`.

Настройте LAC

Приведённая ниже конфигурация реализована на базе маршрутизатора Cisco 7200.

```
enable
configure terminal

hostname LAC

vpdn enable
vpdn aaa attribute nas-ip-address vpdn-nas
vpdn search-order domain dnis

vpdn-group LAC
 request-dialin
  protocol l2tp
  domain rdpinn.ru
 initiate-to ip 10.2.2.1
 source-ip 10.2.2.2
 local name LAC
 l2tp tunnel password 0 tunnelpass

bba-group pppoe MAIN-BBA
 virtual-template 1

interface FastEthernet0/0
 no ip address
 duplex full
 pppoe enable group MAIN-BBA
 no shutdown

interface FastEthernet1/0
 ip address 10.2.2.2 255.255.255.0
 no shutdown
```



```
interface Virtual-Template1
description pppoe MAIN-BBA
no ip address
no peer default ip address
ppp mtu adaptive
ppp authentication chap
```

Настройте PPPOE

/etc/ppp/peers/provider

```
defaultroute
plugin rp-pppoe.so
eth0
user admin10@rdpinn.ru
usepeerdns
```

/etc/ppp/pap-secrets

```
# Secrets for authentication using PAP
# client      server  secret IP addresses
"admin10@rdpinn.ru" * "admin10" *
```

Запустите клиента командой `pon provider`.

Проверьте работу построенной схемы

На L2TP1 и L2TP2 командой `ip a` проверьте появился ли PPP интерфейс.

На EcoRouter командой `show subscribers` проверьте наличие присоединённого абонента admin10@rdpinn.ru.

30.8 Конфигурирование L2TP LNS + LAC 2

Постройте схему в соответствии с рисунком ниже:





```
enable
configure

hostname LNS

ip vrf VRF_RADIUS
    rd 3.3.3.3:50
    route-target both 65030:50

ip vrf VRF_LNS
    rd 3.3.3.3:3001
    route-target both 65030:3001

# Настройка параметров группы RADIUS-серверов
radius-group pfsense-2
    # Пароль lns – идентичен настроенному на RADIUS
    radius-server 172.16.2.1 secret lns vrf VRF_RADIUS priority 10 source
172.16.2.3

# Создание абонентского AAA-профиля
subscriber-aaa AAA_RADIUS
    # Аутентификация и аккаунтинг будут проводиться
    # с помощью серверов указанных в RADIUS-группе pfsense-2
    authentication radius pfsense-2
    accounting radius pfsense-2

ip pool INET 1
    # Назначение диапазона IP-адресов для абонентов
```



```
range 200.0.200.2-200.0.200.254
activate

# Карта фильтрации разрешающая любой IPv4 трафик
filter-map policy ipv4 ANY 10
match any any any
set accept

# Настройка BGP между LNS и LAC
router bgp 65030
  bgp router-id 172.3.3.3
  no bgp default ipv4-unicast
  bgp log-neighbor-changes
  address-family ipv4 vrf VRF_LNS
    max-paths ebgp 5
    redistribute connected
    neighbor 172.21.0.2 remote-as 65010
    neighbor 172.21.0.2 update-source 172.21.0.1
    neighbor 172.21.0.2 activate
    neighbor 172.21.0.2 route-map RM_/32 in
    neighbor 172.21.0.2 route-map RM_/32 out
  exit-address-family

ip prefix-list PL_ANY seq 5 permit any
ip prefix-list PS_/32 seq 10 permit 0.0.0.0/0 ge 31

# Настройка политик для абонентов
subscriber-policy SP_256K 1
  bandwidth in kbps 256
  bandwidth out kbps 256
  set filter-map in ANY
  set filter-map out ANY

subscriber-policy SP_512K 2
  bandwidth in kbps 512
  bandwidth out kbps 512
  set filter-map in ANY
  set filter-map out ANY

subscriber-policy SP_1M 3
  bandwidth in mbps 1
  bandwidth out mbps 1
  set filter-map in ANY
```



```
set filter-map out ANY

subscriber-policy SP_2M 4
  bandwidth in mbps 2
  bandwidth out mbps 2
  set filter-map in ANY
  set filter-map out ANY

# Настройка сервиса абонентов
subscriber-service SS_PPpOE
  set policy SP_256K

subscriber-service SS_PPpOE2
  set policy SP_512K

subscriber-service SS_PPpOE3
  set policy SP_2M

ppp-options PPP
# Подключить AAA-профиль
set aaa AAA_RADIUS
# Подключить сервис абонентов
set subscriber-service SS_PPpOE
# Включить повторное согласование LCP
lcp renegotiation
# Выбрать возможные способы аутентификации
authentication pap chap
# Подключить пул адресов абонентов
pool ipv4 INET 1
  # Задать шлюз по умолчанию и DNS-сервер
  gateway 200.0.200.1
  dns 8.8.8.8

l2tp-profile L2TP
# Подключить профиль PPP
set ppp-options PPP
# Принимать любые адреса отправителя (source)
set allowed-ip PL_ANY
# Принимать любые адреса назначения (destination)
ipv4 address any
# Наименование маршрутизатора
host-name LNS
# Наименование производителя
```



```
vendor-name RDP

port ge0
  description to_LAC
  mtu 1518
  service-instance dot1q.10
    encapsulation dot1q 10
    rewrite pop 1
  service-instance untag
    encapsulation untagged

port ge1
  description to_RADIUS
  mtu 1514
  service-instance untag
    encapsulation untagged

port ge2
  description to_PC4
  mtu 1514
  service-instance untag
    encapsulation untagged

# Интерфейс loopback маршрутизатора LNS
interface loopback.0
  description RID
  ip mtu 1500
  ip address 3.3.3.3/32

# Адрес по умолчанию маршрутизатора
interface loopback.201
  description DEFAULT_ROUTER
  ip mtu 1500
  ip vrf forwarding VRF_LNS
  ip address 200.0.200.1/32

# Интерфейс BMI для PPP-абонентов
interface bmi.3001
  ip mtu 1500
  # В каком VRF находится интерфейс BMI
  ip vrf forwarding VRF_LNS
  connect port ge0 service-instance untag
  # Получение адреса по DHCP является триггером для создания сессий абонентов
```



```
session-trigger dhcp
reject-timeout 5
ip malicious-action accept
ip address 172.3.3.3/31
# Подключение профиля L2TP
set l2tp-profile L2TP

interface ge1.0
description to_RADIUS
ip mtu 1500
# Привязать интерфейс к VRF
ip vrf forwarding VRF_RADIUS
connect port ge1 service-instance untag
ip address 172.16.2.3/24

interface ge0.10
description to_LAC
ip mtu 1500
# Привязать интерфейс к VRF
ip vrf forwarding VRF_LNS
connect port ge0 service-instance dot1q.10
ip address 172.21.0.1/24

route-map RM_/32 permit 10
match ip address prefix-list PS_/32
```

Настройте LAC

Приведённая ниже конфигурация реализована на базе маршрутизатора Nokia 7750 SR с операционной системой TiMOS-B-20.10.R8.

```
#-----
echo "System Configuration"
#-----
    system
        name "LAC"
    exit

#-----
echo "AAA (Declarations) Configuration"
#-----
    aaa
        radius-server-policy "AAA_radius" create
    exit
```



```
exit
#-----
echo "Card Configuration"
#-----
    card 1
        card-type iom-1
        mda 1
            mda-type me16-25gb-sfp28+2-100gb-qsfp28
            no shutdown
        exit
        no shutdown
    exit
#-----
echo "Connector Configuration"
#-----
    port 1/1/c1
        connector
            breakout c1-10g
        exit
        no shutdown
    exit
    port 1/1/c2
        connector
            breakout c1-10g
        exit
        no shutdown
    exit
    port 1/1/c3
        connector
            breakout c1-10g
        exit
        no shutdown
    exit
    port 1/1/c4
        connector
            breakout c1-10g
        exit
        no shutdown
    exit
#-----
echo "Port Configuration"
#-----
    port 1/1/c1/1
```



```
        description "RADIUS"
        ethernet
            mode access
        exit
        no shutdown
    exit
    port 1/1/c2/1
        description "to_LNS"
        ethernet
            mode access
            encap-type dot1q
        exit
        no shutdown
    exit
    port 1/1/c3/1
        ethernet
            encap-type dot1q
            mtu 1518
        exit
        no shutdown
    exit
    port 1/1/c4/1
        ethernet
            mode access
            encap-type dot1q
        exit
        no shutdown
    exit
    port A/3
        shutdown
        ethernet
        exit
    exit
#-----
echo "System Sync-If-Timing Configuration"
#-----
    system
        sync-if-timing
        begin
        commit
    exit
    exit
#-----
```



```
echo "QoS Policy Configuration"
#-----
    qos
        sap-ingress 20 name "20" create
            description "64K_upstream"
            queue 1 create
                rate 64
            exit
            queue 11 multipoint create
            exit
        exit
        sap-ingress 30 name "30" create
            description "128K_upstream"
            queue 1 create
                rate 128
            exit
            queue 11 multipoint create
            exit
        exit
        sap-ingress 40 name "40" create
            description "256K_upstream"
            queue 1 create
                rate 256
            exit
            queue 11 multipoint create
            exit
        exit
        sap-ingress 50 name "50" create
            description "512K_upstream"
            queue 1 create
                rate 512
            exit
            queue 11 multipoint create
            exit
        exit
        sap-egress 20 name "20" create
            description "256K_downstream"
            queue 1 create
                rate 256
            exit
            fc be create
                queue 1
                dot1p 5
```




```
        dscp ef
    exit
exit
sap-egress 30 name "30" create
    description "512K_downstream"
    queue 1 create
        rate 512
    exit
    fc be create
        queue 1
        dot1p 4
        dscp af21
    exit
exit
sap-egress 40 name "40" create
    description "1M_downstream"
    queue 1 create
        rate 1024
    exit
    fc be create
        queue 1
        dot1p 5
        dscp ef
    exit
exit
sap-egress 50 name "50" create
    description "2M_downstream"
    queue 1 create
        rate 2048
    exit
    fc be create
        queue 1
        dot1p 3
        dscp cs1
    exit
exit
exit
#-----
echo "Management Router Configuration"
#-----

    router management
exit
```



```
#-----
echo "Router (Network Side) Configuration"
#-----

    router Base
        interface "system"
            address 1.1.1.1/32
            description "RID"
            no shutdown
        exit
        autonomous-system 65010
    exit

#-----
echo "Subscriber-mgmt Configuration"
#-----

    subscriber-mgmt
        authentication-policy "AUTH_plcy" create
            pppoe-access-method pap-chap
            include-radius-attribute
                acct-session-id
                remote-id
                nas-port-id
                nas-identifier
                mac-address
            exit
        radius-server-policy "AAA_radius"
    exit
    radius-accounting-policy "ACC_plcy" create
        update-interval 10
        include-radius-attribute
            circuit-id
            framed-ip-addr
            nas-identifier
            nas-port-id
            remote-id
            sla-profile
            sub-profile
            subscriber-id
            user-name
            no detailed-acct-attributes
            std-acct-attributes
        exit
        session-id-format number
```



```
radius-server-policy "AAA_radius"
exit
sla-profile "sla-profile-1M" create
    ingress
        qos 40
        exit
    exit
    egress
        qos 40
        exit
        no qos-marking-from-sap
    exit
exit
sla-profile "sla-profile-256K" create
    ingress
        qos 20
        exit
    exit
    egress
        qos 20
        exit
        no qos-marking-from-sap
    exit
exit
sla-profile "sla-profile-2M" create
    ingress
        qos 50
        exit
    exit
    egress
        qos 50
        exit
        no qos-marking-from-sap
    exit
exit
sla-profile "sla-profile-512K" create
    ingress
        qos 30
        exit
    exit
    egress
        qos 30
        exit
```



```
        no qos-marking-from-sap
    exit
exit
sub-profile "sub-profile-default" create
    radius-accounting
        policy "ACC_plcy"
    exit
    sla-profile-map
        use-direct-map-as-default
    exit
exit
sub-ident-policy "sub-id-default" create
    sub-profile-map
        use-direct-map-as-default
    exit
    sla-profile-map
        use-direct-map-as-default
    exit
exit
msap-policy "msap-default" create
    sub-sla-mgmt
        def-sub-id use-auto-id
        def-sub-profile "sub-profile-default"
        sub-ident-policy "sub-id-default"
        multi-sub-sap limit 10
    exit
exit
local-user-db "PPPoE" create
exit
exit
#-----
echo "Service Configuration"
#-----
service
    system
        bgp-auto-rd-range 1.1.1.1 comm-val 301 to 399
    exit
    customer 1 name "1" create
        description "Default customer"
    exit
    customer 50 name "C_RADIUS" create
        description "C_RADIUS"
    exit
```



```
customer 200 name "C_PPPoE" create
    description "Customers for clients PPPoE"
exit
customer 3001 name "C_L2TP-1" create
    description "L2TP custopers 1"
exit
vprn 50 name "VPRN_RADIUS" customer 50 create
    interface "to_radius" create
    exit
exit
vprn 201 name "VPRN_201_PPPoE" customer 200 create
    interface "vprn_201_loopback" create
    exit
    subscriber-interface "vprn_201_sub_int" create
        group-interface "vprn_201_group_int" create
        exit
    exit
exit
vprn 301 name "VPRN_301_PPPoE" customer 200 create
    subscriber-interface "vprn_301_sub_int" create
        group-interface "vprn_301_group_int" create
        exit
    exit
    interface "vprn_301_loopback" create
    exit
exit
vprn 3001 name "VPRN_L2TP_1" customer 3001 create
    interface "to_LNS-1" create
    exit
    interface "vprn_3001-loopback" create
    exit
    interface "to_LNS-2" create
    exit
exit
vprn 50 name "VPRN_RADIUS" customer 50 create
    description "VPRN for RADIUS"
    autonomous-system 65010
    route-distinguisher 1.1.1.1:50
    interface "to_radius" create
        description "to_RADIUS"
        address 172.16.1.2/24
        sap 1/1/c1/1 create
    exit
```

```
exit
radius-server
                                server "pfsense-1" address 172.16.1.1 secret
"pA10c81qTdyZ9MakJRlmKeY/6w==" hash2 create
                                accept-coa
                                exit
                                exit
                                no shutdown
exit
vpls 200 name "VPLS_PPPOE" customer 200 create
description "VPLS for PPPOE clients"
stp
    shutdown
exit
sap 1/1/c4/1:* capture-sap create
trigger-packet pppoe
msap-defaults
    policy "msap-default"
exit
authentication-policy "AUTH_plcy"
no shutdown
exit
no shutdown
exit
vprn 201 name "VPRN_201_PPPOE" customer 200 create
description "VPRN PPPOE clients"
route-distinguisher auto-rd
interface "vprn_201_loopback" create
    description "VPRN loopback interface"
    address 192.168.3.2/32
    loopback
exit
subscriber-interface "vprn_201_sub_int" create
unnumbered "vprn_201_loopback"
group-interface "vprn_201_group_int" create
    description "VPRN group interface"
    authentication-policy "AUTH_plcy"
    oper-up-while-empty
    pppoe
        session-limit 100
        no shutdown
    exit
exit
```



```
    exit
    no shutdown
exit
vprn 301 name "VPRN_301_PPPoE" customer 200 create
    description "VPRN PPPoE clients"
    route-distinguisher auto-rd
    interface "vprn_301_loopback" create
        description "VPRN loopback interface"
        address 192.168.3.1/32
        loopback
    exit
    subscriber-interface "vprn_301_sub_int" create
        unnumbered "vprn_301_loopback"
        group-interface "vprn_301_group_int" create
            description "VPRN group interface"
            authentication-policy "AUTH_plcy"
            oper-up-while-empty
            pppoe
                session-limit 100
                no shutdown
            exit
        exit
    exit
    no shutdown
exit
vprn 3001 name "VPRN_L2TP_1" customer 3001 create
    description "VPRN L2TP for connections 1"
    autonomous-system 65010
    route-distinguisher 1.1.1.1:3001
    interface "to_LNS-1" create
        address 172.21.0.2/24
        sap 1/1/c2/1:10 create
        exit
    exit
    interface "vprn_3001-loopback" create
        address 172.1.1.1/32
        loopback
    exit
    interface "to_LNS-2" create
        address 172.31.0.2/28
        sap 1/1/c2/1:20 create
        exit
    exit
```




```
bgp
  router-id 172.1.1.1
  group "LNS-1"
    family ipv4
    import "import_/32"
    export "export_/32"
    peer-as 65020
    local-address "to_LNS-1"
    neighbor 172.21.0.1
      type external
      peer-as 65030
    exit
  exit
  no shutdown
exit
l2tp
  no shutdown
exit
no shutdown
exit
exit
#-----
echo "Router (Service Side) Configuration"
#-----
  router Base
#-----
echo "Policy Configuration"
#-----
  policy-options
    begin
    prefix-list "/32s"
      prefix 0.0.0.0/0 prefix-length-range 31-32
    exit
    policy-statement "export_/32"
      entry 10
        from
          protocol direct
          prefix-list "/32s"
        exit
        action accept
      exit
    exit
  default-action drop
```



```
        exit
    exit
    policy-statement "import_/32"
        entry 10
            from
                protocol bgp
                prefix-list "/32s"
            exit
            action accept
            exit
        exit
    default-action drop
    exit
exit
commit
exit
exit

#-----
echo "Subscriber-mgmt (Service Side) Configuration"
#-----
    subscriber-mgmt
        local-user-db "PPPoE" create
            description "DB for PPPoE clients"
            ppp
                match-list username
                host "default" create
                    auth-policy "AUTH_plcy"
                    no shutdown
                exit
            exit
        no shutdown
    exit
exit

#-----
echo "Log all events for service vprn Configuration"
#-----
    log
    exit

#-----
echo "AAA Configuration"
#-----
    aaa
```



```
radius-server-policy "AAA_radius" create
  acct-on-off oper-state-change
  servers
    router 50
    buffering
      acct-interim min 60 max 3600 lifetime 5
      acct-stop min 60 max 3600 lifetime 5
    exit
  server 1 name "pfsense-1"
  exit
exit
exit
exit
```

Настройте PPP1, PPP2 и PPP3

Все абонентские PPP устройства настраиваются аналогично друг другу:

/etc/ppp/peers/provider

```
defaultroute
plugin rp-pppoe.so
eth0
user admin@ecorouter.lab
usepeerdns
```

/etc/ppp/chap-secrets

```
# Secrets for authentication using CHAP
# client      server  secret                      IP addresses
"admin@ecorouter.lab" * "pass1234"
```

- PPP1 должен получить имя admin@ecorouter.lab и пароль pass1234.
- PPP2 должен получить имя admin2@ecorouter.lab и пароль pass12345.
- PPP3 должен получить имя admin3@ecorouter.lab и пароль pass123456.

Настройте RAD1:

На интерфейсе eth0 настройте IP-адрес 172.16.1.1 и маску 255.255.255.0.

/etc/raddb/clients.conf

```
client LAC {
  ipaddr = 172.16.1.2
  secret = lac
}
```

Словарь /etc/raddb/dictionary должен быть заполнен атрибутами соответствующими производителю LAC.



В файле /etc/raddb/users задайте настройки для трёх абонентов:

```
admin@ecorouter.lab Cleartext-Password := "pass1234", NAS-Identifier==LAC, Auth-Type=Accept
```

```
Alc-Tunnel-Serv-Id=3001,  
Tunnel-Assignment-Id="ISP-retail-2",  
Tunnel-Type=L2TP,  
Tunnel-Medium-Type=IP,  
Tunnel-Server-Endpoint=172.3.3.3,  
Tunnel-Client-Endpoint=172.1.1.1,  
Tunnel-Assignment-Id+="ISP2",  
Alc-MSAP-Policy="msap-default",  
Alc-MSAP-Interface="vprn_201_group_int",  
Alc-MSAP-Serv-Id=201,  
Alc-SLA-Prof-Str="sla-profile-2M"
```

```
admin2@ecorouter.lab Cleartext-Password := "pass12345", NAS-Identifier==LAC,  
Auth-Type=Accept
```

```
Alc-Tunnel-Serv-Id=3001,  
Tunnel-Assignment-Id="ISP-retail-2",  
Tunnel-Type=L2TP,  
Tunnel-Medium-Type=IP,  
Tunnel-Server-Endpoint=172.3.3.3,  
Tunnel-Client-Endpoint=172.1.1.1,  
Tunnel-Assignment-Id+="ISP2",  
Alc-MSAP-Policy="msap-default",  
Alc-MSAP-Interface="vprn_201_group_int",  
Alc-MSAP-Serv-Id=201,  
Alc-SLA-Prof-Str="sla-profile-2M"
```

```
admin3@ecorouter.lab Cleartext-Password := "pass123456", NAS-Identifier==LAC,  
Auth-Type=Accept
```

```
Alc-Tunnel-Serv-Id=3001,  
Tunnel-Assignment-Id="ISP-retail-2",  
Tunnel-Type=L2TP,  
Tunnel-Medium-Type=IP,  
Tunnel-Server-Endpoint=172.3.3.3,  
Tunnel-Client-Endpoint=172.1.1.1,  
Tunnel-Assignment-Id+="ISP2",  
Alc-MSAP-Policy="msap-default",  
Alc-MSAP-Interface="vprn_201_group_int",
```



```
Alc-MSAP-Serv-Id=201,
Alc-SLA-Prof-Str="sla-profile-2M"
```

Настройте RAD2

На интерфейсе eth0 настройте IP-адрес 172.16.2.1 и маску 255.255.255.0.

/etc/raddb/clients.conf

```
client LNS {
  ipaddr = 172.16.2.3
  secret = lns
}
```

Словарь /etc/raddb/dictionary должен быть заполнен специфичными для производителя РДП Инновации атрибутами:

```
VENDOR      RDP      45555

BEGIN-VENDOR RDP
ATTRIBUTE    SUBSCRIBER_POLICY_OPTIONS      242 string
ATTRIBUTE    SUBSCRIBER_ID              243 string
ATTRIBUTE    SUBSCRIBER_HW_ADDRESS      244 string
ATTRIBUTE    SUBSCRIBER_POLICY_BANDWIDTH 245 string
ATTRIBUTE    TIME_QUOTA                 246 integer
ATTRIBUTE    FILTER_MAP_POLICY          247 string
ATTRIBUTE    POLICY_NAME                248 string
ATTRIBUTE    SUBSCRIBER_OPTION          249 integer
ATTRIBUTE    SUBSCRIBER_SERVICE_NAME    250 string
ATTRIBUTE    SHARED_SERVICES            251 string
ATTRIBUTE    BANDWIDTH_IN               252 integer
ATTRIBUTE    BANDWIDTH_OUT              253 integer
ATTRIBUTE    REDIRECT_URL                254 string
ATTRIBUTE    VRF_NAME                   255 string
ATTRIBUTE    ACCT_IN_POLICY_OCTETS_64    150 octets
ATTRIBUTE    ACCT_OUT_POLICY_OCTETS_64   151 octets
ATTRIBUTE    ACCT_IN_POLICY_PACKETS_64   152 octets
ATTRIBUTE    ACCT_OUT_POLICY_PACKETS_64  153 octets
ATTRIBUTE    PRIMARY_DNS                 93  string
ATTRIBUTE    SECONDARY_DNS               94  string
ATTRIBUTE    VENDOR_CLASS                95  string
ATTRIBUTE    REMOTE_ID                   96  string
ATTRIBUTE    CIRCUIT_ID                  97  string
END-VENDOR  RDP
```

В файле /etc/raddb/users задайте настройки для трёх абонентов:

```
admin@ecorouter.lab  Cleartext-Password := "pass1234", NAS-Identifier==LNS
                      SUBSCRIBER_SERVICE_NAME+="SS_PPPoE",
                      Session-Timeout=1200

admin2@ecorouter.lab  Cleartext-Password := "pass12345", NAS-Identifier==LNS
                      SUBSCRIBER_SERVICE_NAME+="SS_PPPoE2",
                      Session-Timeout=1200

admin3@ecorouter.lab  Cleartext-Password := "pass123456", NAS-Identifier==LNS
                      SUBSCRIBER_SERVICE_NAME+="SS_PPPoE3",
                      Session-Timeout=1200
```

Проверьте работу построенной схемы

На LNS проверьте BGP-соседство с LAC командой `show ip bgp summary vrf VRF_LNS`.

Для проверки соединения, с абонентских устройств (PPP1, PPP2, PPP3) дайте команду `ping 173.0.2.2` (PC4).

На LNS проверьте список присоединённых абонентов командой `show subscribers bmi.3001`. В выводе таблицы должны присутствовать абоненты admin@ecorouter.lab, admin2@ecorouter.lab, admin3@ecorouter.lab.

На LNS командой `show subscribers bmi.3001 <A.B.C.D>` проверьте, какой сервис (Subscriber-service) применён для каждого абонента. Должны быть применены сервисы SS_PPPoE, SS_PPPoE2 и SS_PPPoE3 соответственно.



31 Экспорт данных о трафике

В EcoRouter реализована поддержка IPFIX, согласно RFC5101 (NetFlow v.10), с использованием UDP и порта 4739 для передачи данных коллектору.

Netflow-сенсор выделяет из проходящего трафика потоки, характеризующиеся 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 119 — Команды контекстного режима `config-flow-export`

Команда	Описание
<code>description <DESCRIPTION></code>	Создание описания профиля
<code>destination <IP></code> <code>[port <1-65535>]</code> <code>[vrf <NAME>]</code> <code>[source <IP>]</code>	IP-адрес коллектора. Адрес задаётся в формате A.B.C.D. После указания адреса можно указать UDP-порт коллектора. Также можно указать виртуальную таблицу маршрутизации (VRF), через которую будет производиться передача данных (параметр



Команда	Описание
	недоступен для виртуальных маршрутизаторов). С помощью параметра source можно указать определённый IP адрес, который будет использован как адрес источника в пакетах, отправляемых на коллектор.
<code>packet-sampling <1-1000></code>	Частота выборки пакета из потока. Пакет будет направлен на коллектор. Значение по умолчанию = 500 (будет отобран каждый пятисотый пакет).
<code>timeout active <1-300></code>	Временной интервал, по истечении которого данные будут переданы на коллектор при активной сессии, в секундах. Значение по умолчанию = 60
<code>timeout inactive<5-300></code>	Временной интервал, по истечении которого данные будут переданы на коллектор после закрытия сессии, в секундах. Значение по умолчанию = 15
<code>timeout template <1-30></code>	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.

31.1 Пример настройки

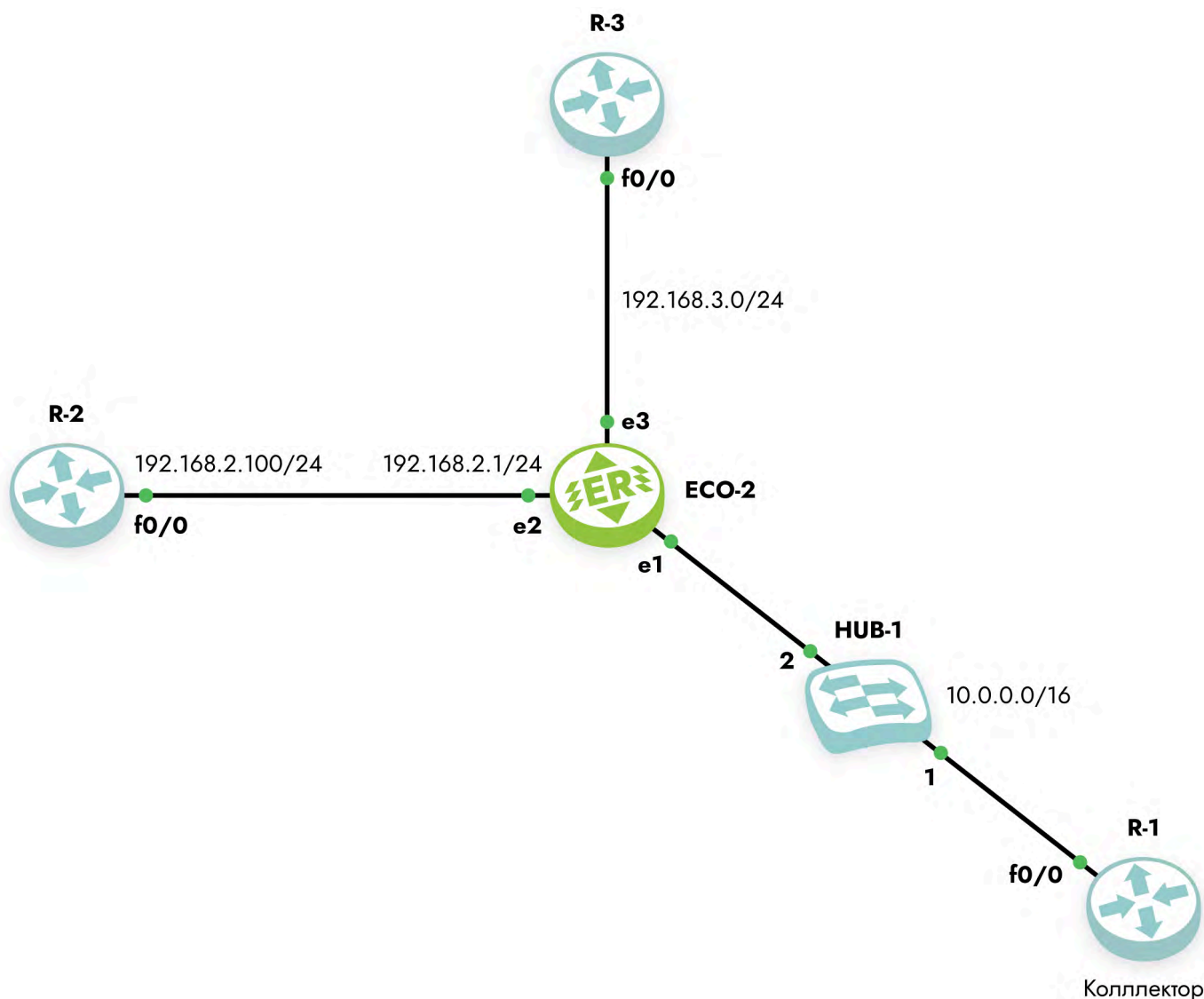


Рисунок 66

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 1
ecorouter(config-flow-export)#description Netflow
ecorouter(config-flow-export)#destination 172.16.0.2
ecorouter(config-flow-export)#packet-sampling 100
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 1
```

31.2 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 1
  Description: Netflow.10
  Destination: 172.16.0.2
  Active timeout: 30
```



Inactive timeout: 30

Packet sampling: 100

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.

32 QoS

QoS (англ. quality of service, качество обслуживания) — это комплекс технологий и механизмов в сетях передачи данных, предназначенный для гарантированного предоставления различных классов обслуживания трафику в зависимости от его критичности, типа приложения и требований к сетевым ресурсам.

Основная цель QoS — обеспечить предсказуемое поведение сети путём управления полосой пропускания, задержкой, джиттером и потерей пакетов, особенно в условиях перегрузки или ограниченных ресурсов.

32.1 Архитектура QoS

В EcoRouter схема реализации QoS разделена логически на несколько взаимодействующих блоков:

- Классификатор/Classifier.
- RED (Random Early Detection) — случайное раннее обнаружение.
- Планировщик/Scheduler.

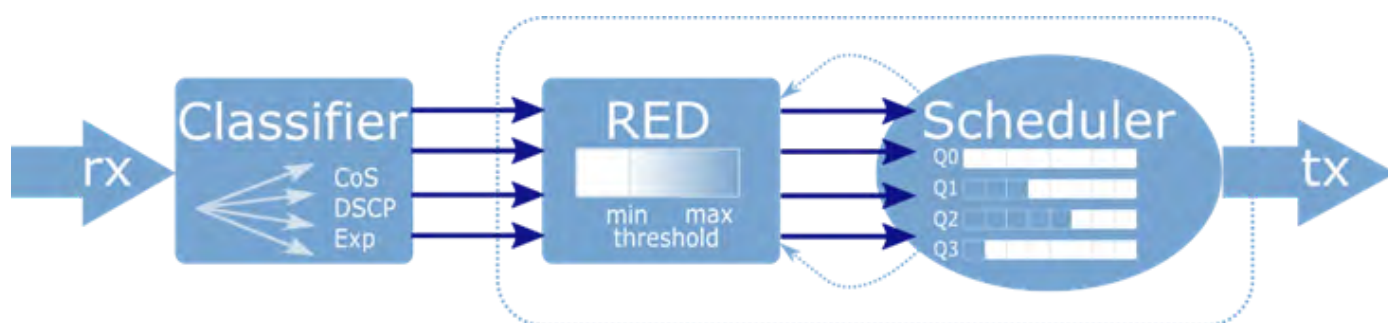


Рисунок 67

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.



32.2 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, но 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 её 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 определённые пакеты из общего потока трафика путём указания значения поля или его наименования в заголовках Ethernet, MPLS или IP. По значениям этих полей будет осуществляться классификация трафика. Ввод 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 ?
RANGE Enter class-of-service values: 1-3,6 or 1-3,5-7
ecorouter(config-cmap)#match dscp ?
RANGE Enter DSCP values: 1-3,32 or 1-3,33-44
af11 Assured Forwarding class for value 10
af12 Assured Forwarding class for value 12
af13 Assured Forwarding class for value 14
af21 Assured Forwarding class for value 18
af22 Assured Forwarding class for value 20
af23 Assured Forwarding class for value 22
af31 Assured Forwarding class for value 26
af32 Assured Forwarding class for value 28
af33 Assured Forwarding class for value 30
af41 Assured Forwarding class for value 34
af42 Assured Forwarding class for value 36
af43 Assured Forwarding class for value 38
cs0 Class Selector Routine for value 0
cs1 Class Selector Priority for value 8
cs2 Class Selector Immediate for value 16
cs3 Class Selector Flash for value 24
cs4 Class Selector Flash override for value 32
cs5 Class Selector Critical for value 40
cs6 Class Selector Internetwork Control for value 48
cs7 Class Selector Network Control for value 56
ef Expedited Forwarding DSCP for value 46
ecorouter(config-cmap)#match exp ?
RANGE Enter MPLS exp values: 1-3,6 or 1-3,5-7
```

Как видно из примера, классификация в EcoRouterOS может осуществляться по полям **cos**, **dscp** и **exp**. Значения могут задаваться в десятичном виде, либо, как в случае с **dscp**, прямым указанием класса согласно стандарту RFC 4594. Можно задавать набор значений, используя в качестве разделителя запятую «,» и диапазоны, используя в качестве разделителя дефис «-».

Также пользователю доступна команда **set**. Она необходима для того, чтобы присвоить отобранным командой **match** пакетам новый класс обслуживания. Команда позволяет присвоить исходящим пакетам такие же как и при использовании команды **match** значения классов трафика: Ethernet — **cos**; MPLS — **exp** и IP — **dscp**. Подробнее об этом читайте в подразделе «Перемаркировка трафика».

```
R3(config-cmap)#set ?
cos IEEE 802.1Q class of service priority values
```




```
dscp  Set DSCP in IP packets
exp   Set MPLS experimental
```

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.

После имени карты классов можно указать ограничения объема трафика в различных единицах измерения.

Пример:

```
ecorouter(config-traffic-profile)#class VOICE ?
gbps    Max bandwidth value in Gbps
kbps    Max bandwidth value in Kbps
mbps    Max bandwidth value in Mbps
percent  Max bandwidth value as a percentage
<cr>
ecorouter(config-profile)#class VIDEO
ecorouter(config-profile)#class VOICE mbps 10
```

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
```

32.3 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* (скопление). 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 ре-инициализации TCP окна, и скорость всех потоков одновременно падает, после чего, одновременно растёт. В итоге, график загрузки интерфейса выглядит пилообразно, 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 раньше, чем они поступят в очередь. Это позволяет добиться того, что TCP-сессии будут изменять размер окна попеременно. Вероятность отбрасывания пакетов в этом случае является адаптивным (зависящим от других показателей) значением.

Пользователь устанавливает нижнее пороговое значение загруженности интерфейса, начиная с которого растёт вероятность отбрасывания. Пользователь устанавливает также и верхнее значение загруженности при котором вероятность отбрасывания пакетов становится максимальной. Максимальная вероятность отбрасывания пакетов также устанавливается пользователем.

При изменении загруженности интерфейса в рамках описанных двух пороговых значений вероятность отбрасывания растёт от 0 до указанного максимального значения, согласно принятой математической функции, учитывающей среднюю загруженность полосы пропускания, количество пакетов, пропущенных без отбрасывания.



32.3.1 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, Peak Information Rate). 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.

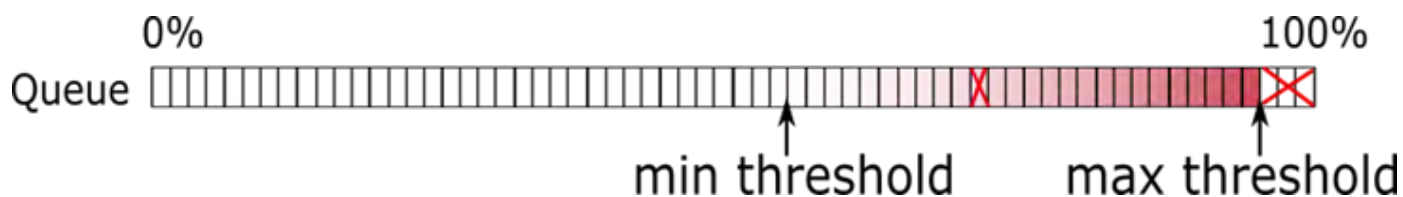


Рисунок 68

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 от 1 до 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.

32.3.2 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 отключён.

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 от **1** до **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.

32.4 Планировщик (Scheduler)

Планировщик управляет механизмом очередей. Под очередью (queue) в концепции EcoRouter понимается программно реализуемая очередь пакетов. Пакеты в такой очереди удерживаются 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 её номером, определяет порядок, в котором они обрабатываются (см. рисунок ниже). То есть, после передачи гарантированного объёма трафика (CIR) первой будет обрабатываться очередь 0 с наивысшим приоритетом. Далее будет обрабатываться очередь 1, 2 и так далее.



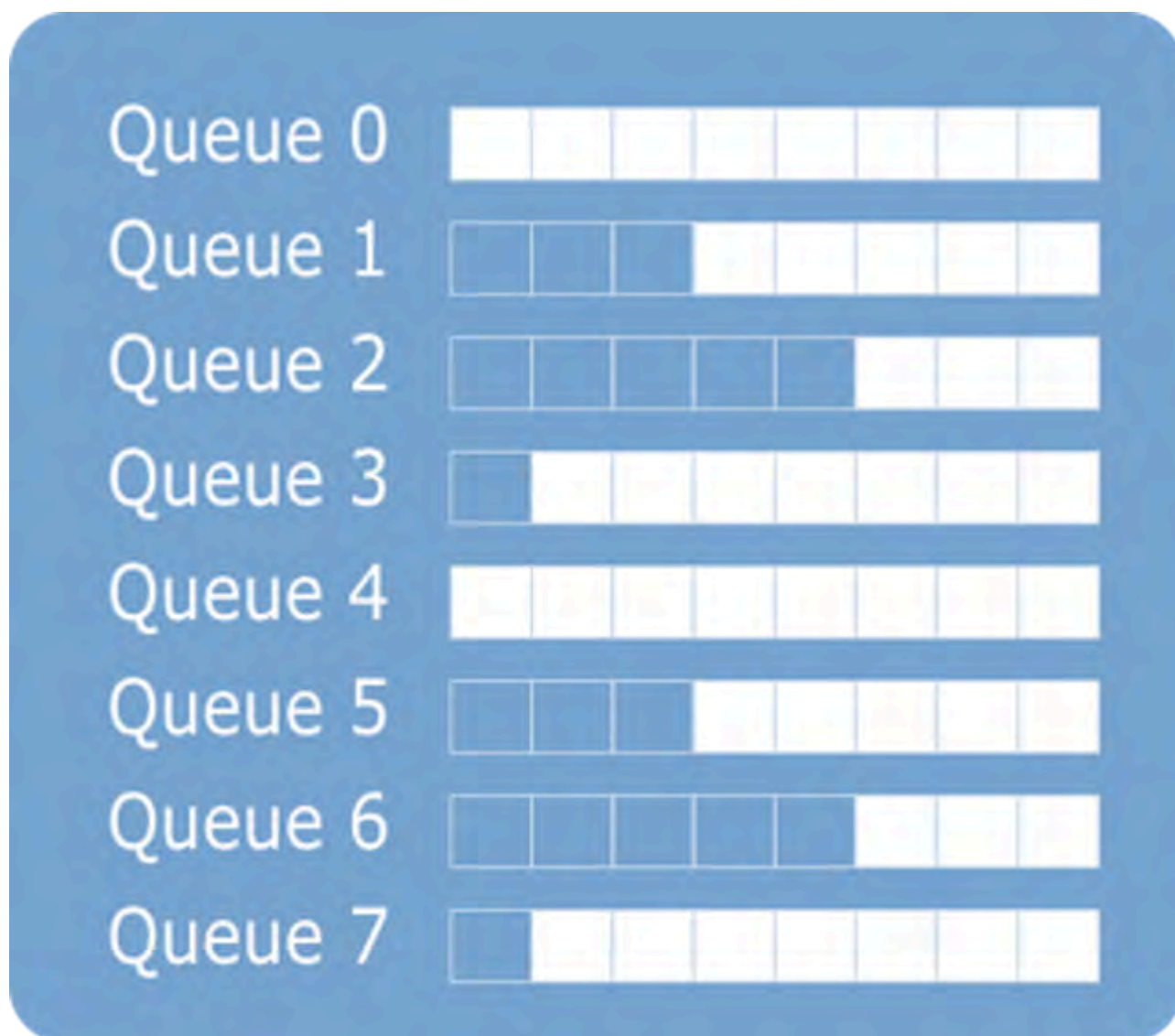


Рисунок 69

Размер каждой очереди динамически изменяется. Это необходимо для поддержания приемлемых значений полосы пропускания, задержки и дрожания фазы для не приоритетных очередей. Это придаёт гибкость при различных вариантах построения сети и типах передаваемого трафика. Сетевому администратору не придётся 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 объём 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 остальных классах. Данный класс нельзя настроить, но можно назначить на любую очередь.

На схеме ниже представлен алгоритм обслуживания очередей планировщиком.

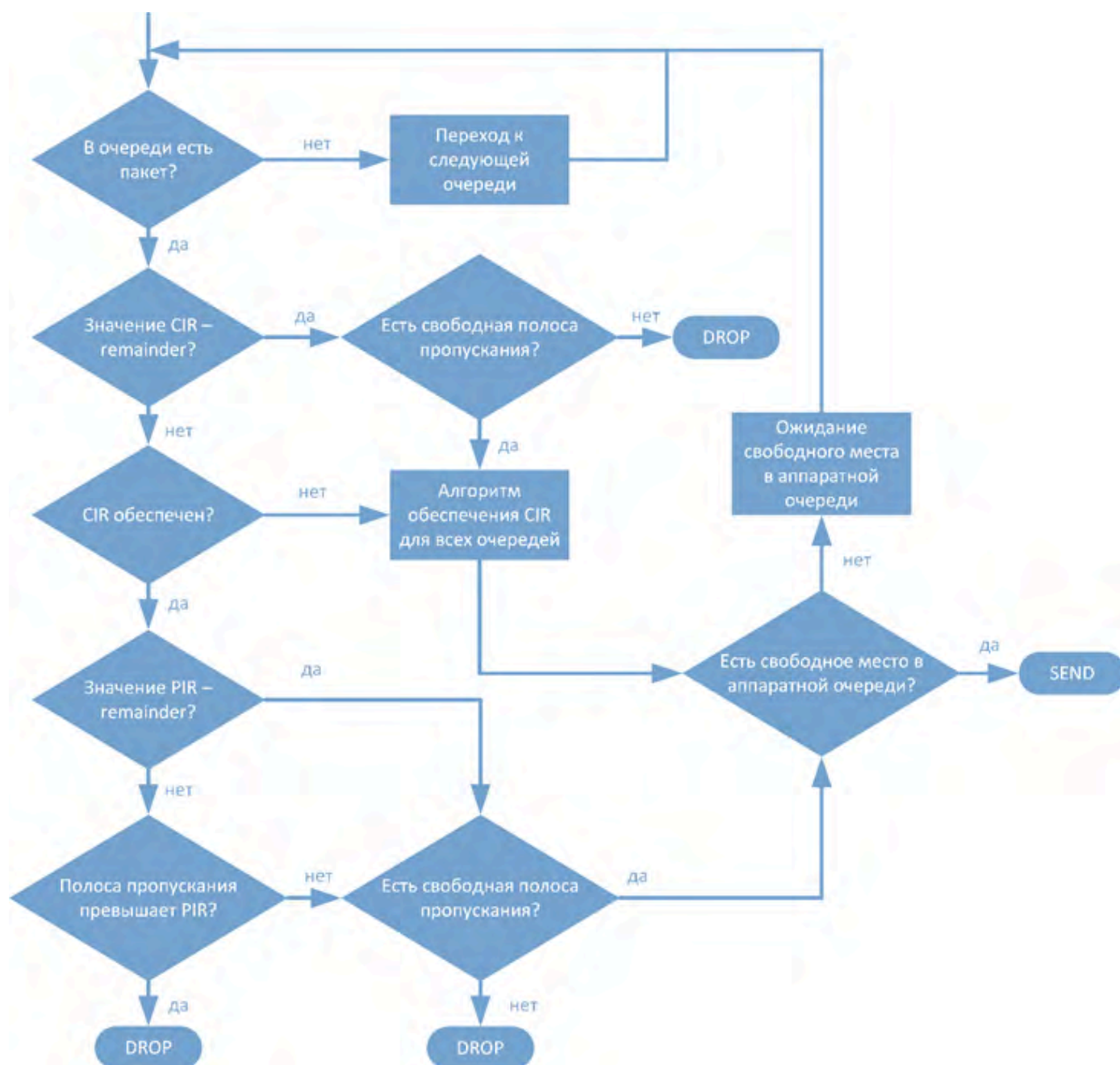


Рисунок 70

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 передаётся 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.

32.4.1 Настройка планировщика и очередей

Для создания планировщика, в конфигурационном режиме используется команда: `traffic-scheduler <NAME>`. Где **NAME** может иметь формат `pqwr.<NUM>` с порядковым номером после точки.

Название планировщика обязательно должно начинаться с префикса **"pqwr."**.

Далее в созданном планировщике задаются очереди.

Синтаксис команды: `queue <0-7> class <NAME> cir <CIR> pir <PIR> [wred-min <0-100> wred-max <0-100>] [wred-inv-prob <1-255>]`.

Параметры команды `queue`:

- **0-7** — номер очереди.
- **NAME** — имя созданного класса трафика или **"default"** (это служебный класс, в который попадает любой трафик, не указанный остальных классах).
- **CIR** — Объем передаваемого за дельту времени трафика, который будет передан гарантированно. Суммарное значение CIR в очередях одного планировщика не может превышать 100%. Задаётся одним из следующих способов:
 - в процентах (от 0 до 100);
 - в абсолютных величинах (в Kbps). Для задания значения в абсолютных величинах, после значения параметра должно стоять обозначение **kbps**, например: **500000 kbps**;
 - оставшаяся нераспределённой полоса — **remainder**.
- **PIR** — Трафик, превышающий PIR (Peak Information Rate), будет безусловно отброшен. Задаётся одним из следующих способов:
 - в процентах (от 0 до 100);

- в абсолютных величинах (в Kbps). Для задания значения в абсолютных величинах, после значения параметра должно стоять обозначение **kbps**, например: **500000 kbps**;
- оставшаяся нераспределённой 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. Задаётся значение знаменателя дроби **Probability = 1 / X**. Значения устанавливаются в диапазоне (0 - 255). Значение по умолчанию — 10.

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 очередь по умолчанию (default) — с наименьшим приоритетом. То есть обслуживается только в случае, если остальные очереди полностью использовали 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 3 class VIDEO cir 100000 kbps pir
```

```
500000 kbps wred-min 5 wred-max 20 wred-inv-prob 200
ecorouter(config-traffic-scheduler)# exit
```

32.5 Счётчики

Для просмотра счётчиков QoS используется команда административного режима `show counters port <NAME> queues`.

Внимание: в EcoRouterOS в командах группы `show` при подсчёте количества данных не учитываются следующие поля Ethernet-фрейма: Preamble, Frame delimiter, FCS, Interpacket gap (24 байта).

Показания счётчиков 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 — вывести значения счётчиков QoS для порта te1: `ecorouter#show counters port te1 queues`

```
Port te0
Service instance te0/eth1
Traffic scheduler pqwrr.0
Early detection algorithm: RED
QoS Statistics:
```

queue class	Match packets/bytes	RED-drop packets/bytes	WRED-drop packets/bytes	Tail-drop packets/bytes	Total-drop packets/bytes
0 IP0	27922/42262228	0/0	0/0	3776/5716144	3776/5716144
1 IP1	5170/7817860	0/0	0/0	1241/1878874	1241/1878874
2 IP2	0/0	0/0	0/0	0/0	0/0
3 IP3	0/0	0/0	0/0	0/0	0/0
4 ---	0/0	0/0	0/0	0/0	0/0
5 ---	0/0	0/0	0/0	0/0	0/0
6 ---	0/0	0/0	0/0	0/0	0/0
7 default	47/4102	0/0	0/0	0/0	0/0

Рисунок 71

Для просмотра счётчиков QoS при использовании алгоритма WRED используется команда административного режима `show counters port <NAME> wred`.

Отображаются: класс трафика, сконфигурированные параметры, глубина очереди в % от PIR и количество отброшенных пакетов/байтов при использовании алгоритма WRED.

Пример — вывести значения счётчиков QoS с учётом WRED для порта te0: `ecorouter#show counters port te0 wred`.

```
Port te0
Service instance te0/eth1
traffic scheduler pqwrr.0
```

queue class	thresholds		mark	current	WRED-drop packets/bytes
	min	max	probability	load	
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#
```

Рисунок 72

Для просмотра счётчиков QoS по количеству ограниченного трафика используется команда административного режима `show counters port <NAME> policer {in | out}`.

Показания счётчиков группируются по портам, выводятся данные по пройденным и отброшенным пакетам/байтам.

Пример — вывести значения счётчиков ограниченного трафика для порта te1, входящий трафик:

```
ecorouter#show counters port te1 policer in
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
```

Для сброса счётчиков можно воспользоваться командами `clear`.

```
ecorouter#clear counters port te1 ?
policer    policer statistics
queues     QoS queues statistics
red-algorithms QoS RED/WRED algorithms statistics
```

32.6 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 задаётся 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, её 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 приёме данных со скоростью, превышающей установленный лимит, отображает следующий график.

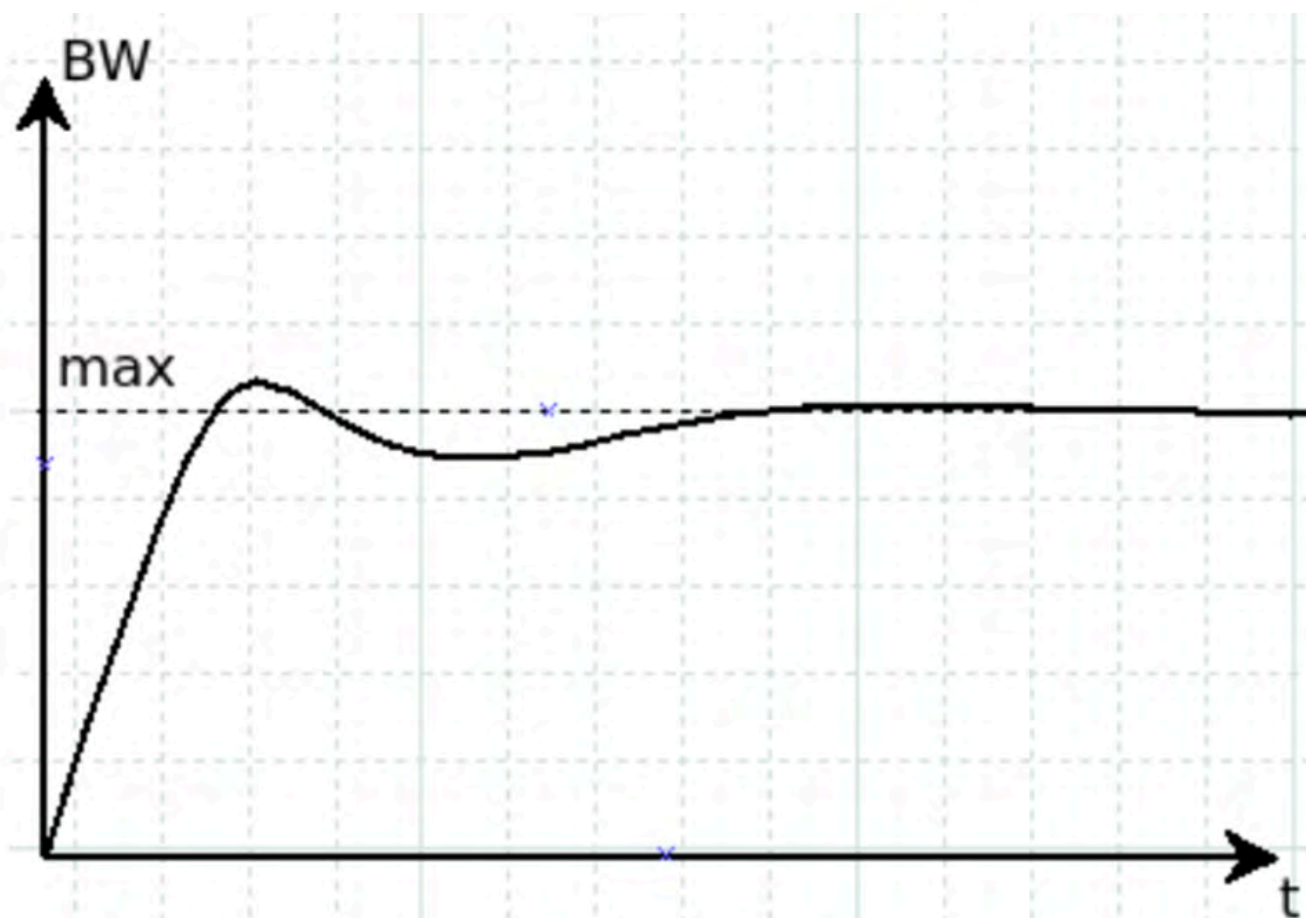


Рисунок 73

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 объёма данных и усреднения необходимо довольно продолжительное время (при подаче трафика на одной и той же скорости ≈ 300 сек). Для определения реального объёма пропускаемого трафика удобнее воспользоваться командой `show counters port queues-speed`.

Описание параметров `scheduler` и `traffic-profile` дано в подразделе «Сервисные политики».

32.7 Маркировка трафика

Маркировка трафика настраивается в EcoRouterOS при помощи сущности `filter-map` (см. раздел "Списки доступа"). Таким образом, к трафику определённого вида применяются различные действия, в том числе, маркировка. Под маркировкой здесь



понимается то, что трафику, попадающему под действие правила, присваивается определённый класс (class-map).

Ниже приведён 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
```

Создаём ещё 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
ecorouter(filter-map-ethernet)#set class-map L2
```

Создаём ещё один блок фильтрации для 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 попра.

```
ecorouter(config)#port te1
ecorouter(config-port)#srevic-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 te1
ecorouter(config-port)#service-instance 100
ecorouter(config-service-instance)#qos reset dscp 63
```

32.8 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**) путём 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 кадров 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 её к экземпляру сервиса (service-instance) в исходящем направлении. Более подробную информацию об этих шагах можно прочесть в разделах посвящённых классификации трафика и созданию сервисных политик. Ниже приведён только пример конфигурирования функционала перемаркировки исходящего трафика в EcoRouterOS. Перемаркировка во 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-traffic-profile)#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
```

32.9 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 её 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`.

32.10 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 создаётся при помощи команды `traffic-profile <NAME>`, где **NAME** может быть любым, рекомендуемый формат имени — заглавные буквы или цифры.

После создании 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 ещё 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 идёт 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.

32.11 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 определённого 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 доступен ввод в класс нескольких 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.

32.12 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 данные статистики можно сбросить при помощи команды `clear counters port <NAME> policer in`.

32.13 Пример классификации трафика

Данный подраздел описывает процедуру настройки и проверки, целью которой является подтверждение того, что трафик с различными метками DSCP корректно классифицируется и распределяется по соответствующим очередям на маршрутизаторе ECO-1.

Постройте схему в соответствии с рисунком ниже.

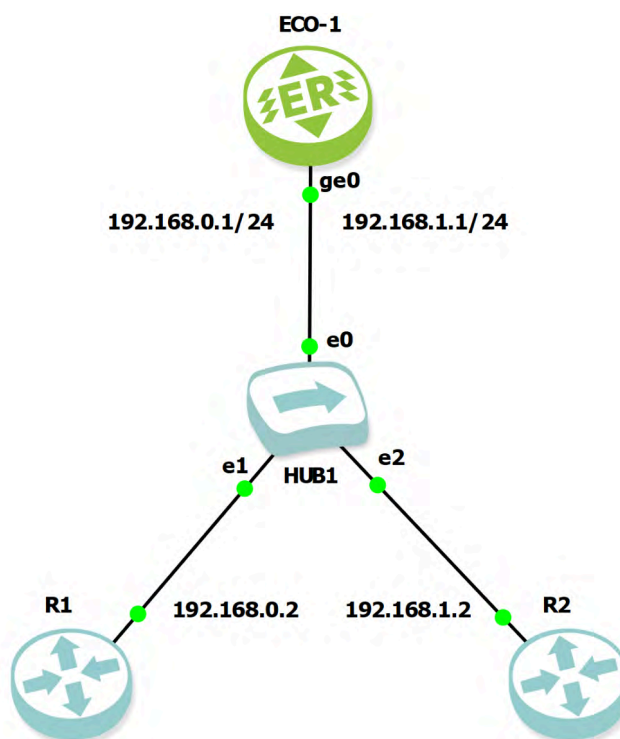


Рисунок 74

Настройте ECO-1:

```
enable
configure
```

```
hostname ECO-1

# Настройка порта и интерфейсов
port ge0
  service-instance unit0
    encapsulation untagged
  service-instance unit1
    encapsulation dot1q 10
    rewrite pop 1
interface 2ge0
  connect port ge0 service-instance unit0
  ip address 192.168.0.1/24
interface 2ge1
  connect port ge0 service-instance unit1
  ip address 192.168.1.1/24

# Создание карт классов
class-map CMAP0
  match dscp 32
class-map CMAP1
  match dscp 40
class-map CMAP2
  match dscp 48
class-map CMAP3
  match dscp 56

# Создание профиля трафика
traffic-profile prof
  class CMAP0
  class CMAP1
  class CMAP2
  class CMAP3

# Создание планировщика очередей трафика
traffic-scheduler pqwrr.0
  weighted-random-detect
  queue 0
    class CMAP0 cir percent 25 pir percent 25
    burst-window 250
  queue 1
    class CMAP1 cir percent 25 pir percent 25
    burst-window 250
  queue 2
```



```
class CMAP2 cir percent 25 pir percent 25
burst-window 250
queue 3
class CMAP3 cir percent 25 pir percent 25
burst-window 250

# Создание политики обслуживания
service-policy policer.0
bandwidth percent 100
traffic-profile prof
scheduler pqwrr.0

# Применение политики к экземплярам сервиса
port ge0
service-instance unit0
service-policy policer.0 in
service-instance unit1
service-policy policer.0 out
```

Настройте R1:

```
configure terminal
hostname R1

# Настройка интерфейса
interface FastEthernet0/0
ip address 192.168.0.2 255.255.255.0
duplex half
no shutdown

ip route 192.168.1.0 255.255.255.0 192.168.0.1
```

Настройте R2:

```
configure terminal
hostname R2

# Настройка интерфейса
interface FastEthernet0/0
no ip address
duplex half
no shutdown

interface FastEthernet0/0.10
```



```
encapsulation dot1Q 10
ip address 192.168.1.2 255.255.255.0

ip route 192.168.0.0 255.255.255.0 192.168.1.1
```

Теперь необходимо проверить работу классификации и распределение трафика по очередям. Это можно сделать с помощью команды `ping` с параметром `tos`.

Значения DSCP соответствуют некоторым значениям TOS:

- Значение TOS 128 соответствует значению DSCP 32 и классу CS4 (Class Selector 4). Realtime Interactive. Интерактивные реального времени приложения (видеоконференции, интерактивное gaming).
- Значение TOS 160 соответствует значению DSCP 40 и классу CS5 (Class Selector 5). Broadcast Video & IP Video Conferencing. Видеотрафик (например, IPTV, видеотрансляции).
- Значение TOS 192 соответствует значению DSCP 48 и классу CS6 (Class Selector 6). Internetwork/Network Control. Критически важный служебный трафик (BGP, OSPF, EIGRP, HSRP). Высший приоритет.
- Значение TOS 224 соответствует значению DSCP 56 и классу CS7 (Class Selector 7). Reserved/Network Control. Зарезервировано для внутреннего использования сетевым оборудованием. Не используется для пользовательского трафика.

Таким образом, мы имеем возможность промаркировать трафик, создаваемый командой `ping`, в соответствии со значениями DSCP, заданными в картах классов.

На R1 выполните следующие команды:

```
ping 192.168.1.2 tos 128 repeat 3
ping 192.168.1.2 tos 160 repeat 3
```

На R2 выполните следующие команды:

```
ping 192.168.0.2 tos 192 repeat 3
ping 192.168.0.2 tos 224 repeat 3
```

На маршрутизаторе ECO-1 выполните команду `show counters port ge0 queues brief` и сверьте полученные результаты с выводом ниже (вывод дан в урезанном виде). При необходимости, чтобы очистить счётчики очередей, воспользуйтесь командой `clear counters port ge0 queues`.

```
ECO-1#sh counters port ge0 queues brief
Port ge0
Service instance unit0
No traffic scheduler
```



Service instance unit1

Traffic scheduler pqwrr.0

Early detection algorithm: WRED

QoS Statistics:

RED-drop

packets/bytes

0/0

Match

queue class

packets/bytes

0 0

3/354

1 1

3/354

2 2

3/354

3 3

3/354

4 ---

0/0

5 ---

0/0

6 ---

0/0

7 default

18/1188

32.14 Пример управления трафиком

Данный пример конфигурации демонстрирует практическое применение таких механизмов QoS как:

- Ограничение общей полосы пропускания.
- Приоритезация трафика на основе DSCP.
- Распределение ширины полосы пропускания между очередями.

Постройте схему в соответствии с рисунком ниже.

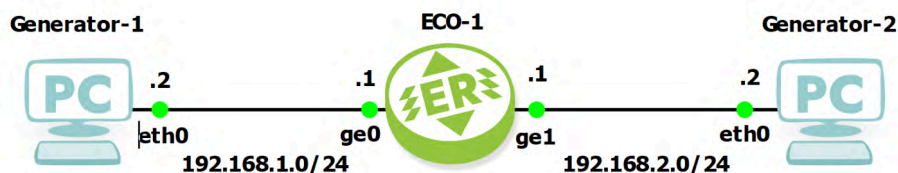


Рисунок 75

Схема состоит из маршрутизатора EcoRouter посередине и двух оконечных устройств выступающих в роли генераторов трафика.

Настройте ECO-1:

```

enable
configure
hostname ECO-1

```

```
# Настройка портов и интерфейсов
interface 2ge0
ip address 192.168.1.1/24
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0

interface 2ge1
ip address 192.168.2.1/24
port ge1
service-instance 4ge1
encapsulation untagged
connect ip interface 2ge1

# Создание профиля трафика
traffic-profile profile-dscp

# Создание политик обслуживания
service-policy policy
  bandwidth percent 100
  traffic-profile profile-dscp

service-policy policer.1
  bandwidth kbps 3000
  traffic-profile profile-dscp

# Применение политики обслуживания к экземпляру сервиса
port ge1
  service-instance 4ge1
    service-policy policer.1 out
```

Настройте интерфейс и маршрутизацию на Generator-1:

```
ip a a 192.168.1.2/24 dev eth0
ip route add default via 192.168.1.1
```

Настройте интерфейс и маршрутизацию на Generator-2:

```
ip a a 192.168.2.2/24 dev eth0
ip route add default via 192.168.2.1
```

На Generator-1 выполните команду `ping` в адрес Generator-2:




```
ping 192.168.2.2 -c 5
PING 192.168.2.2 (192.168.2.2) 56(84) bytes of data.
64 bytes from 192.168.2.2: icmp_seq=1 ttl=63 time=20.6 ms
. . .

--- 192.168.2.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4007ms
rtt min/avg/max/mdev = 19.919/20.428/20.873/0.393 ms
```

Вы должны получить успешный результат (все пакеты переданы и получены) .

На Generator-2 в режиме сервера на порте 5202 запустите утилиту **iperf3**: `iperf3 -s -D -p 5202` .

На Generator-1 для измерения пропускной способности запустите утилиту `iperf3`.

```
iperf3 -c 192.168.2.2 -p 5202 -t 30 -S 0 -b 10M -l 8 --logfile out_iperf
cat out_iperf | grep Mbits | tail -n 1
[ 6]  0.00-30.02 sec  11.3 MBytes  3.14 Mbits/sec  receiver
```

Утилита `iperf3` с указанными параметрами отправляет данные на порт 5202 Generator-2 со скоростью 10 Мбит/с и сохраняет значения измерений за период в 30 секунд в файл **out_iperf**. Команда `cat` выводит на экран предпоследнюю строку файла **out_iperf**, которая содержит среднее значение скорости приёма данных.

Полученная средняя скорость в "3.14 Mbits/sec" приблизительно соответствует заданному в политике **policer.1** значению в 3000 Кбит/сек. Политика ограничения пропускной способности функционирует верно.

Для дополнительной проверки вы можете изменить значение пропускной способности в политике обслуживания и повторить измерения.

На ECO-1 настройте дополнительные политики обслуживания, карты классов трафика, профили трафика и планировщик очередей:

```
# Установка нового значения пропускной способности политики обслуживания
service-policy policer.1
bandwidth kbps 3550

# Создание карт классов трафика
class-map IPP0
match dscp 0
class-map IPP1
match dscp 1
class-map IPP3
match dscp 2
class-map IPP5
match dscp 3
```



```
# Создание профиля трафика
traffic-profile profile-dscp
  class IPP0
  class IPP1
  class IPP2
  class IPP3
  class IPP5

# Создание планировщика очередей трафика
traffic-scheduler pqwrr.0
  queue 0
    class IPP0 cir mbps 1 pir mbps 2
    exit
  queue 1
    class IPP1 cir kbps 500 pir kbps 800
    exit
  queue 2
    class IPP3 cir kbps 500 pir kbps 500
    exit
  queue 3
    class IPP5 cir kbps 100 pir kbps 250
    exit

# Применение планировщика очередей к политике обслуживания
service-policy policer.1
  scheduler pqwrr.0

# Применение политики обслуживания к экземпляру сервиса
port ge0
  service-instance 4ge0
  service-policy policy in
```

Обратите внимание на значения DSCP в картах классов:

- DSCP 0 = TOS 0
- DSCP 1 = TOS 4
- DSCP 2 = TOS 8
- DSCP 3 = TOS 12

Именно эти значения TOS (Type of Service) будут использованы в работе утилиты `iperf3` далее по тексту.



На Generator-2 запустите четыре экземпляра утилиты `iperf3` в режиме сервера:

```
# Удаление ранее запущенных экземпляров серверов iperf3
pkill -f iperf3

# Запуск экземпляров сервера iperf3 на 4 портах для 4 потоков данных
iperf3 -s -D -p 5210
iperf3 -s -D -p 5211
iperf3 -s -D -p 5212
iperf3 -s -D -p 5213
```

На Generator-1 запустите четыре экземпляра утилиты `iperf3` в режиме клиента :

```
iperf3 -c 192.168.2.2 -p 5210 -t 300 -S 0 -b 10M -l 8 --logfile out_iperf_2 &
iperf3 -c 192.168.2.2 -p 5211 -t 300 -S 4 -b 10M -l 8 --logfile out_iperf_3 &
iperf3 -c 192.168.2.2 -p 5212 -t 300 -S 8 -b 10M -l 8 --logfile out_iperf_4 &
iperf3 -c 192.168.2.2 -p 5213 -t 300 -S 12 -b 10M -l 8 --logfile out_iperf_5 &
```

Утилита `iperf3` трафика будет генерировать трафик в течении 5 минут, как это задано значением при ключе `-t` .

Перейдите на ECO-1 и, при необходимости, очистите данные статистики по очередям командой `clear counters port ge1 queues` .

На ECO-1 проверьте наполняемость очередей пакетами:

```
ecorouter#sh counters port ge1 queues brief
Port ge1
Service instance 4ge1
Traffic scheduler pqwrr.0
Early detection algorithm: None
QoS Statistics:
                                RED-drop
                                packets/bytes
                                0/0
                                Match
queue class                    packets/bytes
0 IPP0                         1230/1805146
1 IPP1                         569/855727
2 IPP3                         428/642253
3 IPP5                         151/222875
4 ---                          0/0
5 ---                          0/0
6 ---                          0/0
7 default                      4/222
```

На ECO-1 проверьте скорость передачи данных с разбивкой по очередям:

```
ecorouter#show counters port ge1 queues-speed
```

```
Port ge1
```

```
Service instance 4ge1
```

queue class	match/bps	pass/bps	match/pps	pass/pps
0 IPP0	2.131 m	1.982 m	176	163
1 IPP1	956.848 k	787.280 k	79	65
2 IPP3	662.120 k	512.736 k	54	42
3 IPP5	387.584 k	246.272 k	32	20
4 -----	0	0	0	0
5 -----	0	0	0	0
6 -----	0	0	0	0
7 default	0	0	0	0
total	4.138 m	3.528 m	341	290

Данная команда выдаёт моментальную скорость передачи и выдаваемые показатели колеблются, тем не менее, они держатся приблизительно в пределах заданных в планировщике очередей.

Чтобы убедиться, что на длительном периоде (5 минут) средняя скорость довольно точно укладывается в заданные планировщиком показатели PIR, обратитесь к содержимому файлов `out_iperf_2`, `out_iperf_3`, `out_iperf_4`, `out_iperf_5` на Generator-1. Файлы являются результатом работы экземпляров утилиты `iperf3`, содержат посекундные данные скорости и усреднённые значения в конце.

33 IPv6

33.1 Введение в IPv6

Основная причина перехода на IPv6 — гораздо большее количество адресов, но кроме этого IPv6 предлагает ряд новых функций:

- В IPv6 отсутствует широковещательная передача данных, вместо неё используется многоадресная передача;
- Автоматическая настройка адреса шлюза по умолчанию и других параметров хоста без отслеживания состояния и без использования DHCPv6-сервера.
- Перенумерация адресов из IPv4 в IPv6
- Мобильность — в IPv6 есть встроенная поддержка мобильных устройств. Абонент может перемещаться от сети к сети сохраняя при этом свой текущий IPv6 адрес.
- Отсутствие NAT и PAT — количества адресов IPv6 достаточно для адресации всех существующих устройств без разделения на публичные и локальные.
- В IPv6 встроена поддержка IPsec. Его не нужно применять специально, поскольку шифрование встроено в протокол.
- Усовершенствованный заголовок — в IPv6 проще заголовки и нет контрольных сумм. Он также имеет метку потока, которая используется для быстрого определения принадлежности определённых пакетов к одному и тому же потоку.
- IPv4 и IPv6 несовместимы, но существует множество способов туннелирования для передачи IPv6 по сетям IPv4, равно как и наоборот. Возможна также одновременная работа IPv4 и IPv6.

33.1.1 Формат адреса IPv6

X:X:X:X:X:X:X:X, где X это 16-битное шестнадцатеричное число. По устоявшейся терминологии такое число называют хекстетом (по аналогии с октетом в IPv4).

Пример:

2041:1234:140F:1122:AB91:564F:875B:131B

Можно применять укороченную запись:



- 2041:0000:140F:0000:0000:0000:875B:131B
- 2041:0000:140F::875B:131B

Последовательность нулей опускается и заменяется двойным двоеточием. Это возможно единожды в адресе. IPv6 автоматически дополнит адрес нулями до 128 бит.

Хекстет из четырёх нулей заменяется одним нулём.

- 2041:0000:140F::875B:131B
- 2041:0:140F::875B:131B

Нули предшествующие числу также могут быть опущены.

- 2001:0001:0002:0003:0004:0005:0006:0007
- 2001:1:2:3:4:5:6:7

33.1.2 Префикс подсети

Пример префикса подсети IPv6: 2001:1111:2222:3333::/64, где число после "/" это количество бит, которые будут использованы в качестве префикса, а "::" это сокращённая запись ещё 16 нулей (хостовой части адреса).

Первый хекстет это 16 бит, два первых хекстета это 32 бита, три первых хекстета это 48 битов и так далее до 128 битов — так мы можем определять длину префикса. Например префикс с длиной подсети 64 — это четыре хекстета слева и нули обозначаемые двойным двоеточием справа.

В том случае, если префикс подсети не кратен четырём, для вычисления префикса придётся прибегнуть к битовым вычислениям: перевести хекстет в двоичную систему, биты справа от битов префикса установить в ноль и вернуть число в шестнадцатеричную систему.

33.1.3 Типы адресов IPv6

Одноадресные (unicast) адреса IPv6 похожи на адреса IPv4. Они предназначены для конфигурирования одного интерфейса чтобы он мог получать и отправлять IPv6 пакеты.

Существуют различные unicast адреса IPv6:

- Глобальные unicast IPv6 адреса аналогичны публичным адресам IPv4. Адреса могут быть использованы в интернете. Главное отличие от IPv4 в том, что количества IPv6 адресов хватает для адресации всех устройств в Сети.
- Уникальные локальные адреса аналогичны приватным адресам IPv4. Эти адреса можно использовать в частной локальной сети без выхода в интернет или с выходом в интернет через IPv6 NAT. Уникальные локальные адреса не нужно регистрировать в специализированном органе.
- Локальные адреса link-local — адреса работающие только в локальном соединении, эти адреса никогда не маршрутизируются. Адреса link-local используются для обмена пакетами IPv6 внутри одной подсети для: обнаружения соседства и как адреса next-hop в таблице маршрутизации.
- Местные локальные адреса site-local. Первоначально предполагалось использовать адреса site-local в качестве частных IPv6 диапазонов компаний. В настоящее время этот тип адресов устарел (смотрите RFC 3879) и вместо адресов site-local теперь используются уникальные локальные адреса.
- Неуказанный адрес имеет вид 0:0:0:0:0:0:0:0, или в короткой записи :: (два двоеточия). Такой адрес никогда не следует настраивать на хосте, он используется для указания на тот факт, что хост не имеет ни одного адреса.
- Адрес loopback-интерфейса имеет вид 0:0:0:0:0:0:0:1, или в короткой записи ::1. Устройства IPv6 могут использовать этот адрес для отправки IPv6 пакетов самим себе в целях тестирования и проверок. Loopback-адрес никогда не должен быть назначен ни на один физический интерфейс. Это эквивалент IPv4 адреса 127.0.0.1.

Multicast (многоадресатные) адреса в IPv6 используются как для протоколов маршрутизации, так и при передаче пользовательских данных. Для многоадресатной передачи используется префикс FF00::/8.

Первые восемь битов (1111 1111) означают multicast-адрес, следующие четыре бита используются для установления флагов. Следующие четыре бита задают ограничения передачи multicast-данных: внутри одной подсети, в рамках организации или глобально — в сеть Интернет.

Ниже приведены наиболее часто используемые multicast-адреса:

- FF02::1 — все узлы в локальном сегменте сети.
- FF02::2 — все маршрутизаторы в локальном сегменте сети.
- FF02::5 — все OSPFv3 маршрутизаторы.

- FF02::6 — все OSPFv3 DR маршрутизаторы.
- FF02::9 — RIPv6 маршрутизаторы.
- FF02::A — EIGRP маршрутизаторы.

Anycast (всеадресатные) IPv6 адрес может быть назначен нескольким устройствам и объявлен в протоколе маршрутизации. Когда происходит отправка пакета на anycast адрес, он будет отправлен на ближайший интерфейс. Не существует специальных префиксов для anycast адресов. Любой unicast адрес используемый на более чем одном устройстве становится anycast адресом. Единственное отличие заключается в том, что необходимо будет настроить устройство и указать, что адрес будет использоваться для anycast.

33.1.4 Протокол обнаружения соседей IPv6

Одно из отличий IPv6 от IPv4 состоит в отсутствии протокола разрешения адресов ARP (Address Resolution Protocol). Функции ARP в IPv6 выполняет протокол обнаружения соседей NDP (Neighbor Discovery Protocol).

NDP использует ICMP и групповые адреса запрашиваемых узлов для обнаружения MAC-адресов конечных узлов IPv6 в этой же подсети. Вся процедура проводится с помощью двух сообщений:

- Запрос о соседстве.
- Объявление о соседстве.

Запрос о соседстве в первую очередь используется для поиска MAC-адресов других IPv6 адресов в одной подсети. Также запрос о соседстве используется для определения дублирующих адресов DAD (Duplicated Address Detection). В пакете с данным сообщением исходный адрес будет соответствовать адресу хоста, который отправляет запрос соседу. Адресом назначения будет групповой адрес запрошенного узла. Данное сообщение также включает в себя MAC-адрес отправителя.

Использование групповых адресов запрашиваемых узлов (solicited-node multicast addresses) в качестве адресов назначения гораздо эффективнее, чем широковещательные запросы в IPv4.

Каждое IPv6 устройство вычисляет групповой адрес запрашиваемых узлов: оно берёт групповой адрес FF02::1:FF/104 и дополняет его шестью последними шестнадцатеричными цифрами из своего IPv6 адреса. Затем устройство присоединяется к этому групповому адресу и "прослушивает" его.



Когда устройству нужно узнать MAC-адрес другого устройства в сети, оно отправляет запрос о соседстве на групповой адрес запрашиваемых узлов этого устройства. Хост может вычислить групповой адрес запроса узла другого хоста, поскольку он знает групповой multicast-адрес и знает адрес IPv6, до которого он хочет добраться.

В результате только нужный удалённый хост получает запрос о соседстве.

Сообщение запроса о соседстве используется также для проверки доступности узла. В этом случае, адрес назначения будет одноадресатным с адресом удалённого хоста.

Когда удалённый хост получает запрос о соседстве, он отвечает объявлением о соседстве. В сообщении — адрес источника это IPv6 данного хоста, а адрес назначения это IPv6 адрес удалённого хоста, который отправил запрос о соседстве. Самая важная часть этого сообщения это MAC-адрес хоста. Когда хост получит объявление о соседстве, два IPv6 хоста могут начать коммуницировать друг с другом.

Объявление о соседстве также используется при изменении MAC-адреса хоста. При отправке этого сообщения адресом назначения будет многоадресатный адрес всех узлов.

Постройте схему в соответствии с рисунком ниже.



На обоих маршрутизаторах включите режим поддержки IPv6 и перезагрузитесь.

```
end
enable
configure

#Включение работы с IPv6 и перезагрузка
ipv6 unicast-routing
write
end
reload
```

Настройте R1:

```
enable
configure
hostname R1

#Настройка интерфейса и порта
interface 2ge0
```

```
ipv6 address 2001:DB8:12::1/64
exit
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0
```

Настройте R2:

```
enable
configure
hostname R2

#Настройка интерфейса и порта
interface 2ge0
ipv6 address 2001:DB8:12::2/64
exit
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0
```

На R1 проверьте соседство

```
R1#show ipv6 neighbors
Total neighbors: 4
```

Interface	IPv6 Address	MAC Address	Type
2ge0	fe80::1e87:76ff:fe40:201	1c87.7640.0201	
2ge0	2001:db8:12::1	1c87.7640.0101	SELF
2ge0	2001:db8:12::2	1c87.7640.0201	
2ge0	fe80::1e87:76ff:fe40:101	1c87.7640.0101	SELF LINK_LOCAL

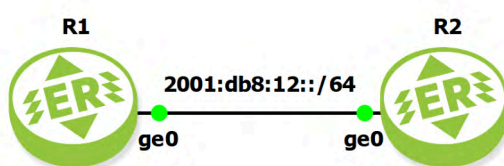
Вы можете видеть результат работы протокола обнаружения соседей NDP. Для установления соседства были автоматически присвоены link-local адреса и произошёл обмен присвоенными IPv6 адресами.

Дополнительно вы можете проверить связь между маршрутизаторами командой `ping ipv6 2001:DB8:12::2` с R1 и `ping ipv6 2001:DB8:12::1` с R2.

33.1.5 Автоконфигурация без сохранения состояния для IPv6

Автоконфигурация без сохранения состояния (Stateless address autoconfiguration, SLAAC) работает подобно простому DHCP-серверу для IPv6. Маршрутизаторы, работающие по протоколу IPv6, могут предоставлять префикс сети и адрес шлюза клиентам, ищущим адрес IPv6. IPv6 использует протокол NDP (Neighbor Discovery Protocol), и одним из преимуществ этого протокола являются сообщения RS (Route Solicitation) и (RA) Router Advertisement, которые помогают устройству IPv6 автоматически настраивать адрес IPv6.

Постройте схему в соответствии с рисунком ниже.



Мы используем два маршрутизатора. У R2 будет адрес IPv6 и он будет отправлять объявления Router Advertisement. R1 будет использовать это для настройки своего собственного адреса IPv6.

Включите IPv6 на обоих маршрутизаторах и перезагрузите их:

```
enable
configure
ipv6 unicast-routing
write
end
reload
```

Настройте R1:

```
enable
configure
interface 2ge0
ipv6 address autoconf
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0
```

Обратите внимание, что мы не указали IPv6 адрес, но вместо этого ввели команду автоконфигурации адреса `ipv6 address autoconf`.

Настройте R2:

```
enable
configure
interface 2ge0
ipv6 address 2001:db8:12::2/64
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0
```

На R2 настроен обычный IP адрес с длиной префикса 64. На его основе, с помощью алгоритма EUI 64, R1 сможет создать свой собственный адрес.

На R1 введите:

```
show interface | grep Ethernet\|inet6
Ethernet address: 1c87.7640.0101 (auto)
inet6 2001:db8:12:0:1e87:76ff:fe40:101/64 autoconfig
```

В получившемся адресе можно увидеть, что: первые 64 бита адреса достались R1 от R2, а оставшиеся преобразованы из MAC-адреса по алгоритму EUI 64.

ВНИМАНИЕ! Автоконфигурация работает только с адресами с длиной префикса в 64 бита.

Если мы смогли сделать это, возникает вопрос — для чего тогда нужен DHCPv6? Не следует забывать что DHCP, кроме того чтобы раздавать IPv6 адреса, может делать ещё множество других полезных вещей:

- Регистрировать имена хостов компьютеров на DNS-сервере.
- Содержать список DNS или WINS серверов.
- Содержать IPv6 адрес менеджера звонков (для VoIP телефонии) или беспроводного LAN-контроллера (для точек доступа).

DHCP, конечно, также доступен для IPv6 и носит название DHCPv6. Большое различие между DHCP для IPv4 и DHCPv6 заключается в том, что в DHCPv6 больше не используется широковещательный трафик. Когда устройство IPv6 ищет сервер DHCPv6, оно отправляет многоадресные пакеты на FF02::1:2. Маршрутизаторы пересылают эти пакеты на серверы DHCP.

Команды управления автоконфигурацией



Команда в режиме конфигурирования L3 интерфейса `ipv6 nd managed-config-flag` устанавливает M-flag (Managed Address Configuration flag) в сообщении RA в единицу и переключает клиентов на использование DHCPv6 для получения IPv6-адресов. Автоконфигурация (SLAAC) может при этом не использоваться или использоваться как дополнительный метод. Отрицание команды с помощью `no` устанавливает M-flag снова в ноль и переключает клиентов на использование автоконфигурации SLAAC. По умолчанию команда отключена.

Команда в режиме конфигурирования L3 интерфейса `ipv6 nd other-config-flag` устанавливает O-flag (Other Configuration flag) в сообщении RA в единицу и указывает клиентам, что они должны обращаться к DHCPv6-серверу для получения дополнительной конфигурационной информации (например DNS), даже если адреса они получают через SLAAC. Отрицание команды с помощью `no` устанавливает O-flag снова в ноль и переключает клиентов на использование автоконфигурации SLAAC. По умолчанию команда отключена.

Таблица 120 — Влияние М и О флагов на режим работы хоста

M-flag	O-flag	Режим работы хоста
0	0	Только SLAAC — только автоконфигурация
0	1	SLAAC + Stateless DHCPv6 — адреса через SLAAC, остальная информация через DHCPv6
1	0	Stateful DHCPv6 — адреса только через DHCPv6
1	1	Stateful DHCPv6 + доп. информация — полное управление через DHCPv6

33.2 Маршрутизация IPv6

33.2.1 Конфигурация статического маршрута IPv6

Если вы умеете конфигурировать статический маршрут для IPv4, у вас не должно возникнуть проблем и с IPv6. Конфигурация и синтаксис сходные, есть только небольшие различия.

33.2.1.1 Конфигурация

Постройте схему в соответствии с рисунком ниже.





Рисунок 76

R1 и R2 последовательно соединены. На R2 настроен loopback-интерфейс с IPv6 адресом **2001:DB8:2:2::2/64**.

Обратите внимание, если вы работаете в виртуальной среде, позаботьтесь о том чтобы устройствам были заданы различные MAC-адреса.

Настройте R1:

```
end
enable
configure
hostname R1

#Включение работы с IPv6 и перезагрузка
ipv6 unicast-routing
write
end
reload
```

ipv6 unicast-routing — включает работу IPv6 в плоскости пользовательских данных.

```
#Настройка интерфейса и порта
enable
configure
interface 2ge0
ipv6 address 2001:DB8:12:12::1/64
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0
exit
exit

#Настройка IPv6 маршрута
ipv6 route 2001:DB8:2:2::/64 2001:DB8:12:12::2
write
```


Аналогичным образом настройте R2 с loopback-интерфейсом:

```
end
enable
configure
hostname R2

#Включение работы с IPv6 и перезагрузка
ipv6 unicast-routing
write
end
reload
```

```
#Настройка интерфейса и порта
enable
configure
interface 2ge0
ipv6 address 2001:DB8:12:12::2/64
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0

#Настройка loopback-интерфейса
interface loopback.0
ipv6 address 2001:DB8:2:2::2/64
exit
exit
```

Проверьте созданный на R1 маршрут:

```
R1#show ipv6 route static
IP Route Table for VRF "default"
S      2001:db8:2:2::/64 [1/0] via 2001:db8:12:12::2, 2ge0, 00:13:56
```

Проверьте доступность loopback-интерфейса с R1:

```
R1#ping 2001:db8:2:2::2
PING 2001:db8:2:2::2 2 56 data bytes
64 bytes from 2001:db8:2:2::2: icmp_seq=1 ttl=64 time=22.8 ms
64 bytes from 2001:db8:2:2::2: icmp_seq=2 ttl=64 time=21.5 ms
64 bytes from 2001:db8:2:2::2: icmp_seq=3 ttl=64 time=20.6 ms
```

Одно из различий между IPv4 и IPv6 заключается в том, что IPv6 генерирует адрес link-local для каждого интерфейса. Эти link-local адреса также используются протоколами

маршрутизации, такими как RIPv6, EIGRP, OSPFv3 и т. д., в качестве адресов следующего перехода (next hop). Проверьте, какой link-local адрес у R2:

```
R2#show interface 2ge0 | include Link-local
Link-local address is fe80::1e87:76ff:fe40:101/64
```

А теперь используйте этот адрес в качестве следующего перехода в маршруте на R1. Когда в качестве следующего перехода используется глобальный unicast адрес, маршрутизатор может просмотреть таблицу маршрутизации и выяснить, какой исходящий интерфейс использовать для достижения этого глобального адреса unicast. В случае с адресами link-local маршрутизатор не имеет информации, какой исходящий интерфейс использовать, поэтому необходимо будет указать как адрес link-local, так и исходящий интерфейс **2ge0**:

```
R2(config)#no ipv6 route 2001:db8:2:2::/64
R2(config)#ipv6 route 2001:db8:2:2::/64 fe80::1e87:76ff:fe40:101 2ge0
```

Проверьте таблицу маршрутизации `show ipv6 route static` и `ping 2001:DB8:2:2::2`.

Статический маршрут по умолчанию

Как и в IPv4 в IPv6 можно создавать статические маршруты по умолчанию. Маршрут по умолчанию состоит из одних нулей `::` и имеет префикс нулевой длины `/0`. Это полная аналогия адреса **0.0.0.0/0** в IPv4.

Настройка адреса по умолчанию в качестве префикса назначения будет означать, что все пакеты для которых не будет найден более специфический маршрут будут отправлены на адрес маршрута по умолчанию.

В качестве следующего перехода можно указать: имя интерфейса, глобальный unicast адрес или link-local адрес.

Настройте маршрут с адресом по умолчанию `::/0` в качестве префикса назначения и именем интерфейса на R1:

```
no ipv6 route 2001:db8:2:2::/64
ipv6 route ::/0 2ge0
```

На R1 проверьте таблицу маршрутизации `show ipv6 route static` и `ping 2001:DB8:2:2::2`.

Настройте маршрут с адресом по умолчанию `::/0` в качестве префикса назначения и unicast адресом R2 в качестве следующего перехода:

```
no ipv6 route ::/0
ipv6 route ::/0 2001:db8:12:12::2 2ge0
```

На R1 проверьте таблицу маршрутизации `show ipv6 route static` и `ping`

```
2001:DB8:2:2::2 .
```

Замените глобальный unicast адрес следующего перехода на link-local адрес маршрутизатора R2.

```
no ipv6 route ::/0
ipv6 route ::/0 fe80::1e87:76ff:fe40:101 2ge0
```

На R1 проверьте таблицу маршрутизации `show ipv6 route static` и `ping 2001:DB8:2:2::2`.

Статический маршрут хоста

Можно также создать статические маршруты для одного адреса IPv6, называемые статическим маршрутом хоста. Здесь приводятся те же примеры что и ранее, но на этот раз нужно будет создать запись для **2001:DB8:2:2::2**.

Сначала пропишите интерфейс в качестве следующего перехода:

```
no ipv6 route ::/0
ipv6 route 2001:DB8:2:2::2/128 2ge0
```

Теперь подставьте в качестве следующего перехода глобальный unicast адрес R2.

```
no ipv6 route 2001:DB8:2:2::2/128
ipv6 route 2001:DB8:2:2::2/128 2001:DB8:12:12::2
```

Теперь link-local адрес:

```
no ipv6 route 2001:DB8:2:2::2/128
ipv6 route 2001:DB8:2:2::2/128 fe80::1e87:76ff:fe40:101 2ge0
```

После каждого варианта проверяйте таблицу маршрутизации `show ipv6 route static` и `ping 2001:DB8:2:2::2`.

Плавающий статический маршрут

Статические маршруты имеют очень низкую административную дистанцию равную единице. Это означает, что маршрутизатор будет предпочитать статический маршрут любым другим маршрутам, которые были изучены через протоколы динамической маршрутизации. Если вы хотите использовать статический маршрут в качестве резервного маршрута, вам придётся изменить его административную дистанцию. Это называют плавающим статическим маршрутом.

Добавьте к схеме ещё один маршрутизатор.

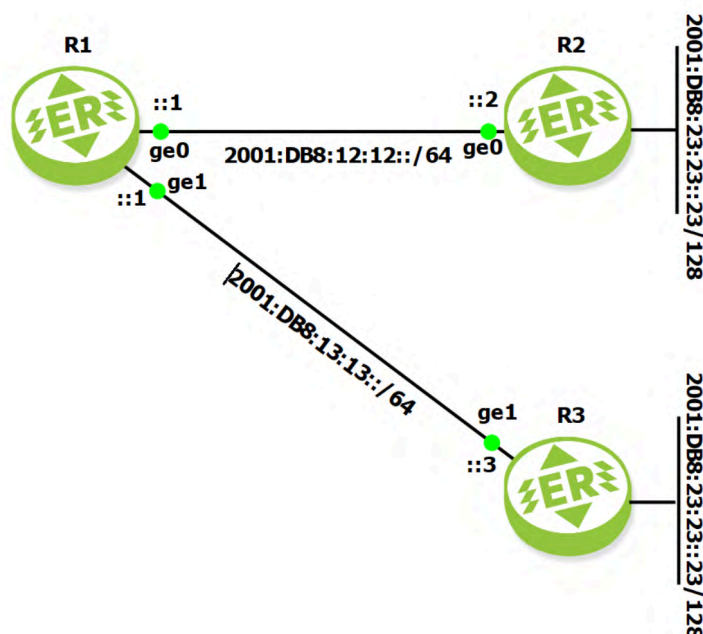


Рисунок 77

Настройте R3:

```
end
enable
configure
hostname R3

#Включение работы с IPv6 и перезагрузка
ipv6 unicast-routing
write
end
reload
```

```
#Настройка интерфейса и порта
interface 2ge1
ipv6 address 2001:DB8:13:13::3/64
port ge1
service-instance 4ge1
encapsulation untagged
connect ip interface 2ge1

#Настройка loopback-интерфейса
interface loopback.0
ipv6 address 2001:DB8:23:23::23/128
```

Перенастройте loopback интерфейс на R2:

```
interface loopback.0
no ipv6 address 2001:db8:2:2::2/64
ipv6 address 2001:DB8:23:23::23/128
```

Настройте новый интерфейс на R1:

```
interface 2ge1
ipv6 address
ipv6 address 2001:DB8:13:13::1/64
port ge1
service-instance 4ge1
encapsulation untagged
connect ip interface 2ge1
```

R3 добавлен в топологию, и вы настроили тот же loopback-адрес **2001:DB8:23:23::23/128** что и на R2. R3 будет использоваться как основной путь для достижения этого адреса. При потере связи, должен использовать R2.

Вот статический маршрут для использования R3 в качестве основного пути:

```
no ipv6 route 2001:db8:2:2::/64
ipv6 route 2001:DB8:23:23::/64 2001:DB8:13:13::3
```

Сначала установите маршрут до R3 через глобальный unicast адрес с увеличенной метрикой равной двум:

```
ipv6 route 2001:DB8:23:23::/64 2001:db8:13:13::3 2
```

Проверьте таблицу маршрутизации:

```
R1(config)#do show ipv6 route static
IP Route Table for VRF "default"
S      2001:db8:23:23::/64 [1/0] via 2001:db8:12:12::2, 2ge0, 00:45:30
```

Заданный маршрут на R3 отсутствует в таблице.

Но он присутствует в конфигурации:

```
R1(config-if)#do show run | include ipv6 route
ipv6 route 2001:db8:23:23::/64 2001:db8:12:12::2
ipv6 route 2001:db8:23:23::/64 2001:db8:13:13::3 2
```

Теперь принудительно отключите интерфейс **2ge0** ведущий к R2:

```
R1(config)#interface 2ge0
R1(config-if)#shutdown
2025-07-04 16:03:01      INFO      Interface 2ge0 changed state to down
```

Проверьте таблицу маршрутизации:



```
R1(config-if)#do show ipv6 route static
IP Route Table for VRF "default"
S      2001:db8:23:23::/64 [2/0] via 2001:db8:13:13::3, 2ge1, 01:10:13
```

Теперь в таблице маршрутизации появился маршрут к R3.

Также как и в предыдущих примерах, вы можете использовать link-local next hop адрес:

```
R1(config)#ipv6 route 2001:db8:23:23::/64 fe80::1e87:76ff:fe40:101 2ge0 2
```

33.2.2 Сравнение OSPFv2 и OSPFv3

OSPFv2 и OSPFv3 очень похожи. OSPFv3 по-прежнему устанавливает соседства, имеет зоны, различные типы сетей, те же метрики, запускает SPF и т. д. Однако есть некоторые различия. OSPFv2 работает поверх IPv4, а поскольку OSPFv3 работает с IPv6, в него пришлось внести некоторые изменения:

- Адреса локального соединения (link-local): пакеты OSPFv3 берут своё начало из link-local адресов IPv6.
- Соединения, а не сети: OSPFv3 использует термин соединения, тогда как в OSPFv2 использовался термин сети.
- Новые типы LSA (Link State Advertisement) : есть два новых типа LSA, и изменились типы LSA 1 и 2.
- Команды интерфейса: OSPFv3 использует команды в контексте интерфейса для включения на нём протокола. Команда network, как это делалось в OSPFv2, больше не используется.
- Идентификатор маршрутизатора OSPFv3: OSPFv3 не может самостоятельно установить собственный идентификатор маршрутизатора, как это делает OSPFv2. Вместо этого вам нужно вручную настроить идентификатор маршрутизатора. Он настраивается как 32-битное значение (в виде IPv4 адреса), как и в OSPFv2.
- Несколько префиксов на интерфейс: если у вас есть несколько префиксов IPv6 на интерфейсе, то OSPFv3 будет объявлять их все.
- Область лавинной рассылки (flooding scope): в OSPFv3 для разных LSA может быть установлена различная область лавинной рассылки.
- Несколько экземпляров на одно соединение: вы можете запустить несколько экземпляров OSPFv3 на одном соединении.

- Аутентификация: OSPFv3 не использует простой текст или аутентификацию MD5, как OSPFv2. Вместо этого он использует аутентификацию IPsec IPv6.
- Префиксы в LSA: OSPFv2 показывает сети в LSA как сеть + маска подсети, OSPFv3 показывает префиксы как префикс + длина префикса.

33.2.2.1 Типы LSA

В OSPFv3 есть два новых LSA, а некоторые типы LSA были переименованы. Ниже — таблица всех типов OSPFv2 и OSPFv3 LSA:

OSPFv3		OSPFv2	
Код типа LSA	Наименование	Код типа LSA	Наименование
0x2001	Router LSA	1	Router LSA
0x2002	Network LSA	2	Network LSA
0x2003	Inter-Area Prefix LSA	3	Network Summary LSA
0x2004	Inter-Area Router LSA	4	ASBR Summary LSA
0x4005	AS-External LSA	5	AS-External LSA
0x2006	Group Membership LSA	6	Group Membership LSA
0x2007	Type-7 LSA	7	NSSA External LSA
0x0008	Link LSA		
0x2009	Intra-Area Prefix LSA		

Типы LSA остались прежними, за исключением того, что тип 3 теперь называется LSA межзональных префиксов, а тип 4 называется LSA межзональных маршрутизаторов. Последние два типа, LSA соединения и LSA внутризональных префиксов, впервые появились в OSPFv3.

В OSPFv2 LSA типа 1 и типа 2 используются для передачи топологии и сетевой информации. Одиночный LSA содержит информацию о топологии и используемых сетях.

Если вы вносите простое изменение, например, меняете IP-адрес на одном из маршрутизаторов, то сама топология не меняется. В OSPFv2 необходимо было сделать массовую рассылку LSA нового типа 1 и, возможно, LSA типа 2. Другие маршрутизаторы, которые получили новые LSA, должны были пересчитать SPF, даже если топология не изменилась.

В OSPFv3 это было изменено путём создания разделения между префиксами и деревом SPF. В LSA типа 1 и 2 нет информации о префиксах, в этих LSA вы найдёте только топологические смежности, в них вы не найдёте никаких префиксов IPv6. Префиксы теперь объявляются в LSA типа 9, а link-local адреса, которые используются

для следующих переходов, объявляются в LSA типа 8. LSA типа 8 рассылаются только по локальному каналу, LSA типа 9 рассылаются внутри зоны. Разработчики OSPFv3 могли бы включить link-local адреса в LSA типа 9, но поскольку они требуются только в локальном соединении, это было бы пустой тратой ресурсов.

Разделяя дерево SPF и префиксы, OSPFv3 становится более эффективным. Когда link-local адрес на интерфейсе изменяется, маршрутизатору нужно только разослать обновлённый LSA соединения и LSA внутризонального префикса. Поскольку топология не меняется, не нужно рассылать LSA типа 1 и 2. В этом случае другим маршрутизаторам не придётся запускать SPF (алгоритм вычисления кратчайшего пути).

33.2.2.2 Область лавинной рассылки (flooding scope)

В таблице с типами LSA выше, вы могли заметить, что типы LSA OSPFv3 имеют шестнадцатеричные значения. Первая их часть определяет область лавинной рассылки LSA:

- 0x0: область link-local, используемая для LSA соединения, нового типа LSA в OSPFv3.
- 0x2: область зоны, используемая для LSA, которые рассылаются внутри одной зоны. Используется для маршрутизатора, сети, межзональных префиксов, межзональных маршрутизаторов и LSA внутризональных префиксов.
- 0x4: область автономных систем (AS), используемая для LSA, которые рассылаются внутри домена маршрутизации OSPFv3, используется для внешних LSA.

33.2.2.3 Заголовки

OSPFv2 and OSPFv3 используют различные заголовки. Вот некоторые различия:

Field	OSPFv3	OSPFv2
Размер заголовка	16 байт	24 байт
Идентификаторы зон и маршрутизаторов	32 бита	32 бита
Instance ID	да	нет
Authentication	IPSec	текст или MD5

Идентификатор экземпляра — это новое поле, которое можно использовать для запуска нескольких экземпляров OSPFv3 на одном соединении. Маршрутизаторы OSPFv3 станут соседями только в том случае, если идентификатор экземпляра одинаков (по умолчанию равен 0). Это позволяет запускать OSPFv3 в широковещательной сети и



формировать соседские отношения только с определёнными соседями, которые используют тот же идентификатор экземпляра.

33.2.3 Конфигурация IPv6 OSPFv3

Для IPv4 использовался протокол OSPFv2, для работы IPv6 протокол был модернизирован до версии OSPFv3. Это два разных протокола маршрутизации и в этом пункте будет разобрано как сконфигурировать OSPFv3 для маршрутизации IPv6 данных.

Постройте схему в соответствии с рисунком ниже.



Рисунок 78

Сконфигурируйте интерфейсы и IPv6 адреса на них и запустите процессы OSPFv3 на маршрутизаторах. Вам не придётся настраивать какие-либо глобальные unicast IPv6-адреса на интерфейсах, поскольку OSPFv3 использует link-local адреса для соседства и отправки LSA.

Сконфигурируйте R1:

```
end
enable
configure
hostname R1

#Включение работы с IPv6 и перезагрузка
ipv6 unicast-routing
write
end
reload
```

```
enable
configure

#Настройка интерфейса
interface 2ge0
ipv6 ospf6 area 0
port ge0
service-instance 4ge0
encapsulation untagged
```

```
connect ip interface 2ge0

#Настройка loopback интерфейса
interface loopback.0
ipv6 address 2001::1/128
ipv6 ospf6 area 0
ipv6 ospf6 passive
exit

#Запуск процесса OSPFv3
router ospf6
ospf6 router-id 1.1.1.1
```

Аналогичным образом настройте маршрутизатор R2.:

```
end
enable
configure
hostname R2

#Включение работы с IPv6 и перезагрузка
ipv6 unicast-routing
write
end
reload
```

```
enable
configure

#Настройка интерфейса
interface 2ge0
ipv6 ospf6 area 0
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0

#Настройка loopback интерфейса
interface loopback.0
ipv6 address 2001::2/128
ipv6 ospf6 area 0
ipv6 ospf6 passive
exit

#Запуск процесса OSPFv3
```



```
router ospf6
ospf6 router-id 2.2.2.2

#Команды контроля результата
show ipv6 ospf6 database
show ipv6 route
show ipv6 ospf6 neighbor
```

Дайте команду `show ipv6 ospf6 neighbor`.

```
R1#show ipv6 ospf6 neighbor
Neighbor ID      Pri    DeadTime      State/IfState      Duration I/F[State]
2.2.2.2          1      00:00:35      Full/DR            02:44:01 2ge0[BDR]
```

Маршрутизатор R1 видит R2 в качестве своего соседа.

Дайте команду `show ipv6 route ospf`.

```
R1#show ipv6 route ospf
IP Route Table for VRF "default"
O        2001::2/128 [110/1] via fe80::1e87:76ff:fe40:201, 2ge0, 02:35:54
```

loopback интерфейс соседнего роутера доступен через link-local адрес маршрутизатора R1. Аналогичную команду можно повторить на R2.

Командой `ping ipv6 2001::2` вы можете проверить передачу данных между маршрутизаторами.

33.2.4 Маршрут по умолчанию IPv6 OSPFv3

Как и в OSPF IPv4 в OSPFv3 IPv6 возможно объявить маршрут по умолчанию.

Постройте схему в соответствии с рисунком ниже.

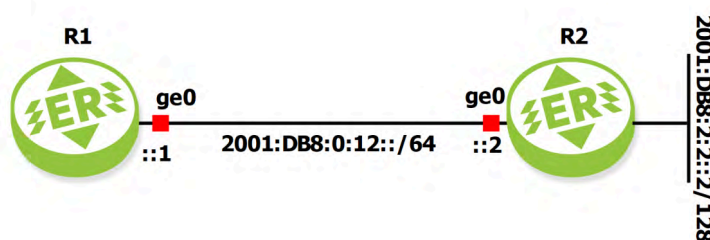


Рисунок 79

На R2 создайте интерфейс обратной связи с IPv6-адресом **2001:DB8:2:2::2/128**. Вам не нужно объявлять об этом в OSPFv3 напрямую, вместо этого нужно будет обратиться к этому адресу с R1 с помощью маршрута по умолчанию, который будет объявлять R2.

Настройте R1:

```
end
enable
configure
hostname R1

#Включение работы с IPv6 и перезагрузка
ipv6 unicast-routing
write
end
reload
```

```
enable
configure

#Настройка интерфейса
interface 2ge0
ipv6 address 2001:DB8:0:12::1/64
ipv6 ospf6 area 0
port ge0
ipv6 address
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0
exit

#Запуск процесса OSPFv3
router ospf6
ospf6 router-id 1.1.1.1
```

Аналогичным образом настройте R2:

```
end
enable
configure
hostname R2

#Включение работы с IPv6 и перезагрузка
ipv6 unicast-routing
write
end
reload
```

```
enable
configure

#Настройка интерфейса
interface 2ge0
ipv6 address 2001:DB8:0:12::2/64
ipv6 ospf6 area 0
port ge0
ipv6 address
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0
exit

#Настройка loopback интерфейса
interface loopback.0
no ip address
ipv6 address 2001:DB8:2:2::2/128
ipv6 ospf6 passive
exit

#Запуск процесса OSPFv3
router ospf6
ospf6 router-id 2.2.2.2
default-information originate always
```

Обратите внимание, прописывать команду `ipv6 ospf6 area 0` для адреса `loopback.0` и тем самым включать его в маршрутизацию OSPFv3 не потребовалось.

С другой стороны в настройки процесса OSPFv3 командой `default-information originate always` был включён маршрут по умолчанию. Эта команда подробно разобрана в подразделе по OSPF.

ВНИМАНИЕ! Чтобы настройки заданные после установления соседства вошли в силу необходимо перезапустить сессию OSPFv3 командой `clear ipv6 ospf6 process`.

Проверьте результат:

```
R1>show ipv6 ospf neighbor
Neighbor ID      Pri   DeadTime   State/IfState   Duration I/F[State]
2.2.2.2          1     00:00:38   Full/BDR        00:40:53 2ge0[DR]
```

R1 видит R2 в качестве соседа.

```
R1>show ipv6 route ospf
IP Route Table for VRF "default"
0      ::/0 [110/1] via fe80::1e87:76ff:fe40:201, 2ge0, 00:42:36
```



Маршрут до адреса по умолчанию построен.

И в завершении проверьте пересылку пакетов командой `ping ipv6 2001:db8:2:2::2`.

33.2.5 Фильтрация маршрутов IPv6 OSPFv3

Постройте схему в соответствии с рисунком ниже.

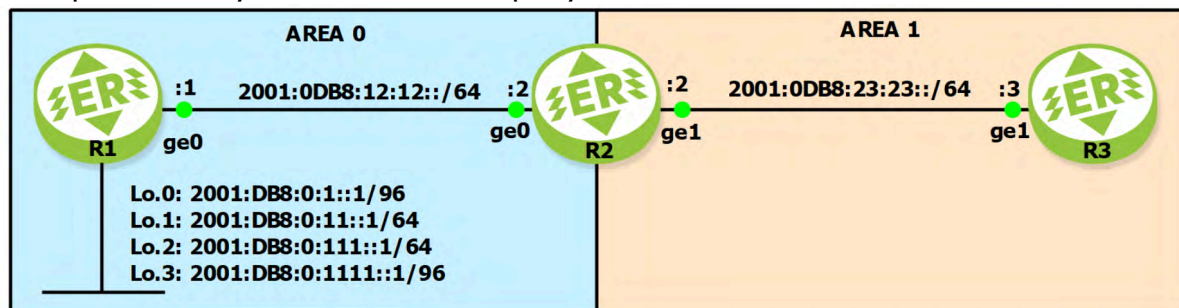


Рисунок 80

Включите IPv6 на всех маршрутизаторах и перезагрузите их:

```
end
enable
configure

#Включение работы с IPv6 и перезагрузка
ipv6 unicast-routing
write
end
reload
```

Настройте R1:

```
enable
configure
hostname R1

interface 2ge0
ipv6 address 2001:db8:12:12::1/64
ipv6 ospf6 area 0
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0
```



```
interface loopback.0
ipv6 address 2001:DB8:0:1::1/96
ipv6 ospf6 area 0
interface loopback.1
ipv6 address 2001:DB8:0:11::1/64
ipv6 ospf6 area 0
interface loopback.2
ipv6 address 2001:DB8:0:111::1/64
ipv6 ospf6 area 0
interface loopback.3
ipv6 address 2001:DB8:0:1111::1/96
ipv6 ospf6 area 0

router ospf6
ospf6 router-id 1.1.1.1
```

Настройте R2:

```
enable
configure
hostname R2

interface 2ge0
ipv6 address 2001:db8:12:12::2/64
ipv6 ospf6 area 0
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0

interface 2ge1
ipv6 address 2001:db8:23:23::2/64
ipv6 ospf6 area 1
port ge1
service-instance 4ge1
encapsulation untagged
connect ip interface 2ge1

router ospf6
ospf6 router-id 2.2.2.2
```

Настройте R3:

```
enable
configure
```



```
hostname R1

interface 2ge1
ipv6 address 2001:db8:23:23::3/64
ipv6 ospf6 area 1
port ge1
service-instance 4ge1
encapsulation untagged
connect ip interface 2ge1

router ospf6
ospf6 router-id 3.3.3.3
```

Проверьте, получил ли R3 все подсети, известные R1, по протоколу OSPFv3.

```
R3#show ipv6 route ospf
IP Route Table for VRF "default"
O  2001:db8::/48 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O  2001:db8:0:1::1/128 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O  2001:db8:0:11::/64 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O  2001:db8:0:11::1/128 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O  2001:db8:0:111::/64 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O  2001:db8:0:111::1/128 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O  2001:db8:0:1111::/96 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O  2001:db8:0:1111::1/128 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O  2001:db8:12:12::/64 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
```

В таблицу попали все loopback-адреса заданные на R1.

Теперь займёмся непосредственно фильтрацией маршрутов и отфильтруем адреса с длиной префикса 96.

Все действия по фильтрации маршрутов будут производиться на R2, поскольку он является граничным маршрутизатором (ABR) OSPF домена. Внутренняя логика работы OSPF позволяет осуществлять фильтрацию и суммаризацию только на ABR и ASBR маршрутизаторах.

В конфигурационном режиме консоли создайте следующий префиксный список: `ipv6 prefix-list SMALL_NETWORKS seq 5 permit 2001::/16 le 64`. Список префиксов с правилом `permit`, разрешает отправку информации о подсетях начиная с `2001::/16` и вплоть до префикса `/64` включительно. Сеть с префиксом `96` не попадает в этот диапазон.

Для того чтобы отфильтровать маршруты из области 0, на ABR (R2) в режиме конфигурирования маршрутизации OSPF (`router ospf6`) введите команду: `area 1`

`filter-list prefix SMALL_NETWORKS in`. Команда указывает, что в зону 1 можно пропустить (permit) только подсети, подпадающие под условия, указанные в `SMALL_NETWORKS`.

На R3 проверьте таблицу маршрутов:

```
R3#show ipv6 route ospf
IP Route Table for VRF "default"
O 2001:db8::/48 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O 2001:db8:0:11::/64 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O 2001:db8:0:111::/64 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O 2001:db8:12:12::/64 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
```

Подсеть с длиной префикса 96, как и ожидалось, пропала из списка.

33.2.6 Суммаризация маршрутов IPv6 OSPFv3

Так как внутри зоны на всех маршрутизаторах должен быть одинаковый набор LSA, выполнять суммаризацию внутри зоны недопустимо. Суммаризация производится только на границе между зонами, то есть на ABR и ASBR маршрутизаторах. Более того, суммаризация возможна только на том ABR, который является ABR-маршрутизатором для той зоны, в которой находятся суммируемые маршруты.

Для того чтобы суммаризация была эффективной и вообще возможной, нужно заранее правильно спроектировать адресное пространство. Префиксы должны суммироваться в одну суммарную сеть.

У суммарного маршрута длина префикса должна быть меньше, чем длины префиксов у суммируемых подсетей.

Постройте схему в соответствии с рисунком ниже.

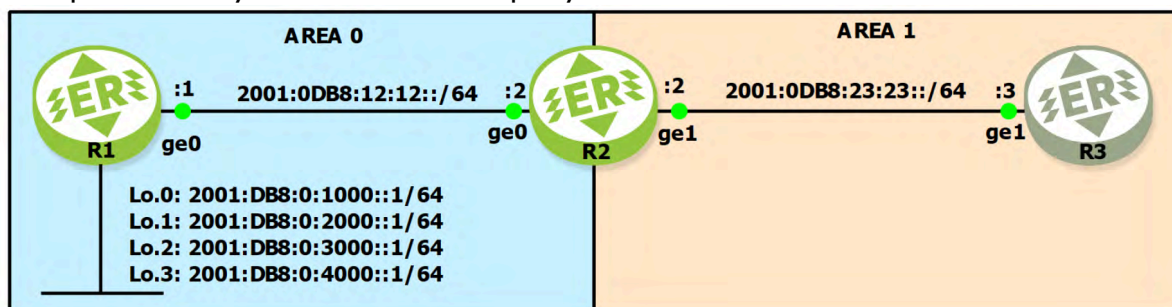


Рисунок 81

Настройте маршрутизаторы аналогично предыдущему пункту, за исключением loopback-адресов.

На R3 проверьте доступны ли все маршруты с R1:

```
R3#show ipv6 route ospf
IP Route Table for VRF "default"
O 2001:db8:0:1000::/64 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O 2001:db8:0:1000::1/128 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O 2001:db8:0:2000::/64 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O 2001:db8:0:2000::1/128 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O 2001:db8:0:3000::/64 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O 2001:db8:0:3000::1/128 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O 2001:db8:0:4000::/64 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O 2001:db8:0:4000::1/128 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O 2001:db8:12:12::/64 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
```

На R2 в режиме конфигурации OSPFv3 (команда `router ospf6`) введите команду `area 0 range 2001:db8::/50 advertise`.

На R3 снова проверьте маршруты:

```
R3#show ipv6 route ospf
IP RouteTable for VRF "default"
O 2001:db8::/50 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O 2001:db8:0:4000::/64 [110/2] via fe80::1e87:76ff:fe40:202
O 2001:db8:0:4000::1/128 [110/2] via fe80::1e87:76ff:fe40:202
O 2001:db8:12:12::/64 [110/2] via fe80::1e87:76ff:fe40:202
```

Как видно, в таблицу попал как суммаризованный маршрут так и подсеть 2001:db8:0:4000::/64. Другими словами этот маршрут не попал в суммаризованный маршрут. Почему? Потому что он не подпадает под длину префикса 50!

Поменяйте на R2 длину префикса с 50 на 49:

```
no area 0 range 2001:db8::/50 advertise
area 0 range 2001:db8::/49 advertise
```

На R3:

```
R3#show ipv6 route ospf
IP Route Table for VRF "default"
O 2001:db8::/49 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
O 2001:db8:12:12::/64 [110/2] via fe80::1e87:76ff:fe40:202, 2ge1
```

Теперь все loopback-адреса на R1 попали в суммаризованный маршрут.

Дополнительно, командой `ping` на R3 вы можете проверить каждый из loopback-адресов.

33.2.7 Поиск неисправностей в отношениях IPv6 OSPFv3 соседства

В этом пункте будут рассмотрены некоторые проблемы соседства OSPFv3. Большая часть того, что вы уже знаете об OSPFv2 для IPv4, применимо и к OSPFv3.

33.2.7.1 Идентификатор маршрутизатора OSPFv3

Постройте следующую схему с двумя маршрутизаторами в одной зоне AREA 0.

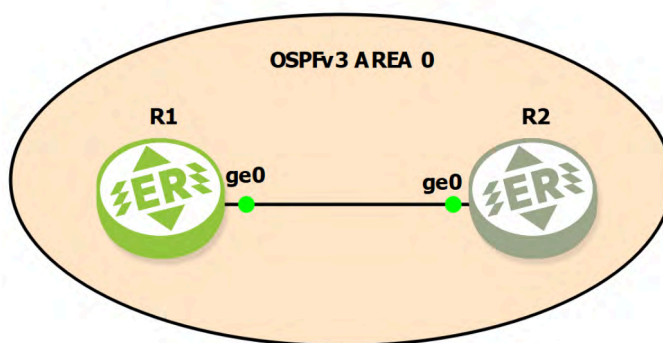


Рисунок 82

Задайте идентичную (за исключением имени маршрутизатора) конфигурацию обоим маршрутизаторам:

```
end
enable
configure
hostname R1

#Включение работы с IPv6 и перезагрузка
ipv6 unicast-routing
write

#Настройка интерфейса
interface 2ge0
ipv6 ospf6 area 0
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0

# Включение процесса OSPFv3
router ospf6
```

```
exit
```

По какой-то причине два маршрутизатора не могут стать соседями и нам предстоит понять по какой.

Узнайте link-local адреса обоих маршрутизаторов:

```
R1#show interface 2ge0 | grep Interface\|Link-local
Interface 2ge0 is up
Link-local address is fe80::1e87:76ff:fe40:101/64
```

```
R2#show interface 2ge0 | include Interface|Link-local
Interface 2ge0 is up
Link-local address is fe80::1e87:76ff:fe40:201/64
```

Протоколы маршрутизации IPv6 используют link-local адреса для соседства и следующих переходов (next-hop). Оба интерфейса имеют link-local адрес IPv6 и активны (up). На всякий случай выполните команду **ping**, чтобы убедиться в наличии подключения:

```
R1#ping fe80::1e87:76ff:fe40:101
PING fe80::1e87:76ff:fe40:101 1e87:76ff:fe40:101 56 data bytes
64 bytes from fe80::1e87:76ff:fe40:101%2ge0: icmp_seq=1 ttl=64 time=0.370
. . .
--- fe80::1e87:76ff:fe40:101 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 0.092/0.186/0.370/0.129 ms
```

Причина не найдена, проверьте соседство:

```
R1#show ipv6 ospf6 neighbor
Neighbor ID      Pri    DeadTime      State/IfState      Duration I/F[State]
```

```
R2#show ipv6 ospf6 neighbor
Neighbor ID      Pri    DeadTime      State/IfState      Duration I/F[State]
```

Соседства отсутствуют.

Проверьте настройки OSPFv3:

```
R1#show ipv6 ospf6
OSPFv3 Routing Process (0) with Router-ID 0.0.0.0
. . .
```

```
R2#show ipv6 ospf6
OSPFv3 Routing Process (0) with Router-ID 0.0.0.0
. . .
```

В первой же строке указан одинаковый идентификатор маршрутизатора 0.0.0.0. Это значит что идентификатор просто не задан. Для OSPFv3 требуется идентификатор маршрутизатора в формате IPv4-адреса, и его нужно настроить самостоятельно:

```
R1(config)#router ospf6
R1(config-router)#ospf6 router-id 1.1.1.1
R1(config-router)#end
R1#clear ipv6 ospf6 process
```

```
R2(config)#router ospf6
R2(config-router)#ospf6 router-id 2.2.2.2
R2#R1(config-router)#end
R2#clear ipv6 ospf6 process
```

Обратите внимание на последнюю команду, она нужна для перезапуска сессии OSPFv3 чтобы процесс установления соседства начался заново.

Проверьте соседство ещё раз:

```
R1#show ipv6 ospf6 neighbor
Neighbor ID      Pri   DeadTime      State/IfState      Duration I/F[State]
2.2.2.2          1     00:00:36      Full/BDR           00:05:02 2ge0[DR]
```

```
R2#show ipv6 ospf6 neighbor
Neighbor ID      Pri   DeadTime      State/IfState      Duration I/F[State]
1.1.1.1          1     00:00:30      Full/DR            00:05:09 2ge0[BDR]
```

Соседство установлено!

33.2.8 Конфигурация MP-BGP

Обычная версия протокола BGP (Border Gateway Protocol) поддерживает только IPv4 адресацию. В настоящее время используется MP-BGP (Multiprotocol BGP), который поддерживает различные адреса:

- IPv4 Unicast
- IPv4 Multicast
- IPv6 Unicast
- IPv6 Multicast

MP-BGP также используется для MPLS VPN, где MP-BGP используется для обмена метками VPN. Для каждого типа адреса MP-BGP использует своё семейство адресов.

Чтобы стало возможным использовать новую IPv6 адресацию в MP-BGP реализованы некоторые новые функции, которые отсутствуют в BGP:

- Идентификатор семейства адресов (AFI): определяет семейство адресов (например, IPv4, IPv6).
- Идентификатор последующего семейства адресов (SAFI): содержит дополнительную информацию для различных семейств адресов (определяет Unicast или Multicast в данном случае).
- Информация о доступности сетевого уровня для недоступной многопротокольной сети (MP_UNREACH_NLRI): этот атрибут используется для передачи данных по недоступным сетям.
- Объявление возможностей BGP: используется маршрутизатором BGP для объявления другому маршрутизатору BGP о поддерживаемых им возможностях. MP-BGP и BGP-4 совместимы, поэтому маршрутизатор BGP-4 может игнорировать непонятные ему сообщения.

Поскольку MP-BGP поддерживает IPv4 и IPv6, у нас есть несколько вариантов. Маршрутизаторы MP-BGP могут стать соседями, используя адреса IPv4 и обмениваться префиксами IPv6, или наоборот. Ниже мы рассмотрим несколько примеров конфигурации...

33.2.8.1 MP-BGP с IPv6 соседством и IPv6 префиксами

Постройте схему в соответствии с рисунком ниже.

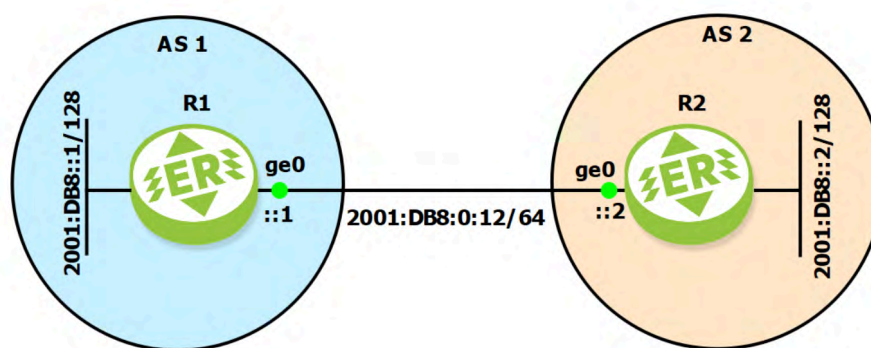


Рисунок 83

На обоих маршрутизаторах запустите маршрутизацию IPv6 и перезагрузитесь:

```
end
enable
configure
```

```
ipv6 unicast-routing
end
write
reload
```

Настройте R1:

```
enable
configure
hostname R1

#Настройка интерфейса
interface 2ge0
ipv6 address 2001:DB8:0:12::1/64
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0

#Настройка loopback интерфейса
interface loopback.0
ipv6 address 2001:DB8::1/128
exit
```

Настройте R2:

```
enable
configure
hostname R2

#Настройка интерфейса
interface 2ge0
ipv6 address 2001:DB8:0:12::2/64
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0

#Настройка loopback интерфейса
interface loopback.0
ipv6 address 2001:DB8::2/128
exit
```

Теперь настройте протокол BGP.

На R1:



```
router bgp 1
bgp router-id 1.1.1.1
neighbor 2001:db8:0:12::2 remote-as 2
address-family ipv4
no neighbor 2001:db8:0:12::2 activate
exit
address-family ipv6
neighbor 2001:db8:0:12::2 activate
network 2001:db8::1/128
end
```

В приведённой выше конфигурации сначала, в обязательном порядке, указывается идентификатор маршрутизатора в формате адреса IPv4.

Затем указывается адрес удалённого соседа. Команда `address-family` используется для изменения настроек IPv4 или IPv6. Далее происходит отключение `address-family` IPv4 и включение IPv6. В конце — объявляется префикс на `loopback`-интерфейсе.

Конфигурация маршрутизатора R2 выглядит аналогично:

```
router bgp 2
bgp router-id 2.2.2.2
neighbor 2001:db8:0:12::1 remote-as 1
address-family ipv4
no neighbor 2001:db8:0:12::1 activate
exit
address-family ipv6
neighbor 2001:db8:0:12::1 activate
network 2001:db8::2/128
end
```

Если всё сделано верно, вы получите на обоих маршрутизаторах сообщения об установлении соседства:

```
2025-07-09 14:40:30      INFO    BGP Neighbor 2001:db8:0:12::2 state UP
2025-07-09 14:40:30      INFO    BGP Neighbor 2001:db8:0:12::1 state UP
```

Тем не менее, проверьте наличие установленных соседств командами:

```
R1#show ipv6 route bgp
IP Route Table for VRF "default"
B      2001:db8::2/128 [20/0] via fe80::1e87:76ff:fe40:201, 2ge0, 00:01:49
```

```
R2#show ipv6 route bgp
IP Route Table for VRF "default"
B      2001:db8::1/128 [20/0] via fe80::1e87:76ff:fe40:101, 2ge0, 00:03:03
```



Вы также можете дать команду `ping` до противоположного loopback-адреса чтобы окончательно убедиться в правильной работе построенной схемы.

33.2.8.2 MP-BGP с IPv4 соседством и IPv6 префиксами

Рассмотрим более сложный пример. Маршрутизаторы станут соседями по IPv4, но будут обмениваться префиксами IPv6.

Немного измените топологию. Теперь маршрутизаторы организуют соседство через IPv4 но при этом обмениваются IPv6 префиксами.

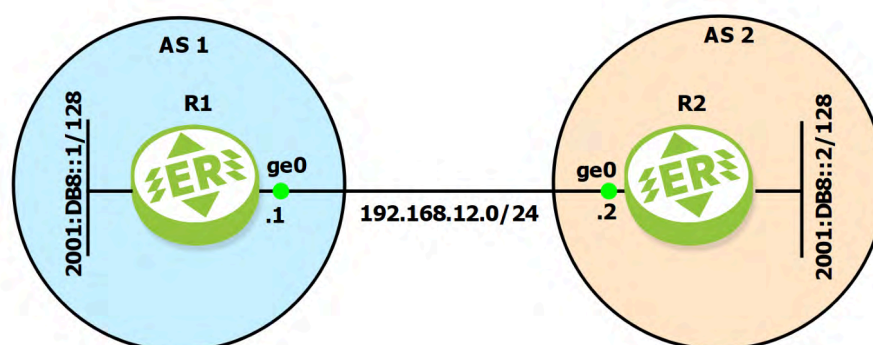


Рисунок 84

Перенастройте R1:

```
end
enable
configure

route-map IPV6_NEXT_HOP permit 10
set ipv6 next-hop 2001:DB8:0:12::2

interface 2ge0
ip address 192.168.12.1/24

router bgp 1
no neighbor 2001:db8:0:12::2 remote-as 2
neighbor 192.168.12.2 remote-as 2
address-family ipv6
neighbor 192.168.12.2 activate
neighbor 192.168.12.2 route-map IPV6_NEXT_HOP in
```

Аналогично перенастройте R2:

```
end
enable
```

```
configure

route-map IPV6_NEXT_HOP permit 10
set ipv6 next-hop 2001:DB8:0:12::1

interface 2ge0
ip address 192.168.12.2/24

router bgp 2
no neighbor 2001:db8:0:12::1 remote-as 1
neighbor 192.168.12.1 remote-as 1
address-family ipv6
neighbor 192.168.12.1 activate
neighbor 192.168.12.1 route-map IPV6_NEXT_HOP in
```

Чтобы заново установить соседства может потребоваться дать команду `clear ip bgp *`.

Были произведены следующие изменения:

- Добавлена маршрутная карта без условий (match) и в ней задан IPv6 адрес следующего перехода. После чего появится возможность использовать IPv4 для пиринга с соседями, чтобы следующий переход не менялся автоматически.
- К интерфейсу 2ge0 добавлены адреса 192.168.12.1(2)/24. Теперь на интерфейсе присутствуют как IPv6 так и IPv4 адреса.
- В настройках маршрутизатора BGP удалены предыдущие настройки соседства IPv6 и вместе с ним настройки `address-family ipv4`.
- Заданы новые соседства IPv4.
- В настройках `address-family ipv6` настроен адрес IPv4 и привязана карта маршрутов задающая IPv6 адрес.

Проверьте результаты:

```
R1#show bgp ipv6 | include 2001
*> 2001:db8::1/128  ::                0          100          32768      i
*> 2001:db8::2/128  2001:db8:0:12::2  1e87:76ff:fe40:201
```

```
R2#show bgp ipv6 | include 2001
*> 2001:db8::1/128  2001:db8:0:12::1  1e87:76ff:fe40:101
*> 2001:db8::2/128  ::                0          100          32768      i
```

IPv6-адреса следующего перехода теперь доступны, и их стало возможно добавить в таблицу маршрутизации.



Недостатком этого решения является то, что приходится самостоятельно определять следующий переход. Преимущество же заключается в том, что присутствует единое соседство BGP, которое можно использовать для обмена префиксами IPv4 и IPv6.

33.2.9 Фильтрация маршрутов BGP IPv6

Фильтрация IPv6 в BGP похожа на IPv4 фильтрацию. Есть три метода, которые можно использовать:

- Prefix-list
- Filter-list
- Карты маршрутов Route-map

Постройте схему в соответствии с рисунком ниже.

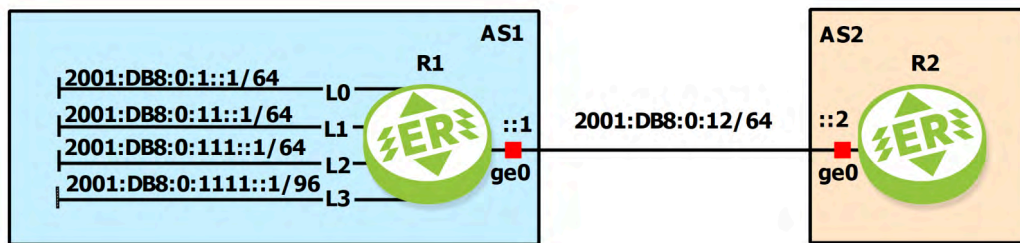


Рисунок 85

Настройте интерфейсы и протокол BGP на R1:

```
end
enable
configure
hostname R1

ipv6 unicast-routing

#Настройка интерфейса
interface 2ge0
ipv6 address 2001:DB8:0:12::1/64
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0

#Настройка loopback интерфейсов
interface loopback.0
```

```
ipv6 address 2001:DB8:0:1::1/64
interface loopback.1
ipv6 address 2001:DB8:0:11::1/64
interface loopback.2
ipv6 address 2001:DB8:0:111::1/64
interface loopback.3
ipv6 address 2001:DB8:0:1111::1/96

#Настройка BGP маршрутизатора
router bgp 1
bgp router-id 1.1.1.1
neighbor 2001:db8:0:12::2 remote-as 2
address-family ipv6
neighbor 2001:db8:0:12::2 activate
network 2001:db8:0:1::/64
network 2001:db8:0:11::/64
network 2001:db8:0:111::/64
network 2001:db8:0:1111::/96
```

Настройте интерфейсы и протокол BGP на R2:

```
end
enable
configure
hostname R2

ipv6 unicast-routing

#Настройка интерфейса
interface 2ge0
ipv6 address 2001:DB8:0:12::2/64
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0

#Настройка BGP маршрутизатора
router bgp 2
bgp router-id 2.2.2.2
neighbor 2001:db8:0:12::1 remote-as 1
address-family ipv6
neighbor 2001:db8:0:12::1 activate
```

Проверьте изучил ли R2 все подсети:




```
R2#show ipv6 route bgp | begin 2001
B 2001:db8:0:1::/64 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:00:04
B 2001:db8:0:11::/64 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:00:04
B 2001:db8:0:111::/64 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:00:04
B 2001:db8:0:1111::/96 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:00:04
```

Отлично, всё что нужно в таблице маршрутизации. Теперь можно поэкспериментировать с фильтрацией.

33.2.9.1 Фильтрация по спискам префиксов

Начнём со списков префиксов. R1 объявляет одну подсеть с длиной префикса /96. Посмотрим, возможно ли настроить R2 для фильтрации этой сети:

```
R2(config)#ipv6 prefix-list SMALL_NETWORKS permit 2001::/16 le 64
```

Этот список префиксов просматривает весь диапазон 2001::/16 и разрешает все подсети с префиксом /64 бит или меньше. Все сети с большей длиной префикса, будут запрещены. Активируйте префиксный список:

```
R2(config)#router bgp 2
R2(config-router)#address-family ipv6
R2(config-router-af)#neighbor 2001:db8:0:12::1 prefix-list SMALL_NETWORKS in
```

Мы активируем входящий префикс-лист на маршрутизаторе R2 для всего, что получаем от маршрутизатора R1.

Перезапустите процесс BGP командой `clear ip bgp *`.

Проверьте R2 исчез ли префикс:

```
R2#show ipv6 route bgp
IP Route Table for VRF "default"
B 2001:db8:0:1::/64 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:02:00
B 2001:db8:0:11::/64 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:02:00
B 2001:db8:0:111::/64 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:02:00
```

Здорово, он успешно отфильтрован!

33.2.9.2 Фильтрация по спискам фильтрации

Настройте список фильтрации (filter-lists) и используйте его для фильтрации префиксов из определённых автономных систем. У всех префиксов которые объявляет R1, имеет только AS 1 в пути автономных систем. Мы настроим добавление автономной системы, чтобы у нас было с чем поиграться:



```
R1(config)#ipv6 prefix-list FIRST_LOOPBACK permit 2001:db8:0:1::/64

R1(config)#route-map PREPEND permit 10
R1(config-route-map)#match ipv6 address prefix-list FIRST_LOOPBACK
R1(config-route-map)#set as-path prepend 11
R1(config)#route-map PREPEND permit 20

R1(config)#router bgp 1
R1(config-router)#address-family ipv6
R1(config-router-af)#neighbor 2001:db8:0:12::2 route-map PREPEND out
```

Конфигурация выше гарантирует, что всякий раз, когда R1 объявляет префикс 2001:db8:0:1::/64, он добавит AS 11 к пути автономных систем. Проверьте это:

```
R2#show bgp ipv6
BGP table version is 3, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best,
i - internal, l - labeled   S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Network	Next Hop	Metric	LocPrf	Weight	Path	
*> 2001:db8:0:1::/64	2001:db8:0:12::1	1e87:76ff:fe40:1	0	100	0	1 11 i
*> 2001:db8:0:11::/64	2001:db8:0:12::1	1e87:76ff:fe40:1	0	100	0	1 i
*> 2001:db8:0:111::/64	2001:db8:0:12::1	1e87:76ff:fe40:1	0	100	0	1 i
*> 2001:db8:0:1111::/96	2001:db8:0:12::1	1e87:76ff:fe40:1	0	100	0	1 i

Total number of prefixes 4

Теперь путь в подсеть 2001:db8:0:1::/64 лежит через автономную систему 11.

Теперь настройте filter-list на R2, чтобы избавиться от этой сети.

```
R2(config)#ip as-path access-list 11 permit ^1$

R2(config)#router bgp 2
R2(config-router)#address-family ipv6
R2(config-router-af)#neighbor 2001:db8:0:12::1 filter-list 11 in
```

```
R2#clear ip bgp *
```

Список доступа as-path, указанный выше, разрешает только префиксы из AS1 и ничего больше. 11 в данном случае это наименование списка доступа, а ^1\$ — регулярное выражение означающее "1", то есть AS 1.

Мы применяем список доступа ко всем входящим пакетам, получаемым от R1.

Вот результат:

```
R2#show ipv6 route bgp
IP Route Table for VRF "default"
B  2001:db8:0:11::/64 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:02:00
B  2001:db8:0:111::/64 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:02:00
```

Сеть пропала из таблицы маршрутизации, миссия выполнена.

33.2.9.3 Фильтрация по картам маршрутов

Карты маршрутов (route-maps) крайне полезны и могут быть использованы для сопоставления множества различных вещей. Мы можем использовать список доступа IPv6 в карте маршрутов, чтобы отфильтровать подсеть 2001:DB8:0:11::/64:

```
R2(config)#ipv6 prefix-list THIRD_LOOPBACK permit 2001:db8:0:11::/64

R2(config)#route-map MY_FILTER deny 10
R2(config-route-map)#match ipv6 address prefix-list THIRD_LOOPBACK
R2(config-route-map)#exit
R2(config)#route-map MY_FILTER permit 20

R2(config)#router bgp 2
R2(config-router)#address-family ipv6
R2(config-router-af)#neighbor 2001:db8:0:12::1 route-map MY_FILTER in

R2#clear ip bgp *
```

В конфигурации выше задан префикс-лист под названием "THIRD_LOOPBACK" который совпадает с префиксом 2001:db8:0:11::/64, а далее он будет отклонён в маршрутной карте "MY_FILTER". Последний, но важный момент — следует применить "MY_FILTER" в качестве входного фильтра на R2.

Вот результат:

```
R2#show ipv6 prefix-list
ipv6 prefix-list THIRD_LOOPBACK: 1 entries
seq 5 permit 2001:db8:0:11::/64
```



```
R2#show ipv6 route bgp
IP Route Table for VRF "default"
B 2001:db8:0:111::/64 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:10:47
```

Prefix-list сообщает нам, что совпадение есть, и вы можете видеть, что маршрут исчез из таблицы маршрутизации.

33.2.9.4 Порядок операций

Разобравшись, как работают prefix-list и route-map для нужд фильтрации префиксов IPv6, вы можете применять всё это одновременно. Взгляните на конфигурацию R2:

```
R2#show run bgp
!
bgp extended-asn-cap
!
router bgp 2
  bgp router-id 2.2.2.2
  neighbor 2001:db8:0:12::1 remote-as 1
  address-family ipv6 unicast
  neighbor 2001:db8:0:12::1 activate
  neighbor 2001:db8:0:12::1 prefix-list SMALL_NETWORKS in
  neighbor 2001:db8:0:12::1 route-map MY_FILTER in
  neighbor 2001:db8:0:12::1 filter-list 11 in
  exit-address-family
!
```

В рабочей сети вы, вероятно, не будете использовать все эти методы одновременно. Карта маршрутов — популярный инструмент, поскольку её можно использовать практически для любых целей, например, для фильтрации и добавления пути автономной системы.

Если вы активируете все эти методы одновременно, то в каком порядке маршрутизатор будет обрабатывать эти методы фильтрации? Порядок следующий:

Входящие:

- Route-map
- Filter-List
- Prefix-List

Исходящие:

- Prefix-List

- Filter-List
- Route-Map

Почему это важно? Представьте, что у вас есть входящая карта маршрутов и список префиксов. Если вы разрешили префикс в списке префиксов, но запретили его в карте маршрутов, то вы никогда не увидите его в таблице BGP, поскольку карта маршрутов обрабатывается раньше списка префиксов.

Для исходящей фильтрации всё наоборот. Если вы разрешаете что-то в карте маршрутов, но запрещаете это в списке фильтров, то это никогда не будет объявлено... список фильтров обрабатывается раньше карты маршрутов для исходящих обновлений.

Не усложняйте себе задачу, лучше использовать только карту маршрутов, поскольку к ней можно прикрепить списки префиксов и списки доступа as-path.

33.2.10 Суммаризация маршрутов BGP IPv6

Постройте схему в соответствии с рисунком ниже.

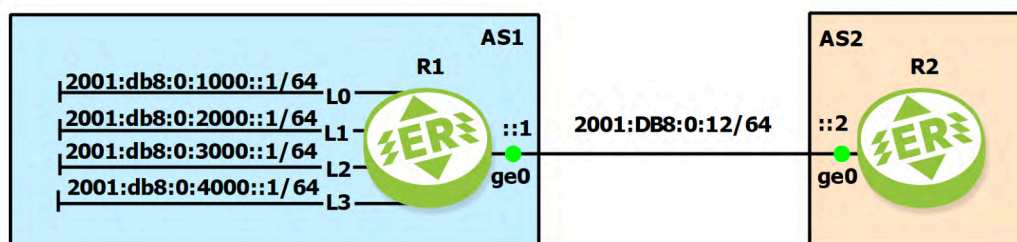


Рисунок 86

Настройте интерфейсы и протокол BGP на R1:

```
end
enable
configure
ipv6 unicast-routing
write
end
reload
```

```
enable
configure
hostname R1

#Настройка интерфейса
interface 2ge0
ipv6 address 2001:DB8:0:12::1/64
```

```
port ge0
service-instance 4ge0
encapsulation untagged
connect ip interface 2ge0

#Настройка loopback интерфейсов
interface loopback.0
ipv6 address 2001:db8:0:1000::1/64
interface loopback.1
ipv6 address 2001:db8:0:2000::1/64
interface loopback.2
ipv6 address 2001:db8:0:3000::1/64
interface loopback.3
ipv6 address 2001:db8:0:4000::1/64

#Настройка BGP маршрутизатора
router bgp 1
bgp router-id 1.1.1.1
neighbor 2001:db8:0:12::2 remote-as 2
address-family ipv6
neighbor 2001:db8:0:12::2 activate
network 2001:db8:0:1000::/64
network 2001:db8:0:2000::/64
network 2001:db8:0:3000::/64
network 2001:db8:0:4000::/64
```

Настройте интерфейсы и протокол BGP на R2:

```
end
enable
configure
ipv6 unicast-routing
write
end
reload
```

```
enable
configure
hostname R2

#Настройка интерфейса
interface 2ge0
ipv6 address 2001:DB8:0:12::2/64
port ge0
service-instance 4ge0
```

```
encapsulation untagged
connect ip interface 2ge0

#Настройка BGP маршрутизатора
router bgp 2
bgp router-id 2.2.2.2
neighbor 2001:db8:0:12::1 remote-as 1
address-family ipv6
neighbor 2001:db8:0:12::1 activate
```

Проверьте изучил ли R2 все подсети:

```
R2#show ipv6 route bgp
IP Route Table for VRF "default"
B 2001:db8:0:1000::/64 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:00:02
B 2001:db8:0:2000::/64 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:00:02
B 2001:db8:0:3000::/64 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:00:02
B 2001:db8:0:4000::/64 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:00:02
```

Все четыре подсети созданные на R1 появились в таблице.

Настройте на R1 суммаризацию:

```
R1(config)#router bgp 1
R1(config-router)#address-family ipv6
R1(config-router-af)#aggregate-address 2001:db8::/50 summary-only
```

Перезапустите соседство BGP командой `clear bgp ipv6 *`.

Проверьте, какие теперь подсети доступны на R2:

```
R2#show ipv6 route bgp
IP Route Table for VRF "default"
B 2001:db8::/50 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:11:27
B 2001:db8:0:4000::/64 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:11:27
```

Как видите, заданные на R1 префиксы 2001:db8:0:1000::/64, 2001:db8:0:2000::/64 и 2001:db8:0:3000::/64 были суммаризованы в маршрут 2001:db8::/50. Маршрут 2001:db8:0:4000::/64 не был суммаризован, так как не подпадает под длину префикса в 50.

Поменяйте на R1 длину префикса с 50 на 49:

```
R1(config)#router bgp 1
R1(config-router)#address-family ipv6
R1(config-router-af)#no aggregate-address 2001:db8::/50 summary-only
R1(config-router-af)#aggregate-address 2001:db8::/49 summary-only
```


Перезапустите соседство BGP командой `clear bgp ipv6 *`.

Снова проверьте R2:

```
R2#show ipv6 route bgp
IP Route Table for VRF "default"
B  2001:db8::/49 [20/0] via fe80::1e87:76ff:fe40:1, 2ge0, 00:00:18
```

В таблице остался лишь один суммаризированный адрес.

При этом с R2 по команде `ping` по прежнему доступны все loopback-адреса на R1.

33.2.11 Перераспределение маршрутов IPv6

Перераспределение (редистрибуция) маршрутов IPv6 во многом похоже на IPv4, здесь действуют те же правила и возникают те же проблемы. Однако есть одна проблема, которая касается только перераспределения IPv6, и она будет раскрыта в этом пункте.

Постройте схему в соответствии с рисунком ниже.

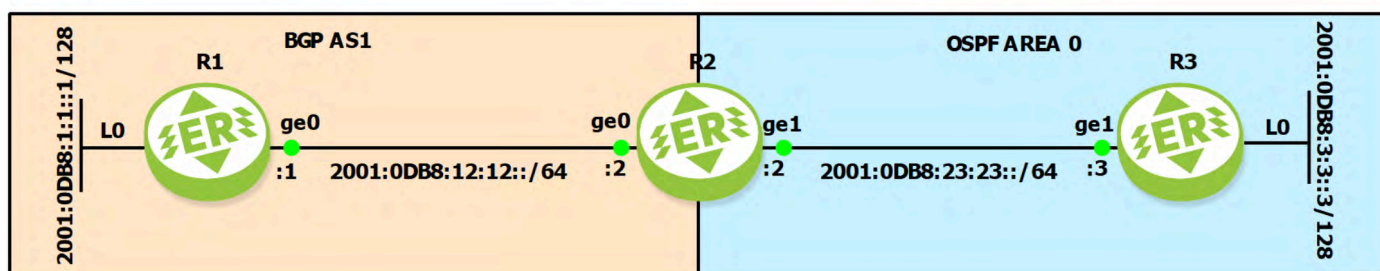


Рисунок 87

На R1 и R2 настроен протокол BGP, на R2 и R3 протокол OSPFv3. R2 перераспределяет маршруты между двумя протоколами. Проблема в том, что R1 не может достичь подсети 2001:0DB8:23:23::/64 а R3 не может достичь 2001:0DB8:12:12::/64. Все остальные подсети доступны.

Задайте имя и включите IPv6 маршрутизацию на R1 и аналогично на R2 и R3, после чего перезагрузите маршрутизаторы.

```
end
enable
configure
hostname R1

#Включение работы с IPv6 и перезагрузка
ipv6 unicast-routing
write
```

```
end  
reload
```

Сконфигурируйте порты и интерфейсы стандартным образом используя данные на схеме.

Ha R1.

```
end  
enable  
configure  
  
interface 2ge0  
ipv6 address 2001:0DB8:12:12::1/64  
port ge0  
service-instance 4ge0  
encapsulation untagged  
connect ip interface 2ge0  
  
interface loopback.0  
ipv6 address 2001:0DB8:1:1::1/128
```

Ha R2

```
end  
enable  
configure  
  
interface 2ge0  
ipv6 address 2001:0DB8:12:12::2/64  
port ge0  
service-instance 4ge0  
encapsulation untagged  
connect ip interface 2ge0  
  
interface 2ge1  
ipv6 address 2001:0DB8:23:23::2/64  
port ge1  
service-instance 4ge1  
encapsulation untagged  
connect ip interface 2ge1
```

Ha R3:



```
end
enable
configure

interface 2ge1
ipv6 address 2001:0DB8:23:23::3/64
port ge1
service-instance 4ge1
encapsulation untagged
connect ip interface 2ge1

interface loopback.0
ipv6 address 2001:0DB8:3:3::3/128
```

Настройте на R1 протокол BGP:

- идентификатор маршрутизатора;
- соседство BGP с маршрутизатором R2;
- сеть loopback-адреса.

```
router bgp 1
bgp router-id 1.1.1.1
neighbor 2001:db8:12:12::2 remote-as 1
address-family ipv6 unicast
network 2001:db8:1:1::1/128
neighbor 2001:db8:12:12::2 activate
```

Настройте на R2 протоколы OSPF и BGP:

- идентификатор маршрутизатора для OSPF6;
- идентификатор маршрутизатора для BGP;
- соседство BGP с маршрутизатором R1;
- нулевую зону протокола OSPF6 на интерфейсе 2ge1.

```
router ospf6
ospf6 router-id 2.2.2.2

router bgp 1
bgp router-id 2.2.2.2
neighbor 2001:db8:12:12::1 remote-as 1
address-family ipv6 unicast
neighbor 2001:db8:12:12::1 activate
```

```
interface 2ge1
ipv6 ospf6 area 0
```

Настройте на R3 протокол OSPF:

- идентификатор маршрутизатора OSPF6;
- нулевую зону протокола OSPF6 на интерфейсе 2ge1;
- нулевую зону протокола OSPF6 на интерфейсе loopback.0.

```
router ospf6
ospf6 router-id 3.3.3.3
```

```
interface 2ge1
ipv6 ospf6 area 0
```

```
interface loopback.0
ipv6 ospf6 area 0
```

Теперь с помощью команды **ping** вы можете удостовериться, что с маршрутизатора R2 доступны как маршрутизаторы R1 и R3 так и loopback-адреса 2001:0DB8:1:1::1 и 2001:0DB8:3:3::3. Т.е. маршрутизация работает внутри каждого домена маршрутизации OSPF6 и BGP. Соединение же между R1 и R3 отсутствует.

Чтобы получить возможность передавать маршруты и данные между R1 и R3, нужно прибегнуть к редистрибуции (перераспределению) маршрутов. На R2 настройте редистрибуцию маршрутов из BGP в OSPF6 и из OSPF6 в BGP — введите следующие команды:

```
router ospf6
redistribute bgp metric 100
router bgp 1
address-family ipv6
redistribute ospf
```

Рассмотрим таблицы маршрутизации:

```
R1#show ipv6 route bgp
IP Route Table for VRF "default"
B 2001:db8:3:3::3/128 [200/1] via fe80::1e87:76ff:fe40:201, 2ge0, 00:03:48
```

R1 "знает" об loopback-адресе на R2.

```
R2#show ipv6 route bgp
IP Route Table for VRF "default"
B 2001:db8:1:1::1/128 [200/0] via fe80::1e87:76ff:fe40:101, 2ge0, 00:05:13
```



```
R2#show ipv6 route ospf
IP Route Table for VRF "default"
O 2001:db8:3:3::3/128 [110/1] via fe80::1e87:76ff:fe40:302, 2ge1, 21:06:03
```

R2 "знает" об обоих loopback-адресах.

```
R3#show ipv6 route ospf
IP Route Table for VRF "default"
O 2001:db8:1:1::1/128 [110/100] via fe80::1e87:76ff:fe40:202, 2ge1
```

R3 "знает" об loopback-адресе на R1.

Введите команду **ping** на R3:

```
R3#ping 2001:db8:3:3::3
PING 2001:db8:3:3::3 3 56 data bytes
64 bytes from 2001:db8:3:3::3: icmp_seq=1 ttl=64 time=0.970 ms
64 bytes from 2001:db8:3:3::3: icmp_seq=2 ttl=64 time=2.47 ms
```

Обмен данными проходит успешно.

Введите команду **ping** на R1:

```
R1#ping 2001:db8:3:3::3
PING 2001:db8:3:3::3 3 56 data bytes

--- 2001:db8:3:3::3 ping statistics ---
7 packets transmitted, 0 received, 100% packet loss, time 6151ms
```

Обмен данными не происходит.

Введите на R2 команду:

```
R2(config)#router ospf6
R2(config-router)#redistribute connected
```

redistribute connected позволяет включать directly connected сети в процесс маршрутизации. Теперь **ping** работает в обе стороны.

34 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 друг от друга и от основного маршрутизатора (Default Router), в котором они созданы.

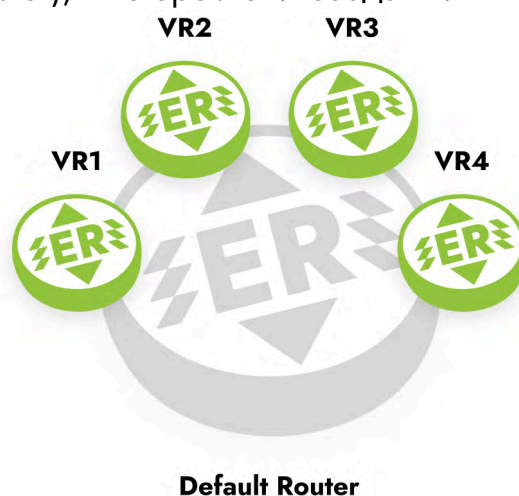


Рисунок 88

34.1 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 121 — Команды режима настройки виртуального маршрутизатора

Команда	Описание
---------	----------

Команда	Описание
<code>bind <INTERFACE_NAME></code>	Привязать интерфейс к виртуальному маршрутизатору. ВНИМАНИЕ. При передаче интерфейса из основного маршрутизатора в виртуальный или обратно все настройки интерфейса сбрасываются
<code>configuration file <имя файла></code>	Создание файла для сохранения конфигурации виртуального маршрутизатора
<code>description <TEXT></code>	Создание комментария к виртуальному маршрутизатору
<code>load {bgp isis ospf pim rip vrrp}</code>	Команда добавления протоколов в виртуальный маршрутизатор: <ul style="list-style-type: none"> - bgp добавить протокол bgpv4, - isis добавить протокол isis, - ospf добавить протокол ospfv2, - pim добавить протокол pimv2, - rip добавить протокол ripv2, - vrrp добавить протокол vrrp

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).

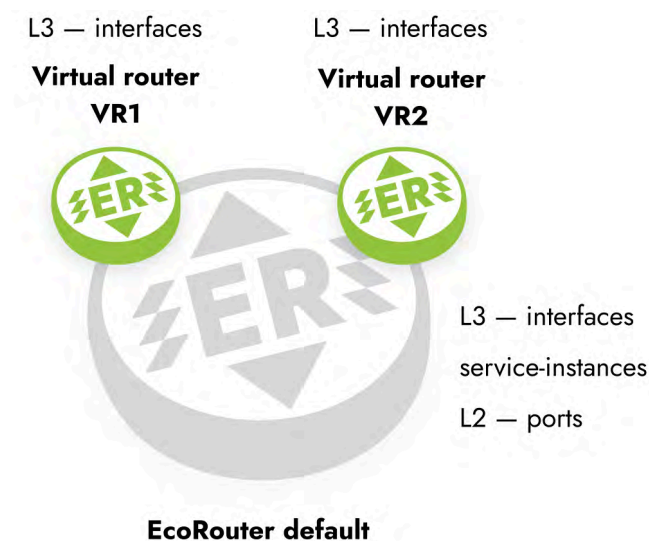


Рисунок 89

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.

34.2 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
```

34.3 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 te1 service-instance 100
virtual-router-forwarding VR10
ip access-group 001 in
!
```

35 Виртуальные машины и контейнеры

35.1 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.

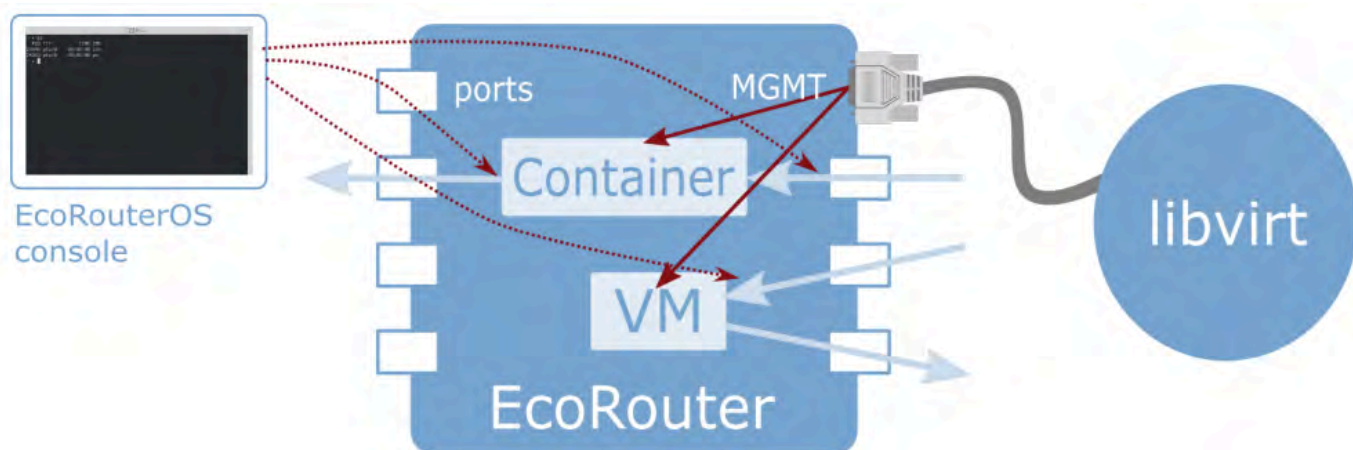


Рисунок 90

Внимание! При использовании сетевых интерфейсов с драйвером **virtio** необходимо

отключить **TCP offload engine**, так как на данный момент существует ошибка при подсчёте контрольной суммы в TCP-заголовке.

Отключить **TCP offload engine** можно следующими способами:

- В ОС на виртуальной машине выполнить следующую команду:

```
ethtool --offload eth0 tx off
```

- 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 удалённому 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
```

35.2 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 маршрутизируемые 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»).

35.3 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**.

35.4 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
```

Таблица 122 — Варианты команды `copy ftp virtual-disk`

Команда	Описание
<code>copy ftp virtual-disk ftp://user:password@xxx.xxx.xxx.xxx/filename mgmt</code>	С FTP-сервера будет скачан указанный файл виртуального диска, FTP-сервер доступен через менеджмент-порт (mgmt)
<code>copy ftp virtual-disk ftp://user:password@xxx.xxx.xxx.xxx/filename vr default</code>	С FTP-сервера будет скачан указанный файл виртуального диска. Доступ к FTP-серверу осуществляется через интерфейс виртуального маршрутизатора, выбранного по умолчанию
<code>copy tftp virtual-disk tftp://xxx.xxx.xxx.xxx/filename vr vrname</code>	С TFTP-сервера будет скачан указанный файл виртуального диска. Доступ к TFTP-серверу осуществляется через интерфейс виртуального маршрутизатора с именем vrname

Команда	Описание
<code>copy tftp virtual-disk tftp://xxx.xxx.xxx.xxx/ filename mgmt</code>	С TFTP-сервера будет скачан указанный файл виртуального диска. Доступ к TFTP-серверу осуществляется через менеджмент-порт (mgmt)

35.5 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 | 10}**, где 0 означает, что ядра не будут выделены; 10 означает, что будет выделено десять ядер.

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`.

35.6 Подключение к виртуальной машине

35.6.1 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
```

35.6.2 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
```



35.6.3 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_ВМ> &
```

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**.

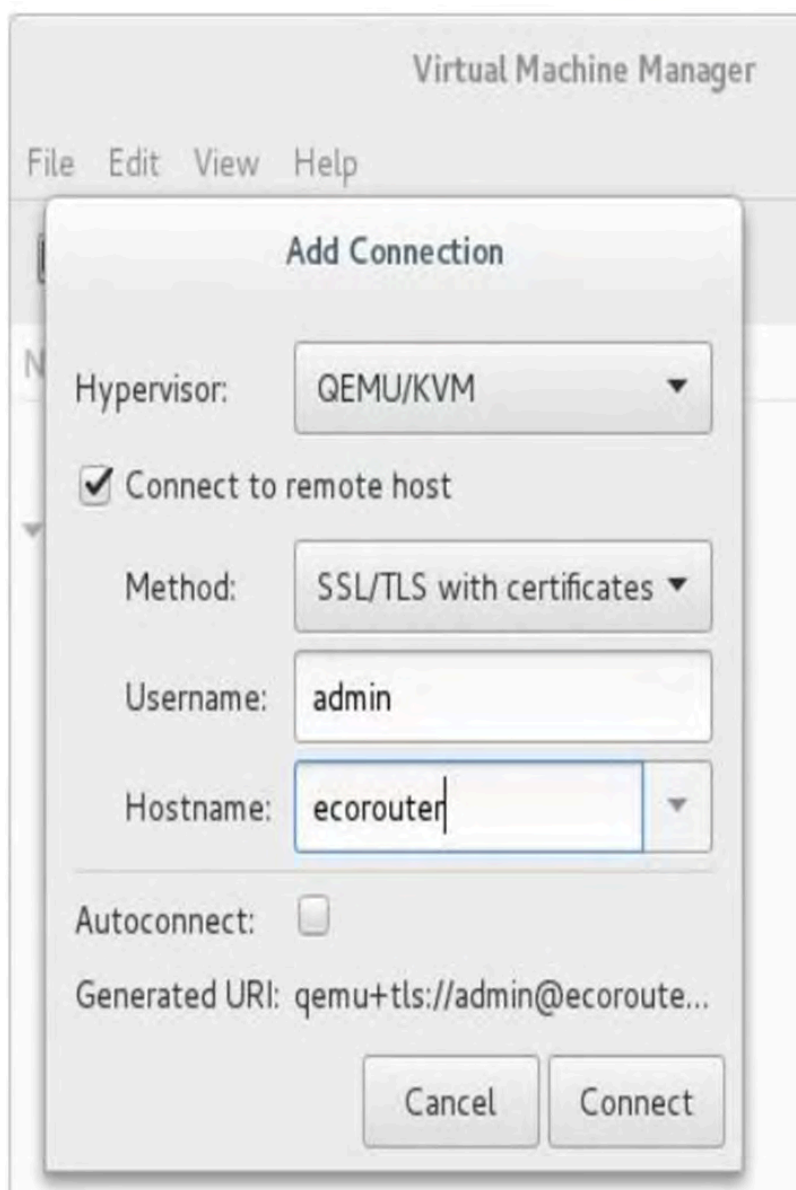


Рисунок 91

35.7 Быстрая настройка виртуальных машин

Для быстрой настройки виртуальных машин в EcoRouter необходимо произвести действия, описанные ниже.

- 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 ресурсоёмкими 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
```

- 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
```

- 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.

- Экспортировать на локальную машину сертификаты пользователя для подключения к libvirt на EcoRouter. Пример экспорта для Linux машин приведён в таблице ниже.

Таблица 123 — Пример экспорта сертификатов пользователя

Вывод команды на EcoRouter	скопировать в файл на локальном компьютере
<code>crypto ca export</code>	<code>/etc/pki/CA/cacert.pem</code>
<code>crypto certificate export</code>	<code>/etc/pki/libvirt/clientcert.pem</code>
<code>crypto key export</code>	<code>/etc/pki/libvirt/private/clientkey.pem</code>

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
```

- Добавить в файл `/etc/hosts` запись об IP-адресе EcoRouter с именем хоста — **ecorouter**.
- 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**.

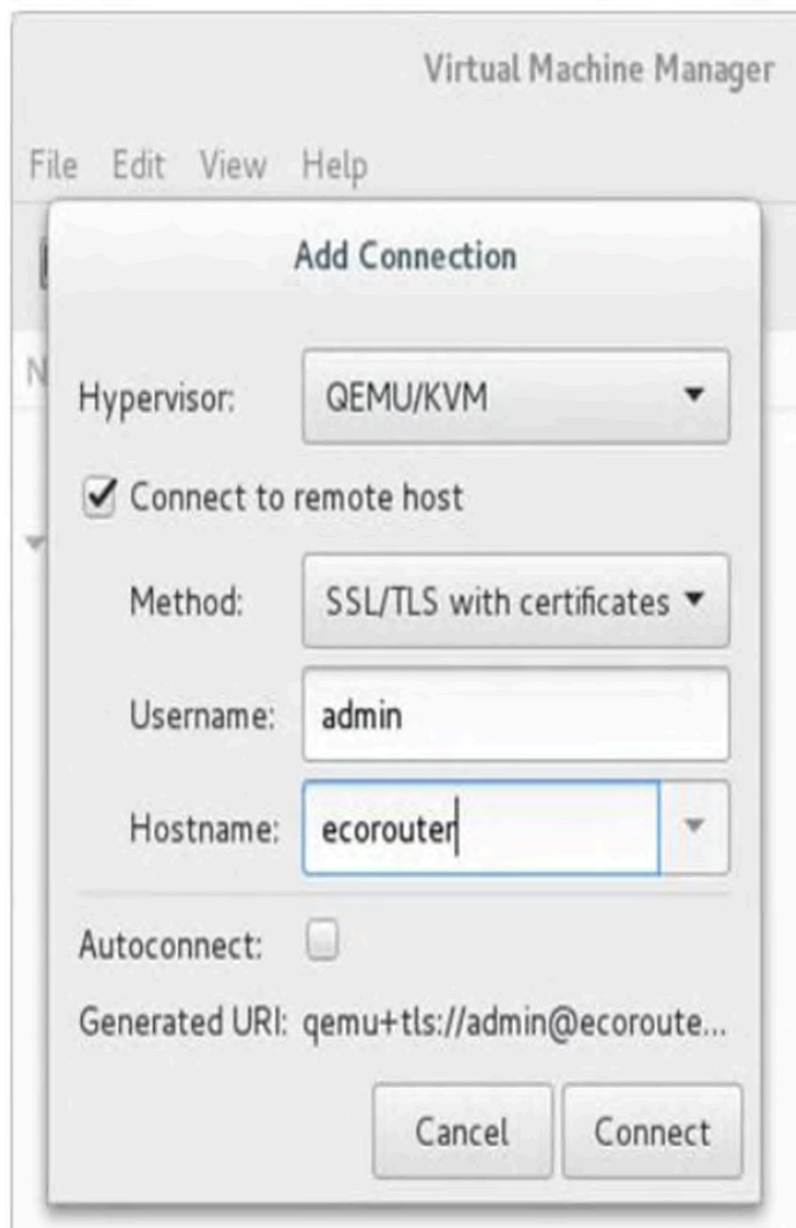


Рисунок 92

1. Создать новую виртуальную машину, используя образ жёсткого диска, скопированный ранее на EcoRouter (см. шаг 2).

- 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).

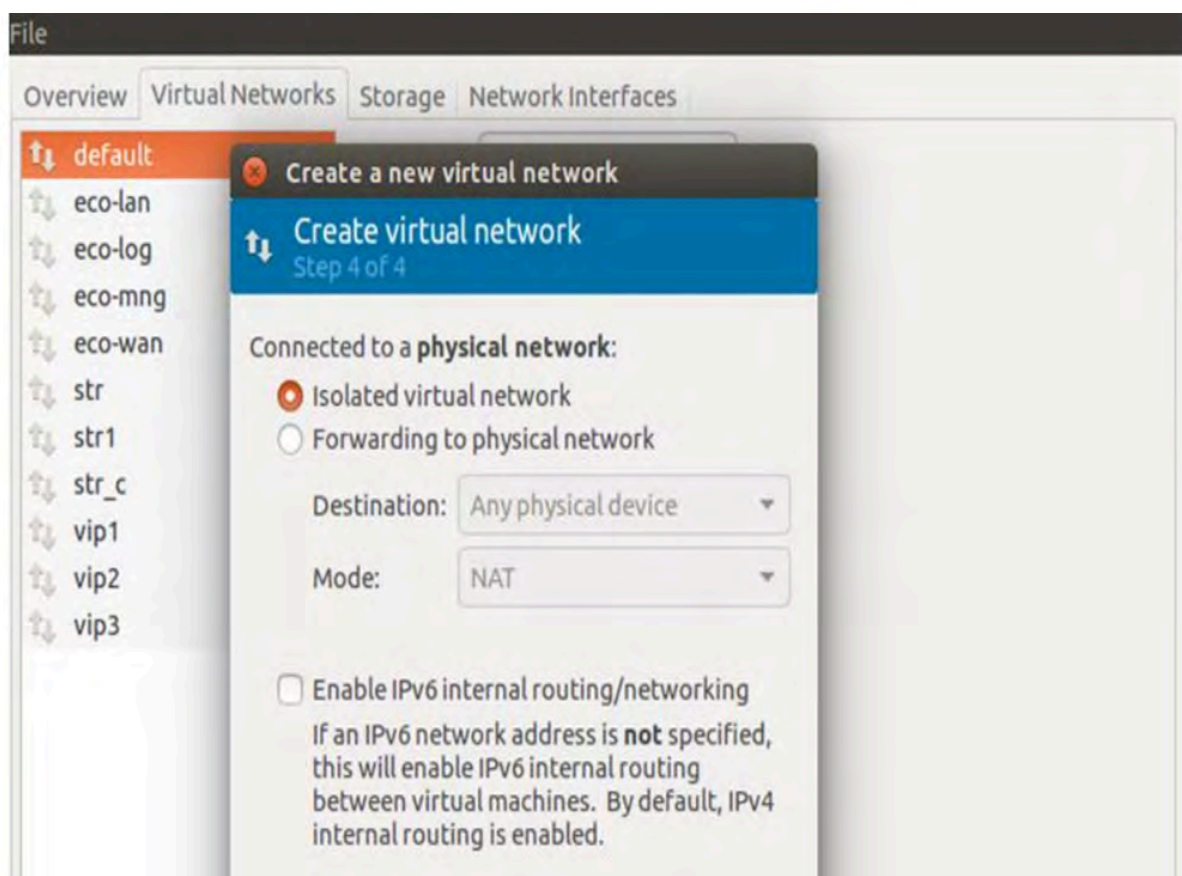


Рисунок 93

1. При необходимости добавить сетевые интерфейсы. Каждый интерфейс подключается к одной из ранее созданных виртуальных сетей.

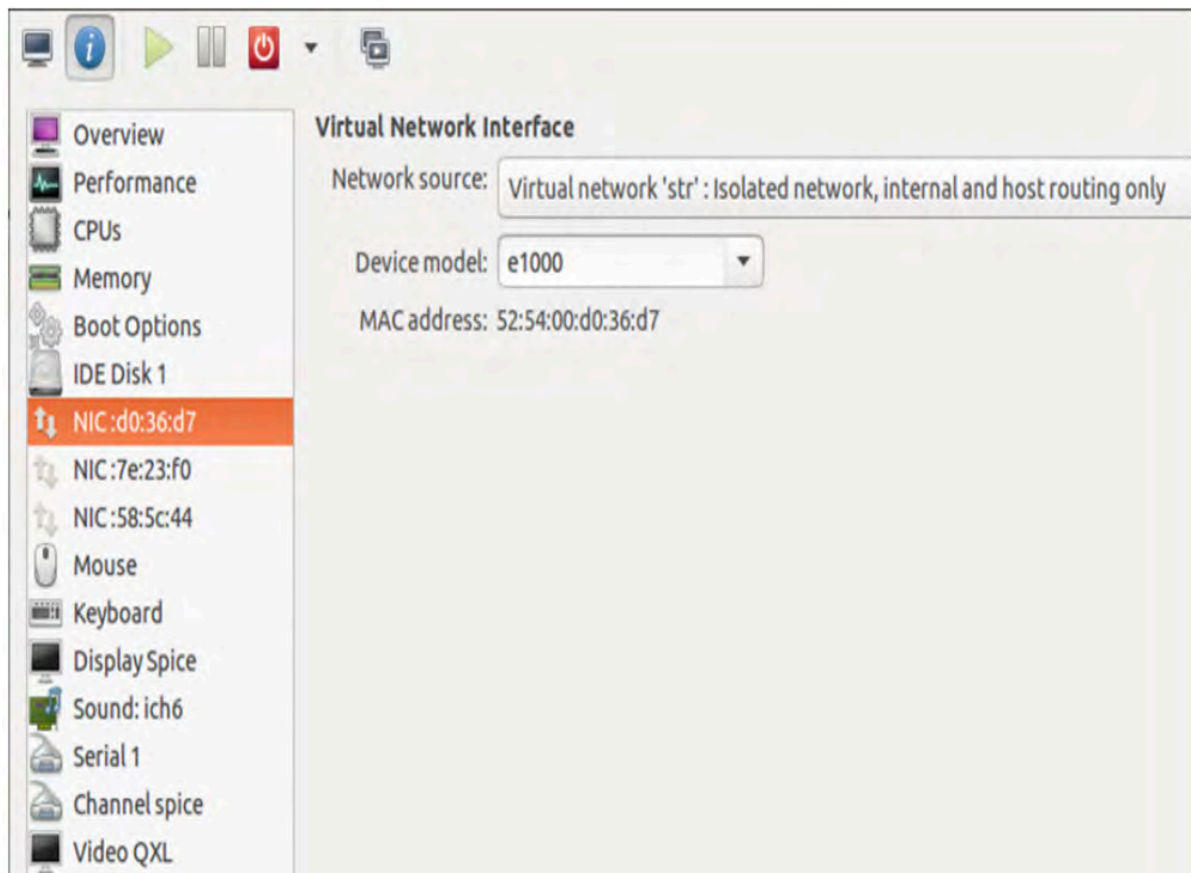


Рисунок 94

1. В пункте **Display Spice** в поле **Address** выбрать **All interfaces**.

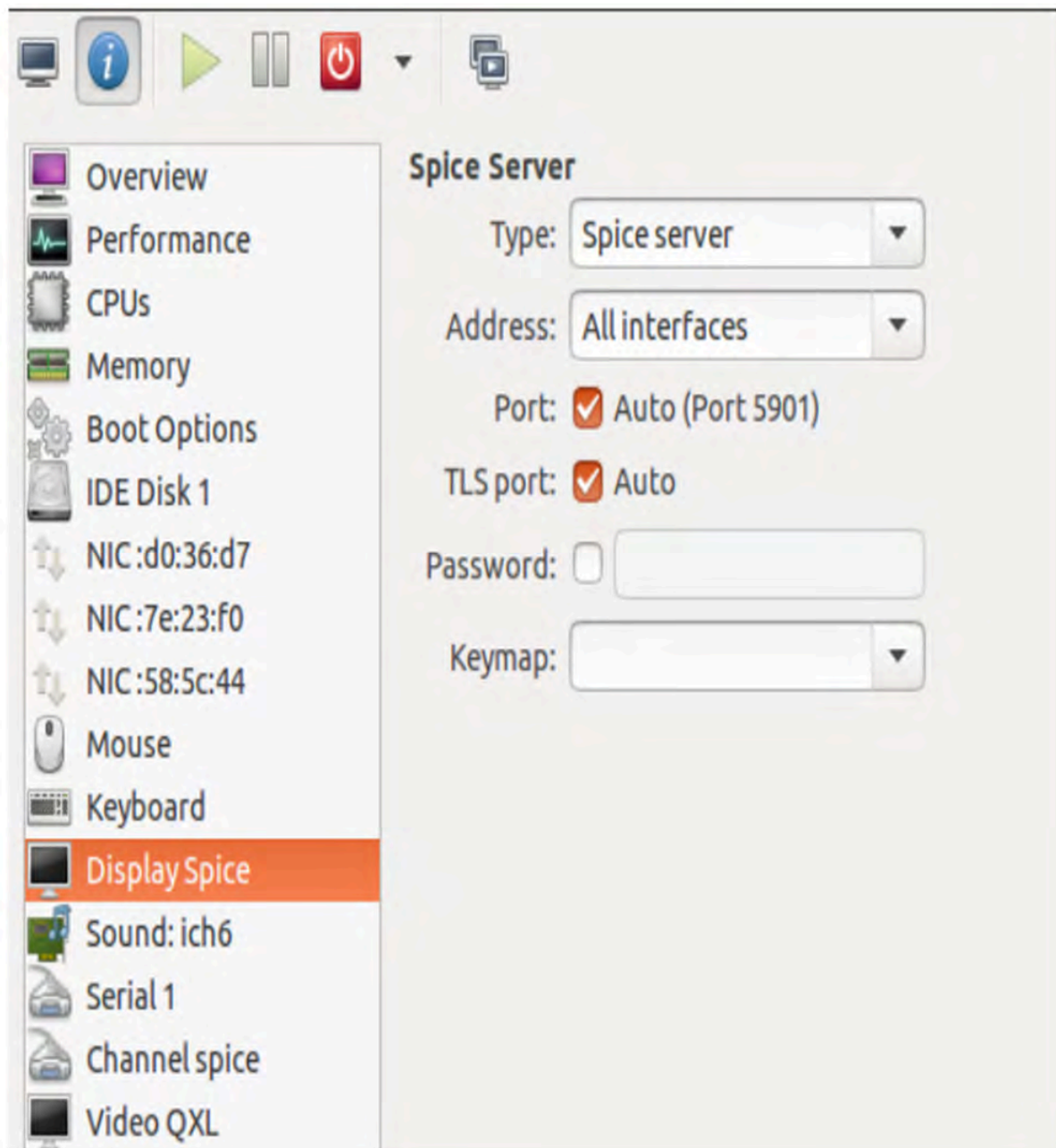
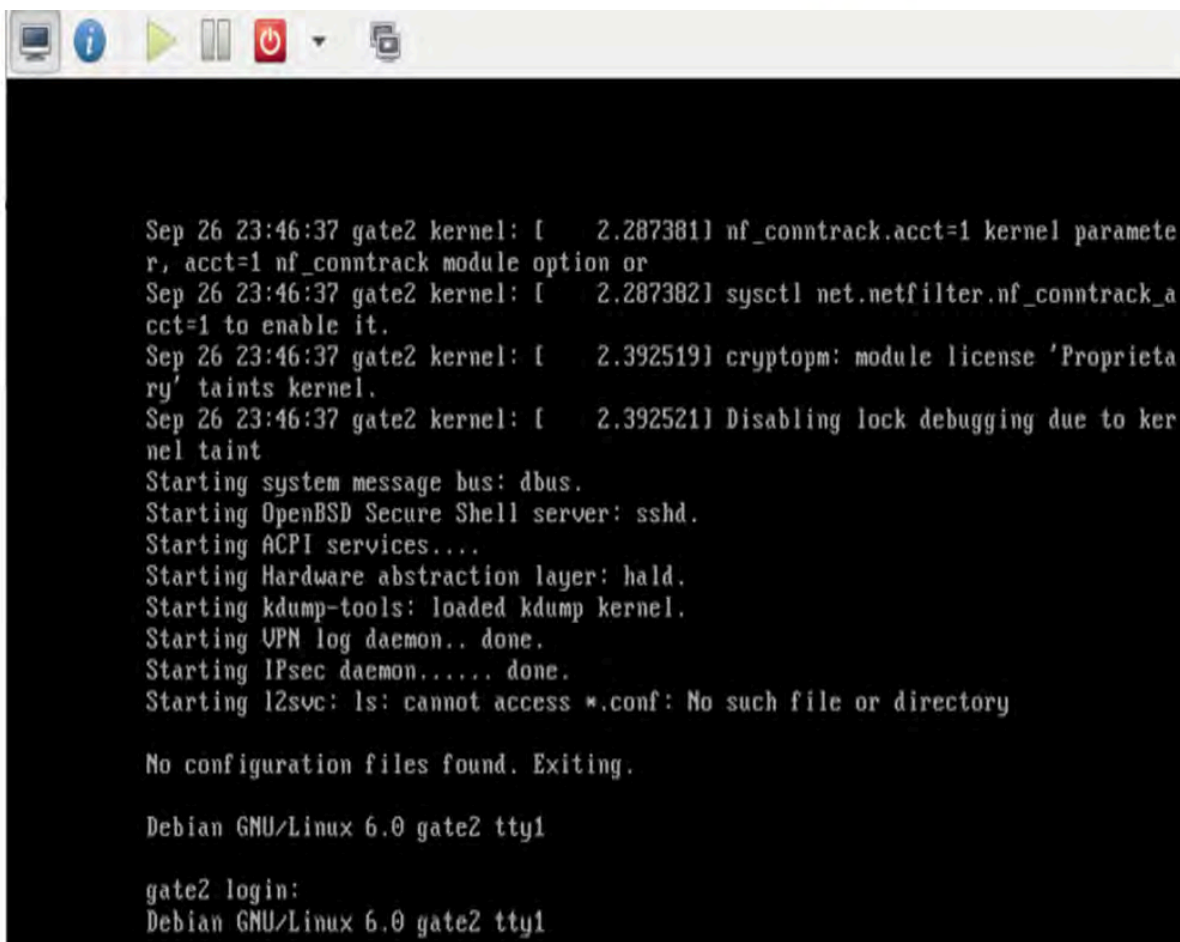


Рисунок 95

1. 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 paramete
r, acct=1 nf_conntrack module option or
Sep 26 23:46:37 gate2 kernel: [ 2.287382] sysctl net.netfilter.nf_conntrack_a
cct=1 to enable it.
Sep 26 23:46:37 gate2 kernel: [ 2.392519] cryptopm: module license 'Proprieta
ry' taints kernel.
Sep 26 23:46:37 gate2 kernel: [ 2.392521] Disabling lock debugging due to ker
nel taint
Starting system message bus: dbus.
Starting OpenBSD Secure Shell server: sshd.
Starting ACPI services....
Starting Hardware abstraction layer: hald.
Starting kdump-tools: loaded kdump kernel.
Starting UPN 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
```

Рисунок 96

1. 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 конфигурирования 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 ge1
```

Конфигурирование внешнего порта EcoRouter.

```
ecorouter#conf t
Enter configuration commands, one per line. End with CNTL/Z.
ecorouter(config)#port ge1
ecorouter(config-port-virt)#service-instance ge1
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 ge1
lACP-priority 32767
mtu 9728
service-instance ge1
  encapsulation untagged
!
...
!
port virt.0
virtual-network vm uplink
service-instance virt0
  encapsulation untagged
!
...
!
flow port ge1 service-instance ge1 port virt.0
!
flow port virt.0 service-instance virt0 port ge1
!
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
```



- Далее все настройки IP-адресации будут производиться на виртуальной машине.

35.8 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 защищённому соединению у пользователя должны храниться: сертификат EcoRouter (CA), сертификат пользователя, закрытый ключ пользователя.

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.

В приведённом 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...l0lBt18=
-----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 приведённом ниже примере сокращён вывод самого ключа. Все ключи хранятся и выводятся на консоль в кодировке Base64. Закрытые ключи должны передаваться на пользовательские компьютеры безопасным образом, исключая 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 приведённом 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:E8:53:0A
      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 экспортировать 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

36 Поток E1

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 передаётся 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, соединённых 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 включён, происходит расчёт контрольной суммы при отправлении и на удалённой 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 обеих сторонах магистрали режимы должны совпадать.

Маршрутизатор использует два типа инкапсуляции в потоке E1: HDLC и PPP. Тип инкапсуляции на обеих сторонах должен совпадать.

36.1 Порты и каналы E1

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 разъёмов — коннекторы 8P8C, известные также как RJ45. На рисунке ниже приведена разводка линий по контактам разъёма.

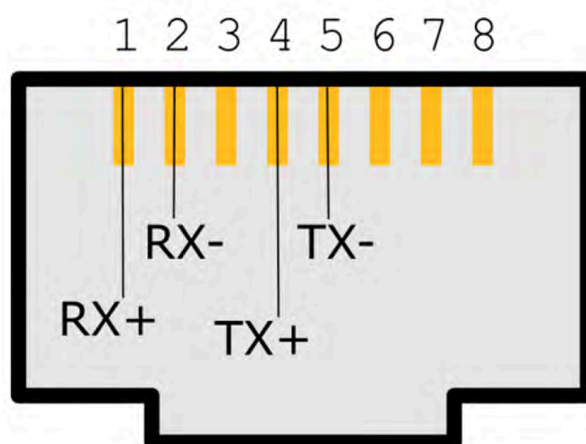


Рисунок 97

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.

36.1.1 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, приведённые в таблице ниже.

Таблица 124 — Команды контекстного режима настройки контроллера

Команда	Описание
<code>clocking {internal remote}</code>	Выбор источника синхронизации: internal — внутренний источник синхронизации, remote — удаленный источник синхронизации
<code>framing {crc4 nocrc4 unframed}</code>	Настройка структуры кадров: crc4 — включен режим CRC-4,

Команда	Описание
	nocrc4 — выключен режим CRC-4, unframed — включен неструктурированный режим
loopback {local remote}	Включение режима петли: local — петля на локальном оборудовании, remote — петля на удалённом 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
```

36.1.2 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 подразделе "Порт" раздела "Виды портов и интерфейсов". Специфичные для портов E1 настройки приведены в таблице ниже. Все они выполняются в контекстном режиме конфигурирования порта.

Таблица 125 — Команды настройки порта E1

Команда	Описание
---------	----------

Команда	Описание
<code>timeslots controller e1.<NUM> (1-31)</code>	Выделение таймслотов с контроллера E1, где NUM — номер контроллера. Для режима <code>unframed</code> диапазон таймслотов не указывается.
<code>service instance <NAME></code>	Задание сервисного интерфейса.
<code>encapsulation untagged</code>	Задание нетегированной инкапсуляции. Обязательная команда.
<code>connect ip interface <NAME></code>	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-instace 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
```

36.1.3 Настройка аутентификации

Для инкапсуляции PPP можно задать аутентификацию для идентификации удалённой стороны. В EcoRouter для аутентификации используется протокол CHAP. Режим аутентификации задаётся контекстной командой настройки порта **ppp** или **mppp** (Multilink ppp). Для порта **mppp** аутентификация конфигурируется на объединённом порте 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 удалённой 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
```

36.2 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
...
```

37 Логирование и отладка

37.1 Локальное логирование

В системе EcoRouter ведётся запись обо всех происходящих событиях (выполняемых операциях, изменениях конфигурации) — логирование. По умолчанию журнал событий (лог) ведётся на самом устройстве.

Сообщения о событиях пишутся в двух форматах, описанных ниже.

Формат системных сообщений — действий, производимых сервисами системы:

```
<DATE> <TIME> [VERBOSE] [SERVICE] <MESSAGE>
```

Формат сообщений об операциях, производимых пользователями (аккаунтинга):

```
<DATE> <TIME> [VERBOSE] [IMISH] AUDIT [USER] <MESSAGE>
```

Таблица 126 — Параметры записи сообщений

Параметр	Описание
DATE	дата события в формате ГГГГ-ММ-ДД
TIME	время события в формате ЧЧ:ММ:СС.CCCCCC
VERBOSE	уровень события: <ul style="list-style-type: none">- 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_erpc.c:488,imi_erpc_init] ->
imi_erpc_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`.

37.2 Включение и 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 127 — Список доступных подсистем и параметров данной команды `debug`



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	

Подсистема/параметр команды	Описание	Режим
<code>igmp encode</code>	IGMP encode	
<code>igmp events</code>	IGMP events	
<code>igmp fsm</code>	IGMP FSM	
<code>igmp tib</code>	IGMP Tree-Info-Base (TIB)	
<code>igmp vrf</code>	VPN Routing/Forwarding instance	
<code>isis</code>	Intermediate System - Intermediate System (IS-IS)	Административный и конфигурационный
<code>isis all</code>	Enable all debugging	
<code>isis authentication</code>	IS-IS Authentication	
<code>isis checksum</code>	IS-IS Check-Sum	
<code>isis events</code>	IS-IS Events	
<code>isis hello</code>	IS-IS Hello Debug	
<code>isis ifsm</code>	IS-IS Interface Finite State Machine	
<code>isis local-updates</code>	IS-IS Local Updates	
<code>isis lsp</code>	IS-IS Link State PDU	
<code>isis mpls</code>	Multi-Protocol Label Switching (MPLS)	
<code>isis nfsm</code>	IS-IS Neighbor Finite State Machine	
<code>isis nsm</code>	IS-IS NSM information	
<code>isis pdu</code>	IS-IS Protocol Data Unit	
<code>isis protocol-errors</code>	IS-IS Protocol Errors	
<code>isis rib</code>	IS-IS RIB information	
<code>isis spf</code>	IS-IS SPF Calculation	
<code>ldp</code>	Label Distribution Protocol (LDP)	Административный и конфигурационный
<code>ldp advertise-labels</code>	List IP access lists of advertise-labels	
<code>ldp all</code>	Enable all debugging	

Подсистема/параметр команды	Описание	Режим
<code>ldp dsm</code>	LDP Downstream SM	
<code>ldp events</code>	LDP events	
<code>ldp fsm</code>	LDP FSM	
<code>ldp graceful-restart</code>	LDP Graceful Restart Debugging	
<code>ldp hexdump</code>	LDP HEXDUMP	
<code>ldp nsm</code>	NSM messages	
<code>ldp packet</code>	LDP packet	
<code>ldp qos</code>	LDP QoS	
<code>ldp rib</code>	RIB messages	
<code>ldp tsm</code>	LDP Trunk SM	
<code>ldp usm</code>	LDP Upstream SM	
<code>ldp vc</code>	LDP VC Info	
<code>mrrib</code>	Multicast Routing Information Base (MRIB)	Административный и конфигурационный
<code>mrrib all</code>	All MRIB debugging	
<code>mrrib event</code>	MRIB events	
<code>mrrib fib-msg</code>	MRIB FIB messages	
<code>mrrib mrrib-msg</code>	MRIB MRIB IPC messages	
<code>mrrib mrt</code>	MRIB route	
<code>mrrib mtrace</code>	MRIB traceroute	
<code>mrrib mtrace-detail</code>	MRIB traceroute detailed debugging	
<code>mrrib nsm-msg</code>	MRIB NSM IPC messages	
<code>mrrib register-msg</code>	MRIB PIM Register messages	
<code>mrrib stats</code>	MRIB statistics	
<code>mrrib vif</code>	MRIB interface	
<code>mrrib vrf</code>	VPN Routing/Forwarding instance	
<code>nsm</code>	Network Service Module (NSM)	Административный и конфигурационный

Подсистема/параметр команды	Описание	Режим
<code>nsm all</code>	Enable all debugging	
<code>nsm events</code>	NSM events	
<code>nsm packet</code>	NSM packets	
<code>ospf</code>	Open Shortest Path First (OSPF)	Административный и конфигурационный
<code>ospf all</code>	Enable all debugging	
<code>ospf database-timer</code>	OSPF Database Timers	
<code>ospf events</code>	OSPF events information	
<code>ospf graceful-restart</code>	OSPF graceful-restart	
<code>ospf ifsm</code>	OSPF Interface State Machine	
<code>ospf lsa</code>	OSPF Link State Advertisement	
<code>ospf n fsm</code>	OSPF Neighbor State Machine	
<code>ospf nsm</code>	OSPF NSM information	
<code>ospf packet</code>	OSPF packets	
<code>ospf policy</code>	OSPF policy information	
<code>ospf redistrib</code>	OSPF redistribute information	
<code>ospf retransmission</code>	OSPF Debug retransmission information	
<code>ospf rib</code>	OSPF RIB information	
<code>ospf route</code>	OSPF route information	
<code>pim</code>	Protocol Independent Multicast (PIM)	Административный и конфигурационный
<code>pim all</code>	All PIM debugging	
<code>pim events</code>	PIM events	
<code>pim mfc</code>	PIM MFC updates	
<code>pim mib</code>	PIM mib	
<code>pim mtrace</code>	Mtrace messages	
<code>pim nexthop</code>	PIM nexthop	
<code>pim nsm</code>	NSM message	

Подсистема/параметр команды	Описание	Режим
<code>pim packet</code>	PIM packet	
<code>pim state</code>	PIM state	
<code>pim timer</code>	PIM timers	
<code>pim vrf</code>	VPN Routing/Forwarding instance	
<code>rib</code>	Routing Information Base (RIB)	Административный и конфигурационный
<code>rib all</code>	Enable all debugging	
<code>rib events</code>	RIB events	
<code>rib nsm</code>	NSM messages	
<code>rib packet</code>	RIB packets	
<code>rib routing</code>	Enable debugging for routing events	
<code>security-profile</code>	Security profile	Административный и конфигурационный
<code>vrrp</code>	Virtual Router Redundancy Protocol (VRRP)	Административный и конфигурационный
<code>vrrp all</code>	Enable all debugging	
<code>vrrp events</code>	VRRP events	
<code>vrrp packet</code>	VRRP packets	
<code>aaa</code>	AAA	Конфигурационный
<code>aaa 1</code>	critical	
<code>aaa 2</code>	error	
<code>aaa 3</code>	warning	
<code>aaa 4</code>	notice	
<code>aaa 5</code>	info	
<code>aaa 6</code>	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**.

37.3 Архив логов

37.3.1 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 рапорты хранятся локально на маршрутизаторе. Для их просмотра следует воспользоваться командой `show files reports`. В результате ее выполнения выводится список архивов логов с указанием размеров архивов и даты и времени их создания.

```
ecorouter#show files 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
```

37.3.2 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 files 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 files 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 files reports
```

No reports found!
ecorouter#

37.3.3 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).

Варианты применения команды `copy report` :

- `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 сохранён 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**.

37.4 Сниффинг

В EcoRouterOS можно включить sniffing трафика на физических портах устройства. Трафик записывается в файл с расширением PCAPNG, и хранится во внутреннем хранилище. Имя файла формируется автоматически и содержит имя порта, оно не может быть изменено.

Для старта sniffing трафика в режиме администрирования `enable-exec` (`ecorouter#`) введите команду: `service dump port <NAME> start`, где **NAME** — имя порта.

По умолчанию для каждого физического порта установлен лимит на запись в PCAPNG файл в 1000 пакетов, после сбора 1000 пакетов в файл sniffing трафика будет автоматически остановлен. Sniffing можно также остановить принудительно командой: `service dump port <NAME> stop`, где **NAME** — имя порта.

Для того, чтобы изменить лимит по умолчанию воспользуйтесь командой: `service dump port <NAME> limit (mbyte <1-100> | pkts <1-1000000>)`, где **NAME** — имя порта, а ключевые слова `mbyte` и `pkts` указывают тип лимита, лимит может быть задан в:

- мегабайтах — размер PCAPNG файла,
- количестве пакетов — в PCAPNG файле.

Вернуть лимит к значению по умолчанию можно командой `service dump port <NAME> limit unset` или командой `no service dump port <NAME> limit`, где **NAME** — имя порта.

При старте sniffing есть возможность задать фильтры для записи трафика, чтобы в конечный PCAPNG файл попал трафик включающий только определённый IP адрес, MAC адрес или протокол. Чтобы задать фильтр воспользуйтесь командой: `service dump port <NAME> filter (ether <WORD> | ip A.B.C.D | mac XXXX.XXXX.XXXX)`, где **NAME** — имя порта, а ключевые слова **ether**, **ip** и **mac** указывают тип фильтра:

- **A.B.C.D** — интересующий IP адрес (может быть как в качестве источника так и получателя в пакете),
- **XXXX.XXXX.XXXX** — интересующий MAC адрес (может быть как в качестве источника так и получателя во фрейме),
- **WORD** — Поле EtherType во фрейме укажет интересующий протокол, значение вводится в формате hex в пределах 0x600-0xffff (воспользуйтесь подсказкой в CLI, чтобы увидеть предустановленные фильтры для EtherType).

Максимальное кол-во созданных фильтров для каждого порта — 10, между ними будет работать логическое правило «ИЛИ». Для удаления правила воспользуйтесь командой:

```
no service dump port te0 filter <1-10> ,
```

где **<1-10>** — номер фильтра, который можно узнать с помощью команды: `show dump port <NAME> stats`, где **NAME** — имя порта.

Пример вывода:

```
ecorouter#show dump port ge1 stats
Stats for port:ge1
  limit: 1000 packets, current 0
  filter 1: enable(mac: any, ipv4: 1.1.1.1, ether type: any)
  filter 2: enable(mac: any, ipv4: 2.2.2.2, ether type: any)
```

После остановки сниффера сохранённые .pcapng файлы можно посмотреть с помощью команды: `show files dump`.

Для дальнейшего анализа .pcapng файлов присутствует возможность отправить их на удалённый сервер или ПК с помощью протокола SSH, воспользовавшись командой: `copy scp dump <FILENAME> <URL> [mgmt] [port <22-65535>] [vr <VR_NAME|default>] [vrf <VRF_NAME>]`, где **FILENAME** — имя PCAPNG файла (воспользуйтесь командой `show files dump`), а **URL** — конечный адрес получателя с указанием имени пользователя и пути в виде `admin@192.168.0.1/admin/reports/`. Можно указать дополнительные параметры: **mgmt** — интерфейс управления в качестве исходящего, **port** — номер SSH порта на сервере, **vr** — имя виртуального маршрутизатора или маршрутизатор по умолчанию, **vrf** — имя VRF.

ВНИМАНИЕ!

Включение сниффинга трафика на высокоскоростных физических портах снижает производительность устройства! Используйте это средство для отладки подконтрольно и с осторожностью, при необходимости воспользуйтесь силами технической поддержки вендора.

38 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 запрещённые для любых других ролей, отмечены буквой **d** (access denied).

Таблица 128 — Справочник по командам EcoRouter

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	+	+	



Команда	Описание	Режим консоли	Роли		
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 files	Show all config, reports or dumps files on storage	Польз	+	+	
show filter-map	Filtering rules	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	+	+	

Команда	Описание	Режим консоли	Роли		
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	+	+	

Команда	Описание	Режим консоли	Роли		
show vrrp	VRRP information	Польз	+	+	
terminal	Set terminal line parameters	Польз	+	+	+
undebg	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 'undebg')	Админ	+		
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	+		

Команда	Описание	Режим консоли	Роли		
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	+		
reload in <1-600>	Reboot device automatically after a certain timespan (in minutes)	Админ	+		
reload at <HH:MM>	Reboot device automatically at a certain time (within 24 hours). HH=00 to 23; MM=00 to 59				
reload cancel	Cancel the scheduled reboot	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	+	+	

Команда	Описание	Режим консоли	Роли		
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	+	+	
show users localdb	Display users database information	Админ	+	+	
show log	Display log	Админ	+	+	
show mirror-session	Mirror session status and configuration	Админ	+	+	

Команда	Описание	Режим консоли	Роли		
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	+	+	

Команда	Описание	Режим консоли	Роли		
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	+	+	+
undebg	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	+		

Команда	Описание	Режим консоли	Роли		
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	+		
dhcp-delay	Set delay after receiving DISCOVER and before sending OFFER message	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	+		

Команда	Описание	Режим консоли	Роли		
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	+		
no	Negate a command or set its defaults	Конф	+		
ntp	Configuration NTP	Конф	+		

Команда	Описание	Режим консоли	Роли		
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	+		

Команда	Описание	Режим консоли	Роли		
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	+		