ФАКЕЛ

Руководство по эксплуатации программы для ЭВМ "Программное обеспечение маршрутизатора «Факел»"

ФАКЕЛ

Оглавление

Обозначения и сокращения

API	–	Application Programming Interface
BFD	–	Bidirectional Forwarding Detection
BGP	–	Border Gateway Protocol
CIDR	–	Classless Inter-Domain Routing
DHCP	–	Dynamic Host Configuration Protocol
DNAT	–	Destination Network Address Translation
DNS	–	Domain Name System
EGP	–	Exterior Gateway Protocol
GRE	–	Generic Routing Encapsulation
HDD	–	Hard Disk Drive
HTTP	–	HyperText Transfer Protocol
ICMP	–	Internet Control Message Protocol
IGP	–	Interior Gateway Protocol
IKE	–	Internet Key Exchange
IP	–	Internet Protocol
IPSec	–	IP Security
IPv4	–	Internet Protocol Version 4
IPv6	–	Internet Protocol Version 6
IS-IS	–	Intermediate System to Intermediate System
L2TP	–	Layer 2 Tunnelling Protocol
MAC	–	Media Access Control
MTU	–	Maximum Transmission Unit
NAT	–	Network Address Translation
NHRP	–	Next Hop Resolution Protocol
OSPF	–	Open Shortest Path First
PBR	–	Policy-based Routing
PPTP	–	Point-to-Point Tunneling Protocol
RADIUS	–	Remote Authentication Dial In User Service
REST	–	Representational State Transfer
RIP	–	Routing Information Protocol
SCP	–	Secure copy
SFTP	–	Secure File Transfer Protocol
SNAT	–	Source Network Address Translation
SSD	–	Solid-State Drive

SSH	–	Secure Shell
Syslog	–	System Log
TCP	–	Transmission Control Protocol
TFTP	–	Trivial File Transfer Protocol
URL	–	Uniform Resource Locator
USB	–	Universal Serial Bus
VPN	–	Virtual Private Networks
VRRP	–	Virtual Router Redundancy Protocol
x86	–	Intel 8086 family
ГБ	–	Гигабайт
ОЗУ	–	Оперативная память
ОС	–	Операционная система
ПО	–	Программное обеспечение

Введение

О программном обеспечении

Программа для ЭВМ "Программное обеспечение маршрутизатора «Факел»" (далее – ПО **Факел,** "Программное обеспечение маршрутизатора «Факел»", программа, ПО) представляет собой комплексную систему по управлению сетевыми соединениями, трафиком и маршрутной информацией. ПО **Факел** основано на дистрибутиве операционной системы Debian GNU/Linux.

ПО **Факел** предназначено для:

- обработки сетевых соединений на существующих сетях передачи данных;
- обмена маршрутной информацией со смежными сетевыми устройствам;
- создания и управления виртуальными частными сетями Virtual Private Networks (далее – VPN);
- межсетевого экранирования сетевых сегментов;
- выполнения трансляции сетевых адресов;
- управление уровнем обслуживания сетевых соединений и их приоритезацией;
- обеспечения удаленного доступа пользователей к внутренним ресурсам;
- обеспечения отказоустойчивости аппаратных комплексов маршрутизаторов Факел.

Описание функциональных характеристик ПО

Функциональным назначением программы является обеспечение управления процессом обработки сетевых пакетов и обмена маршрутной информацией со смежным маршрутизирующим оборудованием.

ПО **Факел** обеспечивает следующие основные функции:

- инсталляция ПО **Факел** на аппаратных платформах и виртуальных машинах;
- обновление ПО **Факел**;
- аутентификация и авторизация администратора;
- управление конфигурацией системных и прикладных параметров ПО **Факел**;
- прием, обработка и отправка сообщений маршрутной информации;
- сбор, хранение, отображение и удаление информации о событиях системы;
- создание и управление параметрами VPN (Virtual Private Networks);

- управление IP-адресацией и параметрами интерфейсов;
- управление правилами межсетевого экранирования;
- трансляция сетевых адресов.

Системные требования

ПО **Факел** подходит для использования на платформах с архитектурой x86-64, включая:

- специализированные аппаратные платформы;
- виртуальные платформы на базе KVM.

При эксплуатации ПО **Факел** на специализированных аппаратных платформах системные требования для функционирования ПО **Факел**, зависят от технических характеристик модели аппаратной платформы. Технические характеристики аппаратных платформ заранее установлены производителем.

При эксплуатации ПО **Факел** на виртуальных платформах на базе KVM, системные требования для функционирования ПО **Факел** на виртуальной платформе, зависят от требований к производительности виртуальной среды и ее технических характеристик.

Минимальные системные требования для установки ПО **Факел** на виртуальную платформу:

- 2 процессорных ядра (Core) CPU;
- 2 Гигабайта (GB) оперативной памяти RAM;
- 10 Гигабайт (GB) дискового пространства.

Установка ПО

ПО **Факел** поддерживает установку на аппаратные платформы или на виртуальные машины. Для установки ПО **Факел** используется ISO-образ, который представляет собой образ для установки операционной системы в режиме реального времени «*live install image*».

Установка ПО на аппаратную платформу

Для установки ПО **Факел** на аппаратную платформу выполните следующие действия:

1. Подготовьте установочный USB-носитель с ISO-образом ПО **Факел**. Чтобы подготовить образ ПО **Факел** можно использовать приложение Rufus (или аналогичные ему) для Windows систем или использовать утилиту dd если вы предпочитаете использовать Linux системы.
2. Подключите USB-носитель с установочным образом в один из доступных USB-портов на аппаратной платформе. Аппаратная платформа во время подключения установочного USB-носителя должна быть выключена.
3. Подключите консольный кабель к консольному порту аппаратной платформы и к ПК или ноутбуку, с которого будет осуществляться подключение к аппаратной платформой.
4. Для подключения к аппаратной платформе через консольный порт:
   - Если вы используете ОС Windows для подключения по консольному кабелю используйте утилиту PuTTy. Запустите Диспетчер устройств и определите какой номер COM порта нужно использовать для подключения к аппаратной платформе. После этого запустите приложение PuTTy выберите тип подключения Serial. В поле Последовательная линия введите номер COM порта, а в поле Скорость установите значение 115200.
   - Если вы используете OS Linux для подключения по консольному кабелю используйте утилиту minicom. Запустите приложение терминала. В приложении терминала выполните команду:

     ```
     minicom -D /dev/ttyUSB0 -b 115200
     ```
5. Включите аппаратную платформу.
6. В процессе загрузки аппаратной платформы дождитесь, когда появится сообщение: *«Press ESC for boot menu»* и нажмите на клавишу «**ESC»**.
7. Выберите номер пункта из загрузочного меню, который отвечает за загрузку системы с USB-носителя и нажмите на клавишу **«ENTER»**.
8. Дождитесь загрузки ПО **Факел**.

9. Авторизуйтесь в системе. Для авторизации используйте имя пользователя *fakel* и пароль *fakel*.

10. После успешной авторизации в систему в командной строке выполните команду:

```
install image
```

11. Подтвердите запуск процесса установки ПО. Когда в консоли появится сообщение с предложением продолжить установку: *«Would you like to continue? (Yes/No) [Yes]»* и нажмите на клавишу «**ENTER»**.

12. Выберете автоматический способ разметки жесткого диска. Когда в консоли появится сообщение с выбором способа настройки разделов жестокого диска: *«Partition (Auto/Parted/Skip) [Auto]»* и нажмите на клавишу «**ENTER»**.

13. Выберите устройство, на которое будет установлено ПО **Факел**. Когда в консоли появится сообщение с выбором устройства на которое будет установлено ПО **Факел** введите имя устройства: *«Install the image on? [sda]»* и нажмите на клавишу «**ENTER»**.

14. Когда в консоли появится сообщение о том, что установка ПО **Факел** удалит все текущие данные с устройства: *«This will destroy all data on /dev/sda. Continue? (Yes/No) [No]»* и введите ***Yes*** и нажмите на клавишу «**ENTER»**.

15. Укажите размер *root* раздела. Когда в консоли появится сообщение о том, сколько места нужно выделить под root раздел: *«How big of a root partition should I create? (2000MB - 55021MB) [55021]MB:»* и нажмите на клавишу «**ENTER»**.

16. Укажите имя для образа, который будет установлен. Когда в консоли появится сообщение с предложением указать имя для устанавливаемого образа: *«What would you like to name this image? [1.0.0-224]»* и нажмите на клавишу «**ENTER»**.

17. Выберите какой файл будет использован для загрузки конфигурации ПО **Факел**. Когда в консоли появится сообщение с предложением выбрать путь к файлу с конфигурацией ПО **Факел**: *«Which one should I copy to sda? [/opt/vyatta/etc/config/config.boot]»* и нажмите на клавишу «**ENTER»**.

18. Установите пароль для учетной записи *fakel*.

19. Выберите устройство, на которое будет установлен *boot* раздел. Когда в консоли появится сообщение с предложением выбрать устройство для установки *boot* раздела:
«Which drive should GRUB modify the boot partition on? [sda]» и введите имя устройства и нажмите на клавишу «**ENTER»**.

20. После завершения процесса установки перезагрузите аппаратную платформу. Извлеките USB-носитель с установочным образом. В командной строке ПО **Факел** выполните команду:

```
reboot
```

21. Дождитесь успешной загрузки ПО **Факел** и авторизуйтесь в системе.

Установка ПО на виртуальную платформу

ПО **Факел** поддерживает следующие среды виртуализации:

- KVM
- ProxMox
- VMvare ESXi

Для установки ПО **Факел** на виртуальную платформу выполните следующие действия:

1. Добавить файл ISO-образа ПО **Факел** в хранилище образов на сервере виртуализации.
2. Создать виртуальную машину на сервере виртуализации со следующими параметрами (в данном примере представлены минимальные характеристики для создания виртуальной машины для установки ПО **Факел**):
   - **CPU**: 2 ядра
   - **RAM**: 2 Гб
   - **Hard Disk**: 10 ГБ
   - **Network Interfaces**: 2 шт.
   - **Disc Image File**: ISO-образ ПО **Факел**
3. Откройте консоль подготовленной виртуальной машины и включите виртуальную машину.
4. Дождитесь загрузки ПО **Факел** на виртуальной машине.
5. Авторизуйтесь в системе. Для авторизации используйте имя пользователя *fakel* и пароль *fakel*.
6. После успешной авторизации в систему в командной строке выполните команду:

```
install image
```

7. Подтвердите запуск процесса установки ПО **Факел**. Когда в консоли появится сообщение с предложением продолжить установку: *«Would you like to continue? (Yes/No) [Yes]»* и нажмите на клавишу «**ENTER»**.

8. Выберете автоматический способ разметки жесткого диска. Когда в консоли появится сообщение с выбором способа настройки разделов жестокого диска: *«Partition (Auto/Parted/Skip) [Auto]»* и нажмите на клавишу «**ENTER»**.

9. Выберите устройство, на которое будет установлено ПО **Факел**. Когда в консоли появится сообщение с выбором устройства на которое будет установлено ПО, введите имя устройства: *«Install the image on? [sda]»* и нажмите на клавишу «**ENTER»**.

10. Когда в консоли появится сообщение о том, что установка ПО **Факел** удалит все текущие данные с устройства: *«This will destroy all data on /dev/sda. Continue? (Yes/No) [No]»* и введите ***Yes*** и нажмите на клавишу «**ENTER»**.

11. Укажите размер *root* раздела. Когда в консоли появится сообщение о том, сколько места нужно выделить под *root* раздел: *«How big of a root partition should I create? (2000MB - 55021MB) [55021]MB»* и нажмите на клавишу «**ENTER»**.

12. Укажите имя для образа, который будет установлен. Когда в консоли появится сообщение с предложением указать имя для устанавливаемого образа: *«What would you like to name this image? [1.0.0-224]»* и нажмите на клавишу «**ENTER»**.

13. Выберите какой файл будет использован для загрузки конфигурации ПО **Факел**. Когда в консоли появится сообщение с предложением выбрать путь к файлу с конфигурацией ПО **Факел**: *«Which one should I copy to sda? [/opt/vyatta/etc/config/config.boot]»* - нажмите на клавишу «**ENTER»**.

14. Установите пароль для учетной записи *fakel*.

15. Выберите устройство, на которое будет установлен boot раздел. Когда в консоли появится сообщение с предложением выбрать устройство для установки boot раздела: *«Which drive should GRUB modify the boot partition on? [sda]»* - введите имя устройства и нажмите на клавишу «**ENTER»**.

16. После завершения процесса установки перезагрузите аппаратную платформу. В командной строке ПО **Факел** выполните команду:

```
reboot
```

17. Дождитесь успешной загрузки ПО **Факел** и авторизуйтесь в системе.

Командная строка

ПО **Факел** допускает использование интерфейса CLI в двух режимах:

- Режим Администрирования
- Режим Конфигурации

Режим Администрирования

Режим Администрирования позволяет использовать команды управления питанием устройства, получать информацию о текущем статусе операционной системы в целом и сервисов в ее составе в частности, а также выполнять диагностические команды по типу ***traceroute*** или ***ping***.

Интерфейс CLI предоставляет возможность получения справки по всем поддерживаемым командам посредством ввода символа «**?**». Клавиша «**TAB**» может быть использована для автозаполнения строки с командами, а также для проверки введенной команды, в случае ввода конфликтующих или неизвестных значений. При вводе в командной строке значения sh и последующем нажатии клавиши «**TAB**» операционная система предложит вариант команды ***show***. Повторное нажатие клавиши «**TAB**» приведет к выводу на экран всех возможных дополнительных команд и параметров, соответствующих команде ***show***.

Пример работы команды show:

```
fakel@fakel:~$ sh[tab]
set   show
fakel@fakel:~$ show [tab]
Possible completions:
    arp           Show Address Resolution Protocol (ARP) information
    bridge          Show bridging information
    cluster         Show clustering information
    configuration   Show running configuration
    conntrack       Show conntrack entries in the conntrack table
    conntrack-sync  Show connection syncing information
    date            Show system date and time
    dhcp           Show Dynamic Host Configuration Protocol (DHCP)
    dhcpv6          Show status related to DHCPv6
    disk            Show status of disk device
    dns             Show Domain Name Server (DNS) information
    file            Show files for a particular image
    firewall        Show firewall information
```

```
    flow-accounting Show flow accounting statistics
    hardware        Show system hardware details
    history         show command history
    host            Show host information
    incoming        Show ethernet input-policy information
: q
```

Вы можете пролистывать выводимую на экран информацию вверх с помощью сочетания клавиш «**Shift+PageUp**» и вниз с помощью сочетания клавиш «**Shift+PageDown**».

Когда вывод данных в командной строке в ответ на заданную команду содержит количество строк, превышающее буфер отображаемой на экране информации, выводимая информация будет разделена на части, о чем будет свидетельствовать отображение на экране символа «**:**».

При просмотре вывода данных с разделенной информацией доступны следующие действия:

- Использование клавиши «**q**» для прекращения вывода.
- Использование клавиши «**SPACE**» для перехода к следующей станице.
- Использование клавиши «**b**» для перехода к предыдущей странице.
- Использование клавиши «⏎» для перемещения выводимой на экран информации на одну строчку вниз.
- Использование клавиш «⇧» и «⇩» для перемещения выводимой информации на одну строчку вверх и на одну строчку вниз соответственно.
- Использование клавиш «⇦» и «⇨» для перемещения выводимой информации влево и вправо соответственно в случае, если вывод данных в строке содержит количество символов, превышающее буфер отображаемой на экране информации по горизонтали.

Режим Конфигурации

Режим Конфигурации позволяет использовать команды, приводящие к изменению в конфигурации операционной системы.

> **Подсказка**
>
> *Обозначение текущего режима при запросе ввода в командной строке поменяется с символа «**$**» (режим Администрирования) на символ «#» (режим Конфигурации).*

Для перевода операционной системы из режима Администрирования в режим Конфигурации используйте команду ***configure***:

```
fakel@fakel:~$ configure
[edit]
fakel@fakel#
```

Для перевода устройства под управлением операционной системы обратно в режим Администрирования используйте команду ***exit***:

```
fakel@fakel# exit
exit
fakel@fakel:~$
```

Работа в режиме Конфигурации

Все команды, выполняемые в режиме Конфигурации, связаны с определенным уровнем иерархической структуры, который определяется администратором в процессе ввода команд. В режиме Конфигурации можно производить изменения в настройках операционной системы с самого верхнего уровня иерархической структуры, но в таком случае цепочка команд при их ручном вводе получится достаточно длинной. Изменение уровня иерархической структуры сократит длину используемых команд и упростит настройку конфигурации операционной системы.

Для изменения текущего уровня иерархической структуры используйте команду ***edit***:

```
fakel@fakel# edit interfaces ethernet eth0
[edit interfaces ethernet eth0]
fakel@fakel#
```

Пример выше демонстрирует работу команды ***edit***. После ввода команды ***edit interfaces ethernet eth0*** операционная система переходит на уровень иерархической структуры конфигурации, связанный с цепочкой команд ***interfaces ethernet eth0***. Все команды, будут выполняться в отношение данного уровня.

Для возвращения на самый верхний уровень иерархической структуры используйте команды:

- `top`
- `exit`

Для перемещения на один уровень иерархической структуры вверх используйте команду:

- `up`

Для вывода на экран рабочей конфигурации используйте команду:

- `show`

Команда ***show***, введенная в режиме Конфигурации, приведет к отображению рабочей конфигурации с обозначением ее строк символами:

- «**+**» - добавленные строки;
- «**>**» - измененные строки;
- «**-**» - удаленные строки.

Пример работы команды show:

```
fakel@fakel# show interfaces
        ethernet eth0 {
        description MY_OLD_DESCRIPTION
        disable
        hw-id 00:53:dd:44:3b:03
    }
    loopback lo {
    }
    [edit]
fakel@fakelset interfaces ethernet eth0 address dhcp
    [edit]
fakel@fakelset interfaces ethernet eth0 description «Management»
    [edit]
fakel@fakel# delete interfaces ethernet eth0 disable
    [edit]
fakel@fakel# show interfaces
    ethernet eth0 {
    +   address dhcp
    >   description Management
    -   disable
    hw-id 00:53:dd:44:3b:03
    }
    loopback lo {
    }
```

Пример выше демонстрирует работу команды ***show interfaces***, которая выводит информацию о рабочей конфигурации сетевых интерфейсов до и после внесения изменений в настройки интерфейса *eth0*.

Для вывода информации обо всех выполненных командах set используется набор команд:

- `show | commands`

Пример работы команды show | commands:

```
fakel@fakelshow interfaces ethernet eth0 | commands
    set address dhcp
    set hw-id 00:53:ad:44:3b:03
```

Пример выше демонстрирует работу команды ***show interfaces ethernet eth0 | commands***, которая выводит информации о выполненных командах для интерфейса *eth0*.

В режиме Конфигурации вывод команды ***show*** зависит от текущего активного уровня иерархической структуры. Команда ***show*** выводит информацию о настройках конфигурации, которая относится к текущему активному уровню иерархической структуры.

```
[edit interfaces ethernet eth0]
fakel@fakel# show
address dhcp
hw-id 00:53:ad:44:3b:03
```

Пример выше демонстрирует вывод команды ***show***, когда система находится на иерархическом уровне, который отвечает за настройку интерфейса *eth0*.

Для выхода их режима Конфигурации используйте команду:

- `exit`

Во время выполнения команды ***exit*** операционная система должна находиться на самом верхнем уровне иерархической структуры. Использование данной команды на любой другом уровне приведет к перемещению на самый верхний уровень иерархической структуры.

```
[edit interfaces ethernet eth0]
fakel@fakel# exit
[edit]
fakel@fakel# exit
fakel@fakel:~$
```

Пример выше демонстрирует работу команды ***exit***. После первого ввода команды ***exit*** система переходит на самый верхний иерархический уровень. Повторный ввод команды ***exit*** переводит систему в режим Администрирования.

Доступ к командам режима Администрирования через режим Конфигурации

Когда ПО **Факел** находится в режиме Конфигурации, команды режима Администрирования не доступны для использования напрямую. Все команды режима Администрирования можно использовать в составе набора команд:

- `run [command]`

Где параметр ***command*** это команда из режима Администрирования.

При использовании набора команд ***run [command]*** автозаполнение строки с командами при помощи клавиши «**TAB**» и получение справки посредством ввода символа «**?**» доступны для использования.

Пример работы команды run:

```
[edit]
fakel@fakel# run show interfaces
Codes: S - State, L - Link, u - Up, D - Down, A - Admin Down
Interface         IP Address                      S/L  Description
---------         ----------                      ---  -----------
eth0              0.0.0.0/0                       u/u
```

Пример выше демонстрирует работу команды ***run show interfaces***, которая выводит информации о сетевых интерфейсах.

Работа с конфигурацией ПО

ПО **Факел** использует унифицированный формат файла для описания всех параметров своей конфигурации:

/config/config.boot

Это обеспечивает легкую работу с конфигурацией системы в части создания шаблонов, резервного копирования и восстановления. ПО **Факел** может быть перенесено на другое устройство с сохранением текущих настроек посредством создания копий определенных конфигурационных файлов.

Типы конфигурации ПО

В контексте ПО **Факел** существует три основных типа конфигурации:

- **Активная конфигурация** - конфигурация системы, которая загружена и используется в настоящий момент. Любое изменение конфигурации должно быть применено для включения в активную конфигурацию.
- **Рабочая конфигурация** - конфигурация системы, которая подвергается изменениям в режиме конфигурирования. Изменения, внесенные в рабочую конфигурацию, не вступят в силу до их применения с помощью команды **commit**, после ввода которой конфигурация становится активной.
- **Хранимая конфигурация** - конфигурация, записанная в файл с помощью команды ***save***. Такой подход позволяет безопасно хранить конфигурацию для последующего использования. В системе может храниться несколько файлов с конфигурацией. Конфигурация по умолчанию или загруженная конфигурация хранится в файле */config/config.boot*.

Просмотр активной конфигурации ПО Факел

В ПО **Факел** информацию об активной конфигурации системы можно получить различными способами:

- Вывод информацию в виде иерархически упорядоченных секций и параметров в их составе;
- Вывод полного списка всех команд, с помощью которых была сформирована активная конфигурация;
- Вывод информации в формате JSON;
- Вывод информации в формате JSON иерархического вида (секции и параметры упорядочены по вложенности).

Для получения информации об активной конфигурации ПО **Факел** в виде иерархически упорядоченных секций и параметров в их составе используйте команду:

```
show configuration
```

Пример работы команды show configuration:

```
fakel@fakel:~$ show configuration
    interfaces {
    ethernet eth0 {
        address dhcp
        hw-id 00:53:00:00:aa:01
    }
    loopback lo {
    }
}
```

```
service {
    ssh {
        port 22
    }
}
system {
    config-management {
        commit-revisions 20
    }
    console {
        device ttyS0 {
            speed 9600
        }
    }
    login {
        user fakel {
            authentication {
                encrypted-password ****************
            }
            level admin
        }
    }
    ntp {
        server 0.pool.ntp.org {
        }
        server 1.pool.ntp.org {
        }
        server 2.pool.ntp.org {
        }
    }
    syslog {
        global {
            facility all {
                level notice
            }
            facility protocols {
                level debug
```

```
            }
        }
    }
}
```

Для вывода полного списка всех команд, с помощью которых была сформирована активная конфигурация, используйте команду:

- `show configuration commands`

Пример работы команды show configuration commands:

```
fakel@fakel:~$ show configuration commands
    set interfaces ethernet eth0 address 'dhcp'
    set interfaces ethernet eth0 hw-id '00:53:dd:44:3b:0f'
    set interfaces loopback 'lo'
    set service ssh port '22'
    set system config-management commit-revisions '20'
    set system console device ttyS0 speed '9600'
    set system login user fakel authentication encrypted-password
    set system login user fakel level 'admin'
    set system ntp server '0.pool.ntp.org'
    set system ntp server '1.pool.ntp.org'
    set system ntp server '2.pool.ntp.org'
    set system syslog global facility all level 'notice'
    set system syslog global facility protocols level 'debug'
```

Подсказка

*Используйте набор команд **show configuration commands | strip-private** в случае, если вы хотите скрыть конфиденциальную информацию, содержащуюся в конфигурации при ее отображении. Это может понадобиться в случае, если вы захотите передать или опубликовать снимок конфигурации кому-либо.*

Для просмотра активной конфигурации в формате JSON, используйте команду:

- `show configuration json`

Пример работы команды show configuration json:

```
fakel@fakel:~$ show configuration json
{"interfaces": {"ethernet": {"eth0": {"address": ["192.0.2.11/24",
"192.0.2.35/24"],     "hw-id":     "52:54:00:48:a0:c6"},     "eth1":
```

```
{"address": ["203.0.113.1/24"], "hw-id": "52:54:00:fc:50:0b"}},
"loopback": {"lo": {}}}, "protocols": {"static": {"route":
{"0.0.0.0/0": {"next-hop": {"192.0.2.254": {}}}}}}}, "service":
{"ssh": {"disable-host-validation": {}}}, "system": {"config-
management": {"commit-revisions": "100"}, "console": {"device":
{"ttyS0": {"speed": "115200"}}}, "host-name": "r11-fakel", "login":
{"user": {"fakel": {"authentication": {"encrypted-password":
"$6$Vt68...F0", "plaintext-password": "", "public-keys":
{"fakel@fakel": {"key": "AAAAxxx=", "type": "ssh-rsa"}}}}}}, "name-
server": ["203.0.113.254"], "ntp": {"server": {"time1.fakel.net":
{}, "time2.fakel.net": {}, "time3.fakel.net": {}}}}, "syslog":
{"global": {"facility": {"all": {"level": "info"}, "protocols":
{"level": "debug"}}}}}, "time-zone": "Russia/Moscow"}}
```


Для просмотра активной конфигурации в формате JSON иерархического вида (секции и параметры упорядочены по вложенности), используйте команду:

- `show configuration json pretty`

Пример работы команды show configuration json pretty:

```
fakel@fakel:~$ show configuration json pretty
{
    "interfaces": {
        "ethernet": {
            "eth0": {
                "address": [
                    "192.0.2.11/24",
                    "192.0.2.35/24"
                ],
                "hw-id": "52:54:00:48:a0:c6"
            },
            "eth1": {
                "address": [
                    "203.0.113.1/24"
                ],
                "hw-id": "52:54:00:fc:50:0b"
            }
        },
        "loopback": {
            "lo": {}
        }
    },
```

```
    "protocols": {
        "static": {
            "route": {
                "0.0.0.0/0": {
                    "next-hop": {
                        "192.0.2.254": {}
                    }
                }
            }
        }
    },
    "service": {
        "ssh": {
            "disable-host-validation": {}
        }
    },
    "system": {
        "config-management": {
            "commit-revisions": "100"
        },
        "console": {
            "device": {
                "ttyS0": {
                    "speed": "115200"
                }
            }
        },
        "host-name": "r11-fakel",
        "login": {
            "user": {
                "fakel": {
                    "authentication": {
                        "encrypted-password": "$6$Vt68...F0",
                        "plaintext-password": "",
                        "public-keys": {
                            "fakel@fakel": {
                                "key": "AAAAxxx=",
```

```
                                "type": "ssh-rsa"
                            }
                        }
                    }
                }
            },
            "name-server": [
                "203.0.113.254"
            ],
            "ntp": {
                "server": {
                    "time1.fakel.net": {},
                    "time2.fakel.net": {},
                    "time3.fakel.net": {}
                }
            },
            "syslog": {
                "global": {
                    "facility": {
                        "all": {
                            "level": "info"
                        },
                        "protocols": {
                            "level": "debug"
                        }
                    }
                }
            },
            "time-zone": "Asia/Novosibirsk"
        }
}
```

Представленные выше сценарии использования команды ***show*** работают, когда операционная система находится в режиме Администрирования. В режиме Конфигурации данные сценарии не работают, однако есть способ получить доступ к командам режима Администрирования, когда операционная система находится в режиме Конфигурации.

Подробная информации об использовании команд режима Администрирования из режима Конфигурации представлена в разделе **Доступ к командам режима Администрирования через режим Конфигурации**.

Редактирование рабочей конфигурации ПО Факел

Для редактирования рабочей конфигурации операционной системы используются команды ***set*** и ***delete***. Команды ***set*** и ***delete*** работают, когда операционная система находится в режиме Конфигурации.

Чтобы установить значение существующего параметра в конфигурации или добавить новый параметр, используется команда:

```
set
```

Команда ***set*** имеет иерархическую структуру. Выполняемые команды выстраиваются в последовательную цепочку, например ***set interface ethernet eth0 address 192.0.2.100/24***. При этом вся введенная информация, являющаяся избыточной для данного уровня, удаляется из введенной цепочки команд.

Команды относятся к уровню, на котором они выполняются:

- Операционная система находится на самом верхнем уровне иерархической структур

  ```
  fakel@fakel# set interface ethernet eth0 address 192.0.2.100/24
  [edit]
  ```

  Пример выше демонстрирует работу команды ***set interface ethernet eth0 address 192.0.2.100/24***, которая выполняет настройку адреса для сетевого интерфейса *eth0* когда операционная система находится на самом верхнем уровне иерархической структуры.

- Операционная система находится на уровне иерархической структуры, который отвечает за конфигурацию сетевого интерфейса eth0

  ```
  fakel@fakel# set address 192.0.2.100/24
  [edit interfaces ethernet eth0]
  ```

  Пример выше демонстрирует работу команды ***set address 192.0.2.100/24***, которая выполняет настройку адреса для сетевого интерфейса *eth0* когда операционная система находится на уровне иерархической структуры, который отвечает за конфигурацию сетевого интерфейса *eth0*.

Обе команды, представленные в примерах выше, идентичны, но выполняются на разных уровнях иерархической структуры.

Чтобы удалить запись из конфигурации, используйте команду:

- `delete`

Выполнение команды ***delete*** также приводит к удалению всех нижестоящих уровней в иерархической структуре. При удалении записи элемента конфигурации его настройки переводятся к значению по умолчанию.

```
fakel@fakel# delete address 192.0.2.100/24
[edit interfaces ethernet eth0]
```

Пример выше демонстрирует работу команды ***delete address 192.0.2.100/24***, которая удаляет IP-адрес сетевого интерфейса *eth0*:

Чтобы применить изменения, которые были внесены в конфигурацию системы используйте команду:

- `commit`

Чтобы сохранить изменения, которые были внесены в конфигурацию системы используйте команду:

- `save`

Использование команды save без дополнительных параметров сохраняет конфигурацию системы в хранилище по умолчанию.

```
fakel@fakel# save
Saving configuration to '/config/config.boot'...
Done
```

По умолчанию конфигурация системы храниться в файле, расположенном по следующему пути:

/config/config.boot

Если необходимо разместить конфигурационный файл в другом месте, в качества хранилища можно указать локальный путь или адреса SCP, FTP и TFTP серверов.

```
fakel@fakel# save tftp://192.168.0.100/fakel.config.boot
Saving configuration to 'tftp://192.168.0.100/fakel-test.config.boot'...
########################################################## 100.0%
Done
```

Пример выше демонстрирует работу команды ***save tftp://192.168.0.100/fakel.config.boot***, которая сохраняет конфигурацию системы в файл *fakel.config.boot* на TFTP сервере *192.168.0.100*.

Для выхода из режима Конфигурации без применения внесенных изменений в конфигурации системы используйте команду ***exit discard***:

```
fakel@admin# exit discard
exit
fakel@admin:~$
```

Команда ***exit discard***, сбрасывает все изменения, внесенные в рабочую конфигурацию.

> **Подсказка**
>
> *При использовании команды exit для выхода из режима Конфигурации без применения внесенных изменений, операционная система выдает сообщение с ошибкой.*

Чтобы временно применить изменения, которые были внесены в конфигурацию, и установить период времени в минутах, необходимый для проверки корректности полученной конфигурации используйте команду:

- `commit-confirm`

Команда ***confirm*** должна быть выполнена в течение заданного периода времени для окончательного применения конфигурации. В противном случае операционная система автоматически выполнит перезагрузку и возврат к конфигурации предыдущей версии конфигурации. По умолчанию используется период времени длительностью в 10 минут.

Пример работы команды commit-confirm:

Нужно настроить межсетевой экран и проверить, что внесенные в конфигурацию изменения, не заблокируют доступ к операционной системе для администратора. Для этого используйте команду ***commit-confirm***. Если после применения, внесенных изменений доступ к межсетевому экрану не заблокирован, используйте команду ***confirm***, чтобы подтвердить примененные изменения. В случае потери доступа к межсетевому экрану после использования команды ***commit-confirm***, будет выполнена перезагрузка и возврат к конфигурации предыдущей версии.

```
fakel@fakel set interfaces ethernet eth0 firewall local name FromWorld
fakel@fakel# commit-confirm
commit confirm will be automatically reboot in 10 minutes unless confirmed
Proceed? [confirm]y
[edit]
```

```
fakel@fakel# confirm
[edit]
```

Команда ***commit-confirm*** временно применяет изменения, внесенные в конфигурацию системе в результате ввода команды ***set interfaces ethernet eth0 firewall local name FromWorld***. Если в течение 10 минут не будет выполнена команда ***confirm***, в результате перезагрузки устройства будет выполнен возврат к предыдущей версии конфигурации системы.

Чтобы скопировать определенный элемент конфигурации используйте команду:

- `copy`

С помощью команды ***copy*** можно копировать целые ветви иерархической структуры конфигурации.

Пример работы команды copy:

Для межсетевого экрана нужно создать набор правил *FromWorld* с одним правилом фильтрации, разрешающим трафик из определенной подсети и добавить в этот набор похожее правило, но уже для другой подсети. Для этого нужно перейти на соответствующий уровень иерархической структуры конфигурации *firewall name FromWorld*, а затем использовать набор команд ***copy rule 10 to rule 20*** для создания нового правила через копирование данных из существующего, после чего модифицировать правило с номером 20 необходимым образом.

- *Просмотр рабочей конфигурации набора правил FromWorld*

```
fakel@fakel# show firewall name FromWorld
        default-action drop
        rule 10 {
            action accept
            source {
                address 203.0.113.0/24
            }
        }
    [edit]
```

- *Переход на иерархический уровень FromWorld*

```
fakel@fakel# edit firewall name FromWorld
[edit firewall name FromWorld]
```

- *Копирование параметров*

```
fakel@fakel# copy rule 10 to rule 20
[edit firewall name FromWorld]
```

- *Изменение параметров IP-адреса*

```
fakel@fakel# set rule 20 source address 198.51.100.0/24
[edit firewall name FromWorld]
```

- *Применение внесенных изменений*

```
fakel@fakel# commit
[edit firewall name FromWorld]
```

Чтобы переименовать определенный элемент конфигурации используйте команду:

- `rename`

С помощью команды ***rename*** можно переименовывать ветви иерархической структуры конфигурации.

```
fakel@fakel# rename rule 10 to rule 5
[edit firewall name FromWorld]
fakel@fakel# commit
[edit firewall name FromWorld]
```

Пример выше демонстрирует работу команды ***rename rule 10 to rule 5***, которая переименовывает 10-е правило межсетевого экрана.

Для просмотра внесенных изменений в рабочую конфигурацию используйте команду ***show***:

```
fakel@fakel# show
    default-action drop
    +rule 5 {
        action accept
        source {
            address 203.0.113.0/24
        }
    }
    rule 20 {
        action accept
        source {
```

```
            address 198.51.100.0/24
        }
    }
```

Пример выше демонстрирует работу команды ***show*** на уровне иерархической структуры, который отвечает за настройку набора привил межсетевого экрана *FromWorld*.

Чтобы добавить текстовый комментарий (аннотацию) к выбранной секции конфигурации <*config node*> используйте команду:

- `comment <config node> "comment text"`

В иерархической структуре конфигурации комментарий добавляется до выбранной секции. При выводе на экран комментарий заключается в рамки, обозначаемые разделителями «***/****» (начало комментария) и «****/***» (конец комментария). Добавление комментариев требует использования команды ***commit***, как и любые другие изменения, вносимые в конфигурацию. Для удаления существующего комментария из текущей конфигурации используйте ту же команду, но укажите пустую строку «**""**» в качестве параметра.

```
fakel@fakel# comment firewall all-ping "Comment example"
fakel@fakel# commit
fakel@fakel# show
    firewall {
        /* Comment example */
        all-ping enable
        broadcast-ping disable
        ...
    }
```

Пример выше демонстрирует работу команды ***comment firewall all-ping "Comment example"***, которая добавляет комментарий к секции *firewall*.

> **Подсказка**
>
> *Важно отметить, что комментарий добавляется до выбранной секции конфигурации, то есть при использовании команды show <section> он не будет отображаться. Например, если добавить комментарий для к секции firewall, а затем использовать набор команд show firewall, то на экран будет выведено содержимое данной секции, но не комментарий.*

Управление версиями конфигурации ПО Факел

ПО **Факел** содержит встроенную систему учета версий конфигурации. Эта система автоматически сохраняет копию каждой предыдущей примененной конфигурации - версию конфигурации. Версии конфигурации хранятся локально и используются для отката, однако они также могут быть размещены на внешнем (удаленном) хранилище с целью организации архива резервных копий.

Локальное хранение версий конфигурации ПО Факел

Версии конфигурации автоматически сохраняются на диске устройства. Можно просмотреть список всех версий, выполнить сравнение версий и откат к любой версии в случае возникновения проблем с функционированием операционной системы.

Для просмотра списка всех версий конфигурации, размещенных локально на диске устройства используйте команду:

```
show system commit
```

Пример работы команды show system commit:

```
fakel@fakel:~$ show system commit
0   2015-03-30 08:53:03 by fakel via cli
1   2015-03-30 08:52:20 by fakel via cli
2   2015-03-26 21:26:01 by root via boot-config-loader
3   2015-03-26 20:43:18 by root via boot-config-loader
4   2015-03-25 11:06:14 by root via boot-config-loader
5   2015-03-25 01:04:28 by root via boot-config-loader
6   2015-03-25 00:16:47 by fakel via cli
7   2015-03-24 23:43:45 by root via boot-config-loader
```

Чтобы установить ограничение на количество версий конфигурации, сохраняемых локально на диске устройства используйте команду:

```
set system config-management commit-revisions <N>
```

Значение параметра *N* может быть в диапазоне от 0 до 65535. Когда количество версий конфигурации превышает заданное значение, старые версии удаляются. Значение параметра *N* по умолчанию подразумевает хранение до 100 версий конфигурации локально.

Сравнение версий конфигурации ПО Факел

ПО **Факел** позволяет сравнивать различные версии конфигурации.

Для вывода на экран списка отличий между выбранными версиями конфигурации используйте команду ***compare <saved | N> <M>***:

```
fakel@fakel# compare [tab]
Possible completions:
    <Enter> Compare working & active configurations
    saved   Compare working & saved configurations
    <N>     Compare working with revision N
    <N> <M> Compare revision N with M
    Revisions:
        0    2013-12-17 20:01:37 root by boot-config-loader
        1    2013-12-13 15:59:31 root by boot-config-loader
        2    2013-12-12 21:56:22 fakel by cli
        3    2013-12-12 21:55:11 fakel by cli
        4    2013-12-12 21:27:54 fakel by cli
        5    2013-12-12 21:23:29 fakel by cli
        6    2013-12-12 21:13:59 root by boot-config-loader
        7    2013-12-12 16:25:19 fakel by cli
        8    2013-12-12 15:44:36 fakel by cli
        9    2013-12-12 15:42:07 root by boot-config-loader
        10   2013-12-12 15:42:06 root by init
```

Команда ***compare*** позволяет выполнять следующие действия:

- Сравнивать различные типы конфигураций;
- Сравнивать различные версии конфигурации с помощью команды ***compare N M***, где параметры *N* и *M* - идентификаторы версий.

При выводе на экран результатов сравнения двух версий символом «**+**» обозначаются добавленные в конфигурацию *N* элементы по сравнению с конфигурацией *M*, а символом «**-**» обозначаются удаленные из конфигурации элементы соответственно.

Пример работы команды compare:

```
fakel@fakel# compare 0 6
[edit interfaces]
+    dummy dum1 {
+        address 10.189.0.1/31
+    }
[edit interfaces ethernet eth0]
+    vif 99 {
```

```
+        address 10.199.0.1/31
+    }
-    vif 900 {
-        address 192.0.2.4/24
-    }
```

Пример выше демонстрирует работу команды ***compare 0 6***, которая сравнивает *0* и *6* версии конфигурации операционной системы.

Для вывода на экран результатов сравнения двух версий примененной конфигурации используйте команду:

- `run show system commit diff <number>`

По умолчанию на экран выводятся результаты сравнения с активной конфигурацией.

Пример работы команды run show system commit diff <number>:

```
fakel@fakel# run show system commit diff 4
[edit system]
+    ipv6 {
+        disable-forwarding
+    }
```

Пример выше демонстрирует работу команды ***run show system commit diff 4***, которая выводит результат сравнения версии активной конфигурации с конфигурацией *4*.

Откат изменений конфигурации ПО Факел

Для отката изменений конфигурации к версии, указанной в параметре *N* используйте команду:

- `rollback <N>`

Пример работы команды rollback <N>:

```
fakel@fakel# compare 1
[edit system]
>    host-name fakel-1
[edit]
fakel@fakel# rollback 1
Proceed with reboot? [confirm][y]
Broadcast message from root@fakel-1 (pts/0) (Tue Dec 17 21:07:45
2013):
```

```
The system is going down for reboot NOW!
```

Ввод команды ***rollback*** приведет к применению указанной версии конфигурации и перезагрузке системы.

Использование внешнего хранилища версий конфигурации ПО Факел

ПО **Факел** позволяет загружать версию конфигурации на внешнее (удаленное) хранилище после каждого использования команды ***commit***. Для сохранения версии конфигурации на внешнее хранилище нужно указать его адрес.

В качестве такого хранилища могут быть использованы TFTP, FTP, SCP и SFTP серверы:

- *scp://<user>:<passwd>@<host>:/<dir>*
- *sftp://<user>:<passwd>@<host>/<dir>*
- *ftp://<user>:<passwd>@<host>/<dir>*
- *tftp://<host>/<dir>*

Таким образом каждый раз, когда конфигурация успешно применяется, файл *config.boot* копируется по указанному адресу. При этом копии задается новое имя *config.boot-hostname.YYYYMMDD_HHMMSS*. Копирование может выполняться по нескольким адресам одновременно.

Для того, чтобы задать адрес внешнего (удаленного) хранилище в формате URI используйте команду:

```
set system config-management commit-archive location <URI>
```

> **Подсказка**
>
> *Количество версий конфигурации, хранимых локально на диске устройства, которое может задать администратор с помощью команды set system config-management commit-revisions <N> не распространяется на внешнее (удаленное) хранилище.*

Операционная система не позволяет установить с внешним (удаленным) хранилищем защищенное (SSL/TLS) соединение, так как невозможно подтвердить легитимность данного хоста. Однако есть обходное решение данной проблемы через добавление отпечатка SSH ключа хоста в файл *~/.ssh/known_hosts*:

```
ssh-keyscan <host> >> ~/.ssh/known_hosts
```

Сохранение и загрузка конфигурации ПО Факел вручную

Команды ***save*** и ***load*** используются для управления файлами конфигурации вручную.

Чтобы сохранить файл конфигурации используйте команду:

```
save
```

Чтобы загрузить конфигурацию из файла используйте команду:

```
load
```

Команда ***load <URI>*** используется для загрузки конфигурации, которая в итоге заменит активную конфигурацию. Для данной команды необходимо определить расположение файла в виде пути до него. В качестве расположения можно указать локальный путь или адреса SCP, SFTP, FTP, HTTP, HTTPS и TFTP серверов:

```
fakel@fakel# load
Possible completions:
  <Enter>                                  Load from system config file
  <file>                                   Load from file on local machine
  scp://<user>:<passwd>@<host>:/<file>     Load from file on remote machine
  sftp://<user>:<passwd>@<host>/<file>     Load from file on remote machine
  ftp://<user>:<passwd>@<host>/<file>      Load from file on remote machine
  http://<host>/<file>                     Load from file on remote machine
  https://<host>/<file>                  Load from file on remote machine
  tftp://<host>/<file>                    Load from file on remote machine
```

Восстановление конфигурации по умолчанию

Если требуется сбросить рабочую конфигурацию до состояния по умолчанию используется команда **load /opt/fakel/etc/config.boot.default**. При этом операционная система запросит подтверждение данного действия. Для подтверждения действия используется команду ***commit***. После подтверждения действия операционная система будет использовать конфигурацию по умолчанию.

> **Подсказка**
>
> *В процессе сброса конфигурации до состояния по умолчанию удаленное подключение к устройству будет сброшено. Поэтому перед выполнением сброса рекомендуется*

> *создать копию конфигурации, внести в нее изменения для обеспечения сетевой связности, а потом загрузить измененную конфигурацию.*

Сетевые интерфейсы

Ethernet интерфейс

Базовый функционал Ethernet интерфейса

Сетевой интерфейс Ethernet позволяет устройству подключаться к локальной вычислительной сети, используя протокол Ethernet в качестве механизма передачи данных.

Существует множество стандартов Ethernet, которым должен соответствовать сетевой интерфейс Ethernet, с различными скоростями передачи и типами корректировки ошибок. Ethernet - это стандарт передачи двоичных данных, который, несмотря на определенные аппаратные характеристики, не зависит от аппаратного обеспечения, поэтому сетевой интерфейс Ethernet может использовать любое оборудование для передачи данных - от оптоволокна, коаксиального медного кабеля до беспроводной связи, в зависимости от возможностей оборудования, на которое отправляется/принимается сигнал, и требуемой скорости передачи данных по сети.

Основные настройки Ethernet интерфейса

- `set interfaces ethernet <ethN> address <address|dhcp|dhcpv6>`

Задает адрес <*address* | *dhcp* | *dhcpv6*> для сетевого интерфейса <*ethN*>.

- адрес может быть указан несколько раз как IPv4 и/или IPv6 адрес, например, *192.0.2.1/24* и/или *2001:db8::1/64*
- адрес интерфейса dhcp принимается по протоколу DHCP от DHCP-сервера данного сегмента
- dhcpv6 адрес интерфейса получен DHCPv6 от DHCPv6-сервера на данном сегменте

- `set interfaces ethernet <ethN> description <description>`

Задает псевдоним <*description*> для сетевого интерфейса <*ethN*>. Например, псевдоним может использоваться командой show interfaces или средствами мониторинга на базе SNMP.

- `set interfaces ethernet <ethN> disable`

Отключает сетевой интерфейс <*ethN*>. После отключения интерфейс будет переведен в состояние административного отключения (A/D).

- `set interfaces ethernet <ethN> disable-flow-control`

Отключает генерацию управления потоком Ethernet (кадров паузы).

Управление потоком Ethernet — это механизм временного прекращения передачи данных в компьютерных сетях семейства Ethernet. Цель этого механизма - обеспечить нулевую потерю пакетов в условиях перегрузки сети.

Станция отправитель (компьютер или сетевой коммутатор) может передавать данные быстрее, чем другой конец канала связи может их принять. Используя управление потоком, принимающая станция может подать сигнал отправителю с просьбой приостановить передачу до тех пор, пока получатель не поймает его.

- `set interfaces ethernet <ethN> disable-link-detect`

С помощью этой команды можно настроить сетевой интерфейс таким образом, чтобы он не обнаруживал изменений физического состояния канала связи, например, при отсоединении кабеля.

По умолчанию сетевые интерфейсы настроены на обнаружение изменения физического состояния канала.

- `set interfaces ethernet <ethN> mac <address>`

Задает MAC адрес для сетевого интерфейса <*ethN*>.

- `set interfaces ethernet <ethN> mtu <mtu>`

Устанавливает размер MTU <*mtu*> для сетевого интерфейса <*ethN*>.

- `set interfaces ethernet <ethN> ip arp-cache-timeout <1-86400>`

Устанавливает тайм-аут записи в ARP-кэш в секундах <*1-86400*> для сетевого интерфейса <*ethN*>. По умолчанию это значение равно 30 секундам.

- `set interfaces ethernet <ethN> ip disable-arp-filter`

Отключает фильтрацию ARP на сетевом интерфейсе <*ethN*>.

Если этот параметр установлен, то ядро может отвечать на запросы ARP адресами с других интерфейсов.

По умолчанию этот параметр отключен. Это позволяет иметь несколько сетевых интерфейсов в одной подсети, и ответ на ARP адреса каждого интерфейса зависит от того, будет ли ядро маршрутизировать пакет с ARP адреса через этот интерфейс. Для работы этой функции необходимо использовать маршрутизацию на основе источника.

- `set interfaces ethernet <ethN> ip disable-forwarding`

Отключает переадресацию IP на сетевом интерфейсе <*ethN*>.

- `set interfaces ethernet <ethN> ip enable-arp-accept`

Включает прием ARP пакетов на сетевом интерфейсе <*ethN*>.

Определяет поведение для ARP пакетов, IP записи которых еще не присутствует в ARP таблице. Если настройка включена, то создаются новые записи в ARP таблице.

Как ответы, так и запросы типа *gratuitous arp* будут вызывать обновление ARP таблицы, если эта настройка включена.

Если в ARP таблице уже содержится IP-адрес кадра *gratuitous arp*, то ARP таблица будет обновлена независимо от того, включена или выключена эта настройка.

```
set interfaces ethernet <ethN> ip enable-arp-accept
```

Включает анонсирование ARP пакетов на сетевом интерфейсе **<***ethN*>.

```
set interfaces ethernet <ethN> ip enable-arp-ignore
```

Включает игнорирование ARP пакетов на сетевом интерфейсе <*ethN*>.

Этот параметр определяет режимы отправки ответов в ответ на полученные ARP запросы, разрешающие локальные целевые IP-адреса:

- Если параметр включен, ответ будет отправлен только в том случае, если целевой IP-адрес является локальным адресом, настроенным на входящем интерфейсе.
- Если параметр отключен (по умолчанию), ответ будет получен для любого локального целевого IP-адреса, настроенного на любом интерфейсе.

```
set interfaces ethernet <ethN> ip enable-proxy-arp
```

Включает прокси ARP на сетевом интерфейсе <*ethN*>. Прокси ARP позволяет интерфейсу Ethernet отвечать своим MAC адресом на ARP запросы IP-адресов назначения в подсетях, подключенных к другим интерфейсам системы.

```
set interfaces ethernet <ethN> ip proxy-arp-pvlan
```

Включить прокси ARP частного VLAN на сетевом интерфейсе <*ethN*>.

```
set interfaces ethernet <ethN> ip source validation <type>
```

Включает политику проверки источника по обратному пути, как указано в RFC 3704.

В настоящее время в RFC 3704 рекомендуется включать строгий режим для предотвращения подмены IP-адресов в результате DDos атак. При использовании асимметричной или другой сложной маршрутизации рекомендуется использовать свободный режим.

В ПО **Факел** вы можете использовать один из трех режимов <*type*>:

- **strict**: Каждый входящий пакет проверяется по FIB, и если интерфейс не является наилучшим обратным путем, то проверка пакета будет неудачной. По умолчанию неудачные пакеты отбрасываются.
- **loose**: Адрес источника каждого входящего пакета также проверяется по FIB, и если адрес источника не достижим ни через один интерфейс, проверка пакета будет неудачной.
- **disable**: Нет проверки источника

```
set interfaces ethernet <ethN> ipv6 address autoconf
```

Включает механизм получения IPv6-адреса с помощью *SLAAC* (*Stateless Address Auto-configuration*).

> **Примечание**
>
> *Этот метод автоматически отключает пересылку трафика IPv6 на сетевом интерфейсе.*

```
set interfaces ethernet <ethN> ipv6 address eui64 <prefix>
```

Устанавливает префикс EUI-64 на основе MAC адреса <*prefix*> для IPv6 адреса сетевого интерфейса <*ethN*>.

```
set interfaces ethernet <ethN> ipv6 address no-default-link-local
```

Удаляет локальный адрес IPv6 по умолчанию для сетевого интерфейса <*ethN*>.

```
set interfaces ethernet <ethN> ipv6 disable-forwarding
```

Отключает IP переадресацию на сетевом интерфейсе <*ethN*>.

```
set interfaces ethernet <ethN> vrf <vrf>
```

Размещает сетевой интерфейс <*ethN*> в указанном экземпляре VRF <*vrf*>.

Настройка получения адреса по DHCPv6 для Ethernet интерфейса

```
set interfaces ethernet <ethN> dhcp-options client-id <text>
```

Устанавливает идентификатор <*text*>, используемый клиентом для идентификации себя с DHCP-сервером.

```
set interfaces ethernet <ethN> dhcp-options host-name <hostname>
```

Изменяет системное имя устройства <*hostname*>, передаваемое DHCP серверу.

- `set interfaces ethernet <ethN> dhcp-options vendor-class-id <text>`

Определите тип клиента производителя <*text*> для DHCP сервера.

Опция *vendor-class-id* может быть использована для запроса у сервера определенного класса опций производителя.

- `set interfaces ethernet <ethN> dhcp-options no-default-route`

При активации параметра *no-default-route* у DHCP сервера запрашивается только адрес, но не запрашивается шлюз по умолчанию.

- `set interfaces ethernet <ethN> dhcpv6-options duid <duid>`

Задает уникальный идентификатор DHCP (DUID) для сетевого интерфейса <*ethN*>.

Уникальный идентификатор DHCP (DUID) используется клиентом для получения IP-адреса от сервера DHCPv6. Он имеет 2-байтовое поле типа DUID и поле идентификатора переменной длины до 128 байт. Его фактическая длина зависит от типа. Сервер сравнивает DUID со своей базой данных и выдает клиенту данные (адрес, время аренды, DNS-серверы и т. д.).

- `set interfaces ethernet <ethN> dhcpv6-options parameters-only`

Опция *parameters-only* определяет, что DHCPv6 должен обмениваться с серверами только информационными настройками. Примером таких параметров является список адресов DNS серверов. Этот параметр полезен, когда клиенту не нужны параметры с сохранением состояния, такие как IPv6 адреса или префиксы.

- `set interfaces ethernet <ethN> dhcpv6-options rapid-commit`

Включает опцию *rapid-commit* для DHCPv6 сервера.

- `set interfaces ethernet <ethN> dhcpv6-options temporary`

При активации параметра *temporary* устройство запрашивает только временный адрес и не создавать партнерство *IA-NA* (*Identity Association for Non-temporary Addresses*).

Настройка делегирования DHCPv6 префикса для Ethernet интерфейса

- `set interfaces ethernet <ethN> dhcpv6 options pd <id> length <length>`

Задает размер делегируемого префикса <*length*> для экземпляра делегирования префикса <*id*> на сетевом интерфейсе <*ethN*>.

Некоторые провайдеры по умолчанию делегируют только префикс /64. Чтобы запросить определенный размер префикса, используйте эту опцию для запроса большего делегирования для данного экземпляра делегирования префикса **<id>**. Это значение находится в диапазоне от 32 до 64, поэтому вы можете запросить делегирование от /32 префикса (если ваш провайдер позволяет это сделать) до /64. Значение по умолчанию соответствует 64.

- `set interfaces ethernet <ethN> dhcpv6 options pd <id> interface <delegatee> address <address>`

Определяет адрес интерфейса <*address*>, используемый локально на интерфейсе <*delegatee*>, на который был делегирован префикс.

Он будет объединен с делегированным префиксом и *sla-id* для формирования полного адреса интерфейса. По умолчанию используется EUI-64 адрес интерфейса.

- `set interfaces ethernet <ethN> dhcpv6-options pd <pd-id> interface <delegatee> sla-id <sla-id>`

Задает значение идентификатора агрегатора уровня сайта SLA <*sla-id*> на интерфейсе <*delegatee*>. ID должен быть десятичным числом, большим 0, которое укладывается в длину идентификаторов SLA.

Настройка Ethernet параметров

- `set interfaces ethernet <ethN> duplex <auto|full|half>`

Определяет в каком режиме дуплекса будет работать сетевой интерфейсе <*ethN*>. По умолчанию определено значение *auto*.

- **auto** - дуплексная настройка интерфейса согласовывается автоматически
- **full** - всегда используется полный дуплекс
- **half** - всегда использовать полудуплексную связь

- `set interfaces ethernet <ethN> speed <speed>`

Определяет на какой скорости <*speed*> будет работать сетевой интерфейс <*ethN*>. По умолчанию определено значение *auto*.

- **auto** - скорость интерфейса согласовывается автоматически
- **10** - 10 MBit/s
- **100** - 100 MBit/s
- **1000** - 1 GBit/s
- **2500** - 2.5 GBit/s

- **5000** - 5 GBit/s
- **10000** - 10 GBit/s
- **25000** - 25 GBit/s
- **40000** - 40 GBit/s
- **50000** - 50 GBit/s
- **100000** - 100 GBit/s

Настройка аппаратной разгрузки

- `set interfaces ethernet <ethN> offload <gro|gso|sg|tso|ufo|rps>`

Определяет тип аппаратной разгрузки *<gro|gso|sg|tso|ufo|rps>* на указанном интерфейсе *<ethN>*.

> **Примечание**
>
> *Для использования TSO/LRO с адаптерами VMXNET3 необходимо включить аппаратную разгрузку типа SG.*

Настройка аутентификации (EAPoL)

EAP over LAN (EAPoL) — это протокол аутентификации сетевых портов, используемый в стандарте IEEE 802.1X (Port Based Network Access Control), разработанный для предоставления общей сетевой подписи для доступа к сетевым ресурсам. EAPoL поставляется с возможностью идентификации. В качестве параметра идентификации автоматически используется MAC адрес интерфейса.

- `set interfaces ethernet <ethN> eapol ca-cert-file <file>`

Добавляет PEM-файл SSL CA x509 *<file>*, используемый для аутентификации удаленной стороны.

- `set interfaces ethernet <ethN> eapol cert-file <file>`

Добавляет файл публичного сертификата SSL/x509 *<file>*, предоставляемый клиентом для аутентификации в системе 802.1x.

- `set interfaces ethernet <ethN> eapol key-file <file>`

Добавляет файл частного сертификата SSL/x509 *<file>*, предоставляемый клиентом для аутентификации в системе 802.1x.

Мониторинг и эксплуатация Ethernet интерфейса

```
show interfaces ethernet
```

Выводит на экран краткую информацию о сетевых интерфейсах.

```
show interfaces ethernet <ethN>
```

Выводит на экран подробную информацию об указанном сетевом интерфейсе <*ethN*>.

```
show interfaces ethernet <ethN> physical
```

Выводит на экран информацию о физических параметрах сетевого интерфейса <*ethN*>.

```
show interfaces ethernet <ethN> physical offload
```

Выводит на экран информацию обо всех типах аппаратной разгрузки для указанного сетевого интерфейса <*ethN*>.

```
show interfaces ethernet <ethN> transceiver
```

Выводит на экран информацию о подключенных к сетевому интерфейсу <*ethN*> трансиверах.

VLAN 802.1q для Ethernet интерфейса

IEEE 802.1q, часто называемый Dot1q, — это сетевой стандарт, поддерживающий виртуальные локальные сети VLAN в сетях Ethernet стандарта IEEE 802.3. Стандарт определяет систему для маркировки кадров Ethernet и сопутствующие процедуры, которые должны использоваться сетевыми устройствами при работе с такими кадрами. Стандарт также содержит положения о схеме приоритезации качества обслуживания, известной как IEEE 802.1p, и определяет протокол регистрации общих атрибутов.

В ПО **Факел** интерфейсы 802.1q представлены как виртуальные интерфейсы, подчиненный физическому интерфейсу. Для обозначения такого интерфейса используется термин *vif*.

Основные настройки VLAN 802.1q для Ethernet интерфейса

```
set interfaces ethernet <ethN> vif <vlan-id>
```

Создает новый интерфейс VLAN на интерфейсе <*ethN*>, используя номер VLAN, указанный в поле <*vlan-id*>.

На одном физическом интерфейсе можно создать несколько интерфейсов VLAN. Диапазон идентификаторов VLAN составляет от 0 до 4094.

Примечание

На Ethernet интерфейсах vif принимаются только пакеты с метками 802.1Q.

- `set interfaces ethernet <ethN> vif <vlan-id> address <address|dhcp|dhcpv6>`

Задает адрес <*address*|*dhcp*|*dhcpv6*> для VLAN <vlan-id> интерфейса.

- address может быть указан несколько раз как IPv4 и/или IPv6-адрес, например, *192.0.2.1/24* и/или *2001:db8::1/64*
- dhcp адрес интерфейса получен по DHCP от DHCP-сервера в данном сегменте.
- dhcpv6 адрес интерфейса получен по протоколу DHCPv6 от сервера DHCPv6 в данном сегменте.

- `set interfaces ethernet <ethN> vif <vlan id> description <text>`

Задает описательный псевдоним <*text*> для VLAN <*vlan-id*> интерфейса. Например, псевдоним используется командой show interfaces или средствами мониторинга на базе SNMP.

- `set interfaces ethernet <ethN> vif <vlan-id> disable`

Отключает VLAN <*vlan-id*> интерфейс. После отключения интерфейс будет переведен в состояние административного отключения (A/D).

- `set interfaces ethernet <ethN> vif <vlan-id> disable-link-detect`

С помощью этой команды можно настроить VLAN <*vlan-id*> интерфейс таким образом, чтобы он не обнаруживал изменений физического состояния канала связи, например, при отсоединении кабеля.

По умолчанию VLAN интерфейсы настроены на обнаружение изменения физического состояния канала.

- `set interfaces ethernet <ethN> vif <vlan-id> mac <address>`

Задает MAC адрес для VLAN <*vlan-id*> интерфейса.

- `set interfaces ethernet <ethN> vif <vlan-id> mtu <mtu>`

Устанавливает размер MTU <*mtu*> для VLAN <*vlan-id*> интерфейса.

- `set interfaces ethernet <ethN> vif <vlan-id> ip arp-cache timeout <1-86400>`

Устанавливает тайм-аут записи в ARP-кэш в секундах <*1-86400*> для VLAN <*vlan-id*> интерфейса. По умолчанию это значение равно 30 секундам.

```
set interfaces ethernet <ethN> vif <vlan-id> ip disable-arp
filter
```

Отключает фильтрацию ARP на VLAN <*vlan-id*> интерфейсе.

Если этот параметр установлен, то ядро может отвечать на запросы ARP адресами с других интерфейсов.

По умолчанию этот параметр отключен. Это позволяет иметь несколько сетевых интерфейсов в одной подсети, и ответ на ARP адреса каждого интерфейса зависит от того, будет ли ядро маршрутизировать пакет с ARP адреса через этот интерфейс. Для работы этой функции необходимо использовать маршрутизацию на основе источника.

```
set interfaces ethernet <ethN> vif <vlan-id> ip disable-
forwarding
```

Отключает переадресацию IP на VLAN <*vlan-id*> интерфейсе.

```
set interfaces ethernet <ethN> vif <vlan-id> ip enable-arp-
accept
```

Включает прием ARP пакетов на VLAN <*vlan-id*> интерфейсе.

Определяет поведение для ARP пакетов, IP записи которых еще не присутствует в ARP таблице. Если настройка включена, то создаются новые записи в ARP таблице.

Как ответы, так и запросы типа *gratuitous arp* будут вызывать обновление ARP таблицы, если эта настройка включена.

Если в ARP таблице уже содержится IP-адрес кадра *gratuitous arp*, то ARP таблица будет обновлена независимо от того, включена или выключена эта настройка.

```
set interfaces ethernet <ethN> vif <vlan-id> ip enable-arp
announce
```

Включает анонсирование ARP пакетов на VLAN <*vlan-id*> интерфейсе.

```
set interfaces ethernet <ethN> vif <vlan-id> ip enable-arp-
ignore
```

Включает игнорирование ARP пакетов на VLAN <*vlan-id*> интерфейсе.

Этот параметр определяет режимы отправки ответов в ответ на полученные ARP запросы, разрешающие локальные целевые IP-адреса:

- Если параметр включен, ответ будет отправлен только в том случае, если целевой IP-адрес является локальным адресом, настроенным на входящем интерфейсе.
- Если параметр отключен (по умолчанию), ответ будет получен для любого локального целевого IP-адреса, настроенного на любом интерфейсе.

```
set interfaces ethernet <ethN> vif <vlan-id> ip enable-proxy-arp
```

Включает прокси ARP на VLAN <*vlan-id*> интерфейсе. Прокси ARP позволяет интерфейсу Ethernet отвечать своим MAC адресом на ARP запросы IP-адресов назначения в подсетях, подключенных к другим интерфейсам системы.

```
set interfaces ethernet <ethN> vif <vlan-id> ip proxy-arp-pvlan
```

Включает прокси ARP частного VLAN на VLAN <*vlan-id*> интерфейсе.

```
set interfaces ethernet <ethN> vif <vlan-id> ip source validation <strict|loose|disable>
```

Включает политику проверки источника по обратному пути, как указано в RFC 3704.

В настоящее время в RFC 3704 рекомендуется включать строгий режим для предотвращения подмены IP-адресов в результате DDos атак. При использовании асимметричной или другой сложной маршрутизации рекомендуется использовать свободный режим.

- **strict**: Каждый входящий пакет проверяется по FIB, и если интерфейс не является наилучшим обратным путем, то проверка пакета будет неудачной. По умолчанию неудачные пакеты отбрасываются
- **loose**: Адрес источника каждого входящего пакета также проверяется по FIB, и если адрес источника не достижим ни через один интерфейс, проверка пакета будет неудачной
- **disable**: Нет проверки источника

```
set interfaces ethernet <ethN> vif <vlan-id> ipv6 address autoconf
```

Включает механизм получения IPv6-адреса с помощью *SLAAC* (*Stateless Address Auto-configuration*).

> **Примечание**
>
> *Этот метод автоматически отключает пересылку трафика IPv6 на сетевом интерфейсе.*

- `set interfaces ethernet <ethN> vif <vlan-id> ipv6 address eui64 <prefix>`

Устанавливает префикс EUI-64 на основе MAC адреса <*prefix*> для IPv6 адреса VLAN <*vlan-id*> интерфейса.

- `set interfaces ethernet <ethN> vif <vlan-id> ipv6 address no-default-link-local`

Удаляет локальный адрес IPv6 по умолчанию для VLAN <*vlan-id*> интерфейса.

- `set interfaces ethernet <ethN> vif <vlan-id> ipv6 disable-forwarding`

Отключает IP переадресацию на VLAN <*vlan-id*> интерфейсе.

- `set interfaces ethernet <ethN> vif <vlan-id> vrf <vrf>`

Размещает VLAN <*vlan-id*> интерфейс в указанном экземпляре VRF <*vrf*>.

Настройка получения адреса по DHCP для интерфейса VLAN 802.1q

- `set interfaces ethernet <ethN> vif <vlan-id> dhcp-options client-id <text>`

Устанавливает идентификатор <*text*>, используемый клиентом для идентификации себя DHCP сервером

- `set interfaces ethernet <ethN> vif <vlan-id> dhcp-options host-name <hostname>`

Изменяет системное имя устройства <*hostname*>, передаваемое DHCP серверу.

- `set interfaces ethernet <ethN> vif <vlan-id> dhcp-options vendor-class-id <vendor-id>`

Определите тип производителя клиента <*vendor-id*> для DHCP сервера.

Опция *vendor-class-id* может быть использована для запроса у сервера определенного класса опций производителя.

- `set interfaces ethernet <ethN> vif <vlan-id> dhcp-options no-default-route`

При активации параметра *no-default-route* у DHCP сервера запрашивается только адрес, но не запрашивается шлюз по умолчанию.

- `set interfaces ethernet <ethN> vif <vlan-id> dhcpv6-options duid <duid>`

Задает уникальный идентификатор DHCP (DUID) для VLAN <*vlan-id*> интерфейса.

Уникальный идентификатор DHCP (DUID) используется клиентом для получения IP-адреса от сервера DHCPv6. Он имеет 2-байтовое поле типа DUID и поле идентификатора переменной длины до 128 байт. Его фактическая длина зависит от типа. Сервер сравнивает DUID со своей базой данных и выдает клиенту данные (адрес, время аренды, DNS-серверы и т. д.).

- `set interfaces ethernet <ethN> vif <vlan-id> dhcpv6-options parameters-only`

Опция *parameters-only* определяет, что DHCPv6 должен обмениваться с серверами только информационными настройками. Примером таких параметров является список адресов DNS серверов. Этот параметр полезен, когда клиенту не нужны параметры с сохранением состояния, такие как IPv6 адреса или префиксы.

- `set interfaces ethernet <ethN> vif <vlan-id> dhcpv6-options rapid-commit`

Включает опцию *rapid-commit*, которая отвечает за получение настроек для VLAN интерфейса от DHCPv6 сервера.

- `set interfaces ethernet <ethN> vif <vlan-id> dhcpv6-options temporary`

При активации параметра *temporary* устройство запрашивает только временный адрес и не создавать партнерство *IANA* (*Identity Association for Non-temporary Addresses*).

Настройка делегирования DHCPv6 префикса для интерфейса VLAN 802.1q

- `set interfaces ethernet <ethN> vif <vlan-id> dhcpv6-options pd <id> length <length>`

Задает размер делегируемого префикса <*length*> для экземпляра делегирования префикса <*id*> на VLAN <*vlan-id*> интерфейсе.

Некоторые провайдеры по умолчанию делегируют только префикс /64. Чтобы запросить определенный размер префикса, используйте эту опцию для запроса большего делегирования для данного экземпляра делегирования префикса **<id>**. Это значение находится в диапазоне от 32 до 64, поэтому вы можете запросить делегирование от /32 префикса (если ваш провайдер позволяет это сделать) до /64. Значение по умолчанию соответствует 64.

- `set interfaces ethernet <ethN> vif <vlan-id> dhcpv6 options pd <id> interface <delegatee> address <address>`

Определяет адрес интерфейса <*address*>, используемый локально на интерфейсе <*delegatee*>, на который был делегирован префикс.

Он будет объединен с делегированным префиксом и *sla-id* для формирования полного адреса интерфейса. По умолчанию используется EUI-64-адрес интерфейса.

```
set interfaces ethernet <ethN> vif <vlan-id> dhcpv6 options pd
<id> interface<delegatee> sla-id <id>
```

Задает значение идентификатора агрегатора уровня сайта SLA <*sla-id*> на интерфейсе <*delegatee*>. ID должен быть десятичным числом, большим 0, которое укладывается в длину идентификаторов SLA.

VLAN 802.1ad для Ethernet интерфейса

IEEE 802.1ad - стандарт для сетей Ethernet. Стандарт 802.1ad был включен в базовый стандарт 802.1q в 2011 году. Эта технология также известна как провайдерский мост или стекированные VLAN.

QinQ позволяет использовать несколько меток VLAN в кадре Ethernet. Вместе эти метки образуют набор меток. В контексте кадра Ethernet кадр QinQ — это кадр, имеющий два заголовка VLAN 802.1q (с двойной меткой).

В ПО **Факел** термины *vif-s* и *vif-c* обозначают используемые теги для сетевого интерфейса.

Внутренняя метка — это метка, которая ближе всего к части полезной нагрузки кадра. Официально она называется *C-TAG* (*customer tag with Ethernet Type = 0x8100*).

Внешняя метка ближе к заголовку Ethernet, ее название *S-TAG* (*service tag with Ethernet Type = 0x88a8*).

Основные настройки VLAN 802.1ad для Ethernet интерфейса

```
set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id>
address <address|dhcp|dhcpv6>
```

Задает адрес <*address* | *dhcp* | *dhcpv6*> для VLAN интерфейса:

- address может быть указан несколько раз как IPv4 и/или IPv6-адрес, например, *192.0.2.1/24* и/или *2001:db8::1/64*
- dhcp адрес интерфейса получен по DHCP от DHCP-сервера в данном сегменте.
- dhcpv6 адрес интерфейса получен по протоколу DHCPv6 от сервера DHCPv6 в данном сегменте.

```
set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id>
description <text>
```

Задает описательный псевдоним <*text*> для VLAN интерфейса. Например, псевдоним используется командой ***show interfaces*** или средствами мониторинга на базе SNMP.

```
set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id>
disable
```

Отключает VLAN интерфейс. После отключения интерфейс будет переведен в состояние административного отключения (A/D).

```
set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-
id> disable-link-detect
```

С помощью этой команды можно настроить VLAN интерфейс таким образом, чтобы он не обнаруживал изменений физического состояния канала связи, например, при отсоединении кабеля.

По умолчанию VLAN интерфейсы настроены на обнаружение изменения физического состояния канала.

```
set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-
id> mac <address>
```

Задает MAC адрес <*address*> для VLAN интерфейса.

```
set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-
id> mtu <mtu>
```

Устанавливает размер MTU <*mtu*> для VLAN интерфейса.

```
set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id>
ip arp-cache-timeout <1-86400>
```

Устанавливает тайм-аут записи в ARP-кэш в секундах <*1-86400*> для VLAN интерфейса. По умолчанию это значение равно 30 секундам.

```
set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id>
ip disable-arp-filter
```

Отключает фильтрацию ARP на VLAN интерфейсе.

Если этот параметр установлен, то ядро может отвечать на запросы ARP адресами с других интерфейсов.

По умолчанию этот параметр отключен. Это позволяет иметь несколько сетевых интерфейсов в одной подсети, и ответ на ARP адреса каждого интерфейса зависит от того, будет ли ядро маршрутизировать пакет с ARP адреса через этот интерфейс. Для работы этой функции необходимо использовать маршрутизацию на основе источника.

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> ip disable-forwarding`

Отключает переадресацию IP на VLAN интерфейсе.

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> ip enable-arp-accept`

Включает прием ARP пакетов на VLAN интерфейсе.

Определяет поведение для ARP пакетов, IP записи которых еще не присутствует в ARP таблице. Если настройка включена, то создаются новые записи в ARP таблице.

Как ответы, так и запросы типа *gratuitous arp* будут вызывать обновление ARP таблицы, если эта настройка включена.

Если в ARP таблице уже содержится IP-адрес кадра *gratuitous arp*, то ARP таблица будет обновлена независимо от того, включена или выключена эта настройка.

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> ip enable-arp-announce`

Включает анонсирование ARP пакетов на VLAN интерфейсе.

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> ip enable-arp-ignore`

Включает игнорирование ARP пакетов на VLAN интерфейсе.

Этот параметр определяет режимы отправки ответов в ответ на полученные ARP запросы, разрешающие локальные целевые IP-адреса:

- Если параметр включен, ответ будет отправлен только в том случае, если целевой IP-адрес является локальным адресом, настроенным на входящем интерфейсе.
- Если параметр отключен (по умолчанию), ответ будет получен для любого локального целевого IP-адреса, настроенного на любом интерфейсе.

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> ip enable-proxy-arp`

Включает прокси ARP на VLAN интерфейсе. Прокси ARP позволяет интерфейсу Ethernet отвечать своим MAC адресом на ARP запросы IP-адресов назначения в подсетях, подключенных к другим интерфейсам системы.

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> ip proxy-arp-pvlan`

Включает прокси ARP частного VLAN на VLAN интерфейсе.

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> ip source-validation <strict|loose|disable>`

Включает политику проверки источника по обратному пути, как указано в RFC 3704.

В настоящее время в RFC 3704 рекомендуется включать строгий режим для предотвращения подмены IP-адресов в результате DDos атак. При использовании асимметричной или другой сложной маршрутизации рекомендуется использовать свободный режим.

- **strict**: Каждый входящий пакет проверяется по FIB, и если интерфейс не является наилучшим обратным путем, то проверка пакета будет неудачной. По умолчанию неудачные пакеты отбрасываются.
- **loose**: Адрес источника каждого входящего пакета также проверяется по FIB, и если адрес источника не достижим ни через один интерфейс, проверка пакета будет неудачной.
- **disable**: Нет проверки источника

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> ipv6 addressautoconf`

Включает механизм получения IPv6-адреса с помощью *SLAAC* (*Stateless Address Auto-configuration*).

> **Примечание**
>
> *Этот метод автоматически отключает пересылку трафика IPv6 на сетевом интерфейсе.*

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> ipv6 addresseui64 <prefix>`

Устанавливает префикс EUI-64 на основе MAC адреса *<prefix>* для IPv6 адреса VLAN интерфейса.

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> ipv6 address no-default-link-local`

Удаляет локальный адрес IPv6 по умолчанию для VLAN интерфейса.

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> ipv6 disable-forwarding`

Отключает IP переадресацию на VLAN интерфейсе.

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> vrf <vrf>`

Размещает VLAN интерфейс в указанном экземпляре VRF <*vrf*>.

Настройка получения адреса по DHCP для интерфейса VLAN 802.1ad

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> dhcp-options client-id <text>`

Устанавливает идентификатор <*text*>, используемый клиентом для идентификации себя DHCP сервером.

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> dhcp-options host-name <hostname>`

Изменяет системное имя устройства <*hostname*>, передаваемое DHCP серверу.

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> dhcp-options vendor-class-id <vendor-id>`

Определите тип производителя клиента <*vendor-id*> для DHCP сервера.

Опция *vendor-class-id* может быть использована для запроса у сервера определенного класса опций производителя.

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> dhcp-options no-default-route`

При активации параметра *no-default-route* у DHCP сервера запрашивается только адрес, но не запрашивается шлюз по умолчанию.

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> dhcpv6-optionsduid <duid>`

Задает уникальный идентификатор DHCP (DUID) для VLAN интерфейса.

Уникальный идентификатор DHCP (DUID) используется клиентом для получения IP-адреса от сервера DHCPv6. Он имеет 2-байтовое поле типа DUID и поле идентификатора переменной длины до 128 байт. Его фактическая длина зависит от типа. Сервер сравнивает DUID со своей базой данных и выдает клиенту данные (адрес, время аренды, DNS-серверы и т. д.).

- `set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> dhcpv6-optionsparameters-only`

Опция *parameters-only* определяет, что DHCPv6 должен обмениваться с серверами только информационными настройками. Примером таких параметров является список

адресов DNS серверов. Этот параметр полезен, когда клиенту не нужны параметры с сохранением состояния, такие как IPv6 адреса или префиксы.

```
set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> dhcpv6-optionsrapid-commit
```

Включает опцию *rapid-commit*, которая отвечает за получение настроек для VLAN интерфейса от DHCPv6 сервера.

```
set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> dhcpv6-optionstemporary
```

При активации параметра *temporary* устройство запрашивает только временный адрес и не создавать партнерство *IANA* (*Identity Association for Non-temporary Addresses*).

Настройка делегирования DHCPv6 префикса для интерфейса VLAN 802.1ad

```
set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> dhcpv6-optionspd <id> length <length>
```

Задает размер делегируемого префикса <*length*> для экземпляра делегирования префикса <*id*> на VLAN интерфейсе.

Некоторые провайдеры по умолчанию делегируют только префикс /64. Чтобы запросить определенный размер префикса, используйте эту опцию для запроса большего делегирования для данного экземпляра делегирования префикса <*id*>. Это значение находится в диапазоне от 32 до 64, поэтому вы можете запросить делегирование от /32 префикса (если ваш провайдер позволяет это сделать) до /64. Значение по умолчанию соответствует 64.

```
set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> dhcpv6-
optionspd <id> interface <delegatee> address <address>
```

Определяет адрес интерфейса <*address*>, используемый локально на интерфейсе <*delegatee*>, на который был делегирован префикс.

Он будет объединен с делегированным префиксом и *sla-id* для формирования полного адреса интерфейса. По умолчанию используется EUI-64 адрес интерфейса.

```
set interfaces ethernet <bondN> vif-s <vlan-id> vif-c <vlan-id> dhcpv6-optionspd <id> interface <delegatee> sla-id <id>
```

Задает значение идентификатора агрегатора уровня сайта SLA <*sla-id*> на интерфейсе <*delegatee*>. ID должен быть десятичным числом, большим 0, которое укладывается в длину идентификаторов SLA.

Зеркалирование Ethernet интерфейса

Зеркалирование интерфейсов позволяет копировать входящий и исходящий трафик на указанный интерфейс, обычно подключенный к специальному оборудованию (система контроля поведения, система обнаружения вторжений или коллектор трафика) и копировать весь соответствующий трафик, полученный на этом интерфейсе. Преимущество зеркалирования трафика заключается в том, что устройства обрабатывают копию сетевого трафика, поэтому это не влияет на работу корпоративных сервисов и локальной сети организации.

Для настройки зеркалирования портов в ПО **Факел** используется опция *mirror*. Зеркалированный порт должен быть настроены для входящего и исходящего трафика.

Основные настройки зеркалирования Ethernet интерфейса

- `set interfaces ethernet <ethN> mirror ingress <monitor-interface>`

Настраивает передачу копии входящего трафика от интерфейса <*ethN*> на интерфейс <*monitor-interface*>.

- `set interfaces ethernet <ethN> mirror egress <monitor-interface>`

Настраивает передачу копии исходящего трафика от интерфейса <*ethN*> на интерфейс <*monitor-interface*>.

Агрегированный интерфейс

Общая информация об агрегированном интерфейса

Механизм агрегирования каналов (*Link Aggregation*) объединяет нескольких сетевых интерфейсов в один логический (агрегированный) интерфейс. Поведение связанных интерфейсов зависит от режима работ. В общем случае режимы обеспечивают либо горячее резервирование, либо балансировку нагрузки. Кроме того, может осуществляться мониторинг состояния канала.

Основные настройки агрегированного интерфейса

- `set interfaces bonding <bondN> member interface <member>`

Добавляет сетевой интерфейс <*member*> в состав агрегированного интерфейса <*bondN*>.

- `set interfaces bonding <bondN> address <address|dhcp|dhcpv6>`

Задает адрес <*address|dhcp|dhcpv6*> для сетевого интерфейса <*bondN*>.

- адрес может быть указан несколько раз как IPv4 и/или IPv6 адрес, например, *192.0.2.1/24* и/или *2001:db8::1/64*
- адрес интерфейса dhcp принимается по протоколу DHCP от DHCP-сервера данного сегмента.
- dhcpv6 адрес интерфейса получен DHCPv6 от DHCPv6-сервера на данном сегменте.

```
set interfaces bonding <bondN> description <text>
```

Задает псевдоним <*text*> для сетевого интерфейса <*bondN*>. Например, псевдоним может использоваться командой ***show interfaces*** или средствами мониторинга на базе SNMP.

```
set interfaces bonding <bondN> disable
```

Отключает сетевой интерфейс <*bondN*>. После отключения интерфейс будет переведен в состояние административного отключения (A/D).

```
set interfaces bonding <bondN> disable-flow-control
```

Отключает генерацию управления потоком Ethernet (кадров паузы).

Управление потоком Ethernet — это механизм временного прекращения передачи данных в компьютерных сетях семейства Ethernet. Цель этого механизма - обеспечить нулевую потерю пакетов в условиях перегрузки сети.

Станция отправитель (компьютер или сетевой коммутатор) может передавать данные быстрее, чем другой конец канала связи может их принять. Используя управление потоком, принимающая станция может подать сигнал отправителю с просьбой приостановить передачу до тех пор, пока получатель не поймает его.

```
set interfaces bonding <bondN> disable-link-detect
```

С помощью этой команды можно настроить сетевой интерфейс таким образом, чтобы он не обнаруживал изменений физического состояния канала связи, например, при отсоединении кабеля.

По умолчанию сетевые интерфейсы настроены на обнаружение изменения физического состояния канала.

```
set interfaces bonding <bondN> mac <address>
```

Задает MAC адрес для сетевого интерфейса <*bondN*>.

```
set interfaces bonding <bondN> mtu <mtu>
```

Устанавливает размер MTU <*mtu*> для сетевого интерфейса <*bondN*>.

- `set interfaces bonding <bondN> ip arp-cache-timeout <1-86400>`

Устанавливает тайм-аут записи в ARP-кэш в секундах *<1-86400>* для сетевого интерфейса *<bondN>*. По умолчанию это значение равно 30 секундам.

- `set interfaces bonding <bondN> ip disable-arp-filter`

Отключает фильтрацию ARP на сетевом интерфейсе *<bondN>*.

Если этот параметр установлен, то ядро может отвечать на запросы ARP адресами с других интерфейсов.

По умолчанию этот параметр отключен. Это позволяет иметь несколько сетевых интерфейсов в одной подсети, и ответ на ARP адреса каждого интерфейса зависит от того, будет ли ядро маршрутизировать пакет с ARP адреса через этот интерфейс. Для работы этой функции необходимо использовать маршрутизацию на основе источника.

- `set interfaces bonding <bondN> ip disable-forwarding`

Отключает переадресацию IP на сетевом интерфейсе *<bondN>*.

- `set interfaces bonding <bondN> ip enable-arp-accept`

Включает прием ARP пакетов на сетевом интерфейсе *<bondN>*.

Определяет поведение для ARP пакетов, IP записи которых еще не присутствует в ARP таблице. Если настройка включена, то создаются новые записи в ARP таблице.

Как ответы, так и запросы типа *gratuitous arp* будут вызывать обновление ARP таблицы, если эта настройка включена.

Если в ARP таблице уже содержится IP-адрес кадра *gratuitous arp*, то ARP таблица будет обновлена независимо от того, включена или выключена эта настройка.

- `set interfaces bonding <bondN> ip enable-arp-announce`

Включает анонсирование ARP пакетов на сетевом интерфейсе <bondN>.

- `set interfaces bonding <bondN> ip enable-arp-ignore`

Включает игнорирование ARP пакетов на сетевом интерфейсе *<bondN>*.

Этот параметр определяет режимы отправки ответов в ответ на полученные ARP запросы, разрешающие локальные целевые IP-адреса:

- Если параметр включен, ответ будет отправлен только в том случае, если целевой IP-адрес является локальным адресом, настроенным на входящем интерфейсе.

- Если параметр отключен (по умолчанию), ответ будет получен для любого локального целевого IP-адреса, настроенного на любом интерфейсе.

```
set interfaces bonding <bondN> ip enable-proxy-arp
```

Включает прокси ARP на сетевом интерфейсе <*bondN*>. Прокси ARP позволяет интерфейсу Ethernet отвечать своим MAC адресом на ARP запросы IP-адресов назначения в подсетях, подключенных к другим интерфейсам системы.

```
set interfaces bonding <bondN> ip proxy-arp-pvlan
```

Включает прокси ARP частного VLAN на сетевом интерфейсе <*bondN*>.

```
set interfaces bonding <bondN> ip source-validation <strict|
loose|disable>
```

Включает политику проверки источника по обратному пути, как указано в RFC 3704.

В настоящее время в RFC 3704 рекомендуется включать строгий режим для предотвращения подмены IP-адресов в результате DDos атак. При использовании асимметричной или другой сложной маршрутизации рекомендуется использовать свободный режим.

- **strict**: Каждый входящий пакет проверяется по FIB, и если интерфейс не является наилучшим обратным путем, то проверка пакета будет неудачной. По умолчанию неудачные пакеты отбрасываются
- **loose**: Адрес источника каждого входящего пакета также проверяется по FIB, и если адрес источника не достижим ни через один интерфейс, проверка пакета будет неудачной
- **disable**: Нет проверки источника

```
set interfaces bonding <bondN> ipv6 address autoconf
```

Включает механизм получения IPv6-адреса с помощью *SLAAC* (*Stateless Address Auto-configuration*).

> **Примечание**
>
> *Этот метод автоматически отключает пересылку трафика IPv6 на сетевом интерфейсе.*

```
set interfaces bonding <bondN> ipv6 address eui64 <prefix>
```

Устанавливает префикс EUI-64 на основе MAC адреса <*prefix*> для IPv6 адреса сетевого интерфейса <*bondN*>.

- `set interfaces bonding <bondN> ipv6 address no-default-link-local`

Удаляет локальный адрес IPv6 по умолчанию для сетевого интерфейса <*bondN*>.

- `set interfaces bonding <bondN> ipv6 disable-forwarding`

Отключает IP переадресацию на сетевом интерфейсе <*bondN*>.

- `set interfaces bonding <bondN> vrf <vrf>`

Размещает сетевой интерфейс <bondN> в указанном экземпляре VRF <vrf>.

Настройка получения адреса по DHCPv6 для агрегированного интерфейса

- `set interfaces bonding <bondN> dhcp-options client-id <text>`

Устанавливает идентификатор <*text*>, используемый клиентом для идентификации себя DHCP сервером.

- `set interfaces bonding <bondN> dhcp-options host-name <hostname>`

Изменяет системное имя устройства <*hostname*>, передаваемое DHCP серверу.

- `set interfaces bonding <bondN> dhcp-options vendor-class-id <vendor-id>`

Определите тип клиента производителя <*vendor-id*> для DHCP сервера.

Опция *vendor-class-id* может быть использована для запроса у сервера определенного класса опций производителя.

- `set interfaces bonding <bondN> dhcp-options no-default-route`

При активации параметра *no-default-route* у DHCP сервера запрашивается только адрес, но не запрашивается шлюз по умолчанию.

- `set interfaces bonding <bondN> dhcpv6-options duid <duid>`

Задает уникальный идентификатор DHCP (DUID) для сетевого интерфейса <*bondN*>.

Уникальный идентификатор DHCP (DUID) используется клиентом для получения IP-адреса от сервера DHCPv6. Он имеет 2-байтовое поле типа DUID и поле идентификатора переменной длины до 128 байт. Его фактическая длина зависит от типа. Сервер сравнивает DUID со своей базой данных и выдает клиенту данные (адрес, время аренды, DNS-серверы и т. д.).

```
set interfaces bonding <bondN> dhcpv6-options parameters-only
```

Опция *parameters-only* определяет, что DHCPv6 должен обмениваться с серверами только информационными настройками. Примером таких параметров является список адресов DNS серверов. Этот параметр полезен, когда клиенту не нужны параметры с сохранением состояния, такие как IPv6 адреса или префиксы.

```
set interfaces bonding <bondN> dhcpv6-options rapid-commit
```

Включает опцию *rapid-commit*, которая отвечает за получение настроек для VLAN интерфейса от DHCPv6 сервера.

```
set interfaces bonding <bondN> dhcpv6-options temporary
```

При активации параметра *temporary* устройство запрашивает только временный адрес и не создавать партнерство *IANA* (*Identity Association for Non-temporary Addresses*).

Настройка делегирования DHCPv6 префикса для агрегированного интерфейса

```
set interfaces bonding <bondN> dhcpv6-options pd <id> length
<length>
```

Задает размер делегируемого префикса <*length*> для экземпляра делегирования префикса <*id*> на сетевом интерфейсе <*bondN*>.

Некоторые провайдеры по умолчанию делегируют только префикс /64. Чтобы запросить определенный размер префикса, используйте эту опцию для запроса большего делегирования для данного экземпляра делегирования префикса **<id>**. Это значение находится в диапазоне от 32 до 64, поэтому вы можете запросить делегирование от /32 префикса (если ваш провайдер позволяет это сделать) до /64. Значение по умолчанию соответствует 64.

```
set interfaces bonding <bondN> dhcpv6-options pd <id> interface
<delegatee> address <address>
```

Определяет адрес интерфейса <*address*>, используемый локально на интерфейсе <*delegatee*>, на который был делегирован префикс.

Он будет объединен с делегированным префиксом и *sla-id* для формирования полного адреса интерфейса. По умолчанию используется EUI-64 адрес интерфейса.

```
set interfaces bonding <bondN> dhcpv6-options pd <id> interface
<delegatee> sla-id <id>
```

Задает значение идентификатора агрегатора уровня сайта SLA <*sla-id*> на интерфейсе <*delegatee*>. ID должен быть десятичным числом, большим 0, которое укладывается в длину идентификаторов SLA.

Настройка параметров для агрегированного интерфейса

- `set interfaces bonding <bondN> mode <802.3ad|active-backup|broadcast|round-robin|transmit-load-balance|adaptive-load-balance|xor-hash>`

Определяет режим работы агрегированного интерфейса <*bondN*>.

- `set interfaces bonding <bondN> min-links <0-16>`

Определяет минимальное количество сетевых интерфейсов <*0-16*>, входящих в состав агрегированного интерфейса <*bondN*>. Значение по умолчанию равно 0.

Этот параметр активен только при режим работы 802.3ad.

> **Примечание**
>
> *Поскольку агрегированный канал не может быть активным без хотя бы одного доступного канала связи, установка этого параметра в 0 или в 1 имеет совершенно одинаковый эффект.*

- `set interfaces bonding <bondN> lacp-rate <slow|fast>`

Определяет скорость передачи LPDU пакетов <*slow|fast*> между участками внутри агрегированного интерфейса <*bondN*>. По умолчанию используется значение slow.

Данная опция поддерживается только при работе агрегированного интерфейса в режиме 802.3ad.

- **slow**: запрашивать партнера о передаче LACPDU каждые 30 секунд
- **fast**: запрашивать партнера о передаче LACPDU каждые 1 секунд

- `set interfaces bonding <bondN> hash-policy <policy>`

Определяет политику хеширования <*policy*> для агрегированного интерфейса <*bondN*>.

- `set interfaces bonding <bondN> primary <primary-interface>`

Определяет основной интерфейс <*primary-interface*> среди всех интерфейсов, входящих в состав агрегированного интерфейса <*bondN*>.

Этот параметр актуален для следующих режимов работы агрегированного интерфейса: *active-backup*, *transmit-load-balance*, и *adaptive-load-balance*.

- `set interfaces bonding <bondN> arp-monitor interval <time>`

Задает интервал ARP мониторинга <*time*> для агрегированного интерфейса <*bondN*>. По умолчанию значение интервала равно 0.

Значение 0 отключает мониторинг ARP.

- `set interfaces bonding <bondN> arp-monitor target <address>`

Определяет IP-адреса <*address*>, которые будут использоваться в качестве пиров для ARP мониторинга, если значение параметра *arp-monitor interval* больше 0. На эти адреса отправляется ARP запрос для определения работоспособности интерфейса. Можно указать несколько целевых IP-адресов. Для работы ARP мониторинга должен быть указан хотя бы один IP-адрес.

Максимальное количество адресов, которое может быть указано равно 16. Значение по умолчанию - не указано ни одного IP-адреса.

Мониторинг и эксплуатация агрегированного интерфейса

- `show interfaces bonding`

Выводит на экран краткую информацию об агрегированных интерфейсах.

- `show interfaces bonding <bondN>`

Выводит на экран информацию об указанном агрегированном интерфейсе <*bondN*>.

- `show interfaces bonding <bondN> detail`

Выводит на экран подробную информацию об указанном агрегированном интерфейсе <*bondN*>.

VLAN для агрегированного интерфейса

IEEE 802.1q, часто называемый Dot1q, — это сетевой стандарт, поддерживающий виртуальные локальные сети VLAN в сетях Ethernet стандарта IEEE 802.3. Стандарт определяет систему для маркировки кадров Ethernet и сопутствующие процедуры, которые должны использоваться сетевыми устройствами при работе с такими кадрами. Стандарт также содержит положения о схеме приоритизации качества обслуживания, известной как IEEE 802.1p, и определяет протокол регистрации общих атрибутов.

В ПО **Факел** интерфейсы 802.1q представлены как виртуальные интерфейсы, подчиненные физическому интерфейсу. Для обозначения такого интерфейса используется термин *vif*.

Основные настройки VLAN для агрегированного интерфейса

- `set interfaces bonding <bondN> vif <vlan-id>`

Создает новый интерфейс VLAN на интерфейсе <*bondN*>, используя номер VLAN, указанный в поле <*vlan-id*>.

На одном физическом интерфейсе можно создать несколько интерфейсов VLAN. Диапазон идентификаторов VLAN составляет от 0 до 4094.

> **Примечание**
>
> *На Ethernet интерфейсах vif принимаются только пакеты с метками 802.1Q.*

- `set interfaces bonding <bondN> vif <vlan-id> address <address|dhcp|dhcpv6>`

Задает адрес <*address*|*dhcp*|*dhcpv6*> для VLAN <*vlan-id*> интерфейса:

- address может быть указан несколько раз как IPv4 и/или IPv6-адрес, например, *192.0.2.1/24* и/или *2001:db8::1/64*
- dhcp адрес интерфейса получен по DHCP от DHCP-сервера в данном сегменте.
- dhcpv6 адрес интерфейса получен по протоколу DHCPv6 от сервера DHCPv6 в данном сегменте.

- `set interfaces bonding <bondN> vif <vlan-id> description <text>`

Задает описательный псевдоним <*text*> для VLAN <*vlan-id*> интерфейса. Например, псевдоним используется командой ***show interfaces*** или средствами мониторинга на базе SNMP.

- `set interfaces bonding <bondN> vif <vlan-id> disable`

Отключает VLAN <*vlan-id*> интерфейс. После отключения интерфейс будет переведен в состояние административного отключения (A/D).

- `set interfaces bonding <bondN> vif <vlan-id> disable-link-detect`

С помощью этой команды можно настроить VLAN <*vlan-id*> интерфейс таким образом, чтобы он не обнаруживал изменений физического состояния канала связи, например, при отсоединении кабеля.

По умолчанию VLAN интерфейсы настроены на обнаружение изменения физического состояния канала.

- `set interfaces bonding <bondN> vif <vlan-id> mac <address>`

Задает MAC адрес для VLAN <*vlan-id*> интерфейса.

- `set interfaces bonding <bondN> vif <vlan-id> mtu <mtu>`

Устанавливает размер MTU <*mtu*> для VLAN <*vlan-id*> интерфейса.

- `set interfaces bonding <bondN> vif <vlan-id> ip arp-cache-timeout <1-86400>`

Устанавливает тайм-аут записи в ARP-кэш в секундах <*1-86400*> для VLAN <*vlan-id*> интерфейса. По умолчанию это значение равно 30 секундам.

- `set interfaces bonding <bondN> vif <vlan-id> ip disable-arp-filter`

Отключает фильтрацию ARP на VLAN <*vlan-id*> интерфейсе.

Если этот параметр установлен, то ядро может отвечать на запросы ARP адресами с других интерфейсов.

По умолчанию этот параметр отключен. Это позволяет иметь несколько сетевых интерфейсов в одной подсети, и ответ на ARP адреса каждого интерфейса зависит от того, будет ли ядро маршрутизировать пакет с ARP адреса через этот интерфейс. Для работы этой функции необходимо использовать маршрутизацию на основе источника.

- `set interfaces bonding <bondN> vif <vlan-id> ip disable-forwarding`

Отключает переадресацию IP на VLAN <*vlan-id*> интерфейсе.

- `set interfaces bonding <bondN> vif <vlan-id> ip enable-arp-accept`

Включает прием ARP пакетов на VLAN <*vlan-id*> интерфейсе.

Определяет поведение для ARP пакетов, IP записи которых еще не присутствует в ARP таблице. Если настройка включена, то создаются новые записи в ARP таблице.

Как ответы, так и запросы типа *gratuitous arp* будут вызывать обновление ARP таблицы, если эта настройка включена.

Если в ARP таблице уже содержится IP-адрес кадра *gratuitous arp*, то ARP таблица будет обновлена независимо от того, включена или выключена эта настройка.

- `set interfaces bonding <bondN> vif <vlan-id> ip enable-arp-announce`

Включает анонсирование ARP пакетов на VLAN <*vlan-id*> интерфейсе.

- `set interfaces bonding <bondN> vif <vlan-id> ip enable-arp-ignore`

Включает игнорирование ARP пакетов на VLAN <*vlan-id*> интерфейсе.

Этот параметр определяет режимы отправки ответов в ответ на полученные ARP запросы, разрешающие локальные целевые IP-адреса:

- Если параметр включен, ответ будет отправлен только в том случае, если целевой IP-адрес является локальным адресом, настроенным на входящем интерфейсе.
- Если параметр отключен (по умолчанию), ответ будет получен для любого локального целевого IP-адреса, настроенного на любом интерфейсе.

```
set interfaces bonding <bondN> vif <vlan-id> ip enable-proxy-arp
```

Включает прокси ARP на VLAN <*vlan-id*> интерфейсе. Прокси ARP позволяет интерфейсу Ethernet отвечать своим MAC адресом на ARP запросы IP-адресов назначения в подсетях, подключенных к другим интерфейсам системы.

```
set interfaces bonding <bondN> vif <vlan-id> ip proxy-arp-pvlan
```

Включает прокси ARP частного VLAN на VLAN <*vlan-id*> интерфейсе.

```
set interfaces bonding <bondN> vif <vlan-id> ip source-validation <strict|loose|disable>
```

Включает политику проверки источника по обратному пути, как указано в RFC 3704.

В настоящее время в RFC 3704 рекомендуется включать строгий режим для предотвращения подмены IP-адресов в результате DDos атак. При использовании асимметричной или другой сложной маршрутизации рекомендуется использовать свободный режим.

- **strict**: Каждый входящий пакет проверяется по FIB, и если интерфейс не является наилучшим обратным путем, то проверка пакета будет неудачной. По умолчанию неудачные пакеты отбрасываются
- **loose**: Адрес источника каждого входящего пакета также проверяется по FIB, и если адрес источника не достижим ни через один интерфейс, проверка пакета будет неудачной
- **disable**: Нет проверки источника

```
set interfaces bonding <bondN> vif <vlan-id> ipv6 address autoconf
```

Включает механизм получения IPv6-адреса с помощью *SLAAC* (*Stateless Address Auto-configuration*).

Примечание

Этот метод автоматически отключает пересылку трафика IPv6 на сетевом интерфейсе.

- `set interfaces bonding <bondN> vif <vlan-id> ipv6 address eui64 <prefix>`

Устанавливает префикс EUI-64 на основе MAC адреса <*prefix*> для IPv6 адреса VLAN <*vlan-id*> интерфейса.

- `set interfaces bonding <bondN> vif <vlan-id> ipv6 address no-default-link-local`

Удаляет локальный адрес IPv6 по умолчанию для VLAN <*vlan-id*> интерфейса.

- `set interfaces bonding <bondN> vif <vlan-id> ipv6 disable-forwarding`

Отключает IP переадресацию на VLAN <*vlan-id*> интерфейсе.

- `set interfaces bonding <bondN> vif <vlan-id> vrf <vrf>`

Размещает VLAN <*vlan-id*> интерфейс в указанном экземпляре VRF <*vrf*>.

Настройка получения адреса по DHCP для VLAN интерфейса

- `set interfaces bonding <bondN> vif <vlan-id> dhcp-options client-id <text>`

Устанавливает идентификатор <*text*>, используемый клиентом для идентификации себя DHCP сервером

- `set interfaces bonding <bondN> vif <vlan-id> dhcp-options host-name <hostname>`

Изменяет системное имя устройства <*hostname*>, передаваемое DHCP серверу.

- `set interfaces bonding <bondN> vif <vlan-id> dhcp-options vendor-class-id <vendor-id>`

Определите тип производителя клиента <*vendor-id*> для DHCP сервера.

Опция *vendor-class-id* может быть использована для запроса у сервера определенного класса опций поставщика.

- `set interfaces bonding <bondN> vif <vlan-id> dhcp-options no-default-route`

При активации параметра *no-default-route* у DHCP сервера запрашивается только адрес, но не запрашивается шлюз по умолчанию.

- `set interfaces bonding <bondN> vif <vlan-id> dhcpv6-options duid <duid>`

Задает уникальный идентификатор DHCP (DUID) для VLAN <*vlan-id*> интерфейса.

Уникальный идентификатор DHCP (DUID) используется клиентом для получения IP-адреса от сервера DHCPv6. Он имеет 2-байтовое поле типа DUID и поле идентификатора переменной длины до 128 байт. Его фактическая длина зависит от типа. Сервер сравнивает DUID со своей базой данных и выдает клиенту данные (адрес, время аренды, DNS-серверы и т. д.).

- `set interfaces bonding <bondN> vif <vlan-id> dhcpv6-options parameters-only`

Опция *parameters-only* определяет, что DHCPv6 должен обмениваться с серверами только информационными настройками. Примером таких параметров является список адресов DNS серверов. Этот параметр полезен, когда клиенту не нужны параметры с сохранением состояния, такие как IPv6 адреса или префиксы.

- `set interfaces bonding <bondN> vif <vlan-id> dhcpv6-options rapid-commit`

Включает опцию *rapid-commit*, которая отвечает за получение настроек для VLAN интерфейса от DHCPv6 сервера.

- `set interfaces bonding <bondN> vif <vlan-id> dhcpv6-options temporary`

При активации параметра *temporary* устройство запрашивает только временный адрес и не создавать партнерство *IANA* (*Identity Association for Non-temporary Addresses*).

Настройка делегирования DHCPv6 префикса для VLAN интерфейса

- `set interfaces bonding <bondN> vif <vlan-id> dhcpv6-options pd <id> length<length>`

Задает размер делегируемого префикса <length> для экземпляра делегирования префикса <*id*> на VLAN <*vlan-id*> интерфейсе.

Некоторые провайдеры по умолчанию делегируют только префикс /64. Чтобы запросить определенный размер префикса, используйте эту опцию для запроса большего делегирования для данного экземпляра делегирования префикса <*id*>. Это значение

находится в диапазоне от 32 до 64, поэтому вы можете запросить делегирование от /32 префикса (если ваш провайдер позволяет это сделать) до /64. Значение по умолчанию соответствует 64.

```
set interfaces bonding <bondN> vif <vlan-id> dhcpv6-options pd
<id> interface <delegatee> address <address>
```

Определяет адрес интерфейса <*address*>, используемый локально на интерфейсе <*delegatee*>, на который был делегирован префикс.

Он будет объединен с делегированным префиксом и *sla-id* для формирования полного адреса интерфейса. По умолчанию используется EUI-64 адрес интерфейса.

```
set interfaces bonding <bondN> vif <vlan-id> dhcpv6-options pd
<id> interface <delegatee> sla-id <id>
```

Задает значение идентификатора агрегатора уровня сайта SLA <*sla-id*> на интерфейсе <*delegatee*>. ID должен быть десятичным числом, большим 0, которое укладывается в длину идентификаторов SLA.

Зеркалирование агрегированного интерфейс

Зеркалирование интерфейсов позволяет копировать входящий и исходящий трафик на указанный интерфейс, обычно подключенный к специальному оборудованию (система контроля поведения, система обнаружения вторжений или коллектор трафика) и копировать весь соответствующий трафик, полученный на этом интерфейсе. Преимущество зеркалирования трафика заключается в том, что устройства обрабатывают копию сетевого трафика, поэтому это не влияет на работу корпоративных сервисов и локальной сети организации.

Для настройки зеркалирования портов в ПО **Факел** используется опция *mirror*. Зеркалированный порт должен быть настроены для входящего и исходящего трафика.

Основные настройки зеркалирования агрегированного интерфейса

```
set interfaces bonding <bondN> mirror ingress <monitor-
interface>
```

Настраивает передачу копии входящего трафика от интерфейса <*bondN*> на интерфейс <*monitor-interface*>.

```
set interfaces bonding <bond> mirror egress <monitor-interface>
```

Настраивает передачу копии исходящего трафика от интерфейса <*bondN*> на интерфейс <*monitor-interface*>.

Интерфейс сетевого моста

Общая информация об интерфейсе сетевого моста

Сетевой мост соединяет независимые друг от друга сетевые сегменты, чтобы обеспечить связь между ними и позволить им работать как единая сеть. Передача пакетов осуществляется на основе MAC-адреса, а не IP-адреса. Поскольку пересылка осуществляется на втором уровне, все протоколы могут проходить через мост прозрачно.

Пример настройки интерфейса сетевого моста

Пример настройки интерфейса сетевого моста. Для настройки сетевого моста будут использоваться следующие параметры:

- *br100* - имя интерфейса сетевого моста;
- *eth1* и *eth2.10* - интерфейсы, входящие в состав интерфейса сетевого моста *br100*;
- Протокол STP на интерфейсе сетевого моста *br100*;
- *192.0.2.1/24* - IPv4 адрес интерфейса сетевого моста *br100*;
- *2001:db8::ffff/64* - IPv6 адрес интерфейса сетевого моста *br100*;

Список команд для настройки интерфейса сетевого моста:

```
set interfaces bridge br100 address 192.0.2.1/24
set interfaces bridge br100 address 2001:db8::ffff/64
set interfaces bridge br100 member interface eth1
set interfaces bridge br100 member interface eth2.10
set interfaces bridge br100 stp
```

Пример конфигурации интерфейса сетевого моста:

```
fakel@fakel:~$ show interfaces bridge br100
 address 192.0.2.1/24
 address 2001:db8::ffff/64
 member {
     interface eth1 {
     }
     interface eth2.10 {
     }
```

```
}
stp
```

Основные настройки интерфейса сетевого моста

- `set interfaces bridge <brN> member interface <member>`

Добавляет сетевой интерфейс <*member*> в состав интерфейса моста <*brN*>.

- `set interfaces bridge <brN> member interface <member> priority <priority>`

Устанавливает приоритет <*priority*> для сетевого интерфейса <*member*>, который входит в состав сетевого моста <*brN*>.

- `set interfaces bridge <brN> member interface <member> cost <cost>`

Устанавливает стоимость <*cost*> для сетевого интерфейса <*member*>, который входит в состав сетевого моста <*brN*>.

- `set interfaces bridge <brN> address <address|dhcp|dhcpv6>`

Задает адрес <*address|dhcp|dhcpv6*> для сетевого интерфейса <*brN*>.

- адрес может быть указан несколько раз как IPv4 и/или IPv6 адрес, например, *192.0.2.1/24* и/или *2001:db8::1/64*
- адрес интерфейса dhcp принимается по протоколу DHCP от DHCP-сервера данного сегмента.
- dhcpv6 адрес интерфейса получен DHCPv6 от DHCPv6-сервера на данном сегменте.

- `set interfaces bridge <brN> description <text>`

Задает псевдоним <*text*> для сетевого интерфейса <*brN*>. Например, псевдоним может использоваться командой show interfaces или средствами мониторинга на базе SNMP.

- `set interfaces bridge <brN> disable`

Отключает сетевой интерфейс <*brN*>. После отключения интерфейс будет переведен в состояние административного отключения (A/D).

- `set interfaces bridge <brN> disable-flow-control`

Отключает генерацию управления потоком Ethernet (кадров паузы).

Управление потоком Ethernet — это механизм временного прекращения передачи данных в компьютерных сетях семейства Ethernet. Цель этого механизма - обеспечить нулевую потерю пакетов в условиях перегрузки сети.

Станция отправитель (компьютер или сетевой коммутатор) может передавать данные быстрее, чем другой конец канала связи может их принять. Используя управление потоком, принимающая станция может подать сигнал отправителю с просьбой приостановить передачу до тех пор, пока получатель не поймает его.

```
set interfaces bridge <brN> disable-link-detect
```

С помощью этой команды можно настроить сетевой интерфейс таким образом, чтобы он не обнаруживал изменений физического состояния канала связи, например, при отсоединении кабеля.

По умолчанию сетевые интерфейсы настроены на обнаружение изменения физического состояния канала.

```
set interfaces bridge <brN> mac <address>
```

Задает MAC адрес <*address*> для сетевого интерфейса <*brN*>.

```
set interfaces bridge <brN> mtu <mtu>
```

Устанавливает размер MTU <*mtu*> для сетевого интерфейса <*brN*>.

```
set interfaces bridge <brN> ip arp-cache-timeout <1-86400>
```

Устанавливает тайм-аут записи в ARP-кэш в секундах <*1-86400*> для сетевого интерфейса <*brN*>. По умолчанию это значение равно 30 секундам.

```
set interfaces bridge <brN> ip disable-arp-filter
```

Отключает фильтрацию ARP на сетевом интерфейсе <*brN*>.

Если этот параметр установлен, то ядро может отвечать на запросы ARP адресами с других интерфейсов.

По умолчанию этот параметр отключен. Это позволяет иметь несколько сетевых интерфейсов в одной подсети, и ответ на ARP адреса каждого интерфейса зависит от того, будет ли ядро маршрутизировать пакет с ARP адреса через этот интерфейс. Для работы этой функции необходимо использовать маршрутизацию на основе источника.

```
set interfaces bridge <brN> ip disable-forwarding
```

Отключает переадресацию IP на сетевом интерфейсе <*brN*>.

```
set interfaces bridge <brN> ip enable-arp-accept
```

Включает прием ARP пакетов на сетевом интерфейсе <*brN*>.

Определяет поведение для ARP пакетов, IP записи которых еще не присутствует в ARP таблице. Если настройка включена, то создаются новые записи в ARP таблице.

Как ответы, так и запросы типа *gratuitous arp* будут вызывать обновление ARP таблицы, если эта настройка включена.

Если в ARP таблице уже содержится IP-адрес кадра *gratuitous arp*, то ARP таблица будет обновлена независимо от того, включена или выключена эта настройка.

```
set interfaces bridge <brN> ip enable-arp-announce
```

Включает анонсирование ARP пакетов на сетевом интерфейсе <*brN*>.

```
set interfaces bridge <brN> ip enable-arp-ignore
```

Включает игнорирование ARP пакетов на сетевом интерфейсе <*brN*>.

Этот параметр определяет режимы отправки ответов в ответ на полученные ARP запросы, разрешающие локальные целевые IP-адреса:

- Если параметр включен, ответ будет отправлен только в том случае, если целевой IP-адрес является локальным адресом, настроенным на входящем интерфейсе.
- Если параметр отключен (по умолчанию), ответ будет получен для любого локального целевого IP-адреса, настроенного на любом интерфейсе.

```
set interfaces bridge <brN> ip enable-proxy-arp
```

Включает прокси ARP на сетевом интерфейсе <*brN*>. Прокси ARP позволяет интерфейсу Ethernet отвечать своим MAC адресом на ARP запросы IP-адресов назначения в подсетях, подключенных к другим интерфейсам системы.

```
set interfaces bridge <brN> ip proxy-arp-pvlan
```

Включает прокси ARP частного VLAN на сетевом интерфейсе <*brN*>.

```
set interfaces bridge <brN> ip source-validation <strict|loose
|disable>
```

Включает политику проверки источника по обратному пути, как указано в RFC 3704.

В настоящее время в RFC 3704 рекомендуется включать строгий режим для предотвращения подмены IP-адресов в результате DDos атак. При использовании асимметричной или другой сложной маршрутизации рекомендуется использовать свободный режим.

- **strict**: Каждый входящий пакет проверяется по FIB, и если интерфейс не является наилучшим обратным путем, то проверка пакета будет неудачной. По умолчанию неудачные пакеты отбрасываются.
- **loose**: Адрес источника каждого входящего пакета также проверяется по FIB, и если адрес источника не достижим ни через один интерфейс, проверка пакета будет неудачной.
- **disable**: Нет проверки источника.

```
set interfaces bridge <brN> ipv6 address autoconf
```

Включает механизм получения IPv6-адреса с помощью *SLAAC* (*Stateless Address Auto-configuration*).

> **Примечание**
>
> *Этот метод автоматически отключает пересылку трафика IPv6 на сетевом интерфейсе.*

```
set interfaces bridge <brN> ipv6 address eui64 <prefix>
```

Устанавливает префикс EUI-64 на основе MAC адреса <*prefix*> для IPv6 адреса сетевого интерфейса <*brN*>.

```
set interfaces bridge <brN> ipv6 address no-default-link-local
```

Удаляет локальный адрес IPv6 по умолчанию для сетевого интерфейса <*brN*>.

```
set interfaces bridge <brN> ipv6 disable-forwarding
```

Отключает IP переадресацию на сетевом интерфейсе <*brN*>.

```
set interfaces bridge <brN> vrf <vrf>
```

Размещает сетевой интерфейс <*brN*> в указанном экземпляре VRF <*vrf*>.

Настройка получения адреса по DHCPv6 для интерфейса сетевого моста

```
set interfaces bridge <brN> dhcp-options client-id
<description>
```

Устанавливает идентификатор <*text*>, используемый клиентом для идентификации себя DHCP сервером.

```
set interfaces bridge <brN> dhcp-options host-name <hostname>
```

Изменяет системное имя устройства <*hostname*>, передаваемое DHCP серверу.

- `set interfaces bridge <brN> dhcp-options vendor-class-id <vendor-id>`

Определите тип производителя клиента <*vendor-id*> для DHCP сервера.

Опция *vendor-class-id* может быть использована для запроса у сервера определенного класса опций производителя.

- `set interfaces bridge <brN> dhcp-options no-default-route`

При активации параметра *no-default-route* у DHCP сервера запрашивается только адрес, но не запрашивается шлюз по умолчанию.

- `set interfaces bridge <brN> dhcpv6-options duid <duid>`

Задает уникальный идентификатор DHCP (DUID) для сетевого интерфейса <*brN*>.

Уникальный идентификатор DHCP (DUID) используется клиентом для получения IP-адреса от сервера DHCPv6. Он имеет 2-байтовое поле типа DUID и поле идентификатора переменной длины до 128 байт. Его фактическая длина зависит от типа. Сервер сравнивает DUID со своей базой данных и выдает клиенту данные (адрес, время аренды, DNS-серверы и т. д.).

- `set interfaces bridge <brN> dhcpv6-options parameters-only`

Опция *parameters-only* определяет, что DHCPv6 должен обмениваться с серверами только информационными настройками. Примером таких параметров является список адресов DNS серверов. Этот параметр полезен, когда клиенту не нужны параметры с сохранением состояния, такие как IPv6 адреса или префиксы.

- `set interfaces bridge <brN> dhcpv6-options rapid-commit`

Включает опцию *rapid-commit*, которая отвечает за получение настроек для VLAN интерфейса от DHCPv6 сервера.

- `set interfaces bridge <brN> dhcpv6-options temporary`

При активации параметра *temporary* устройство запрашивает только временный адрес и не создавать партнерство *IANA* (*Identity Association for Non-temporary Addresses*).

Настройка делегирования DHCPv6 префикса для интерфейса сетевого моста

- `set interfaces bridge <brN> dhcpv6-options pd <id> length <length>`

Задает размер делегируемого префикса <*length*> для экземпляра делегирования префикса <*id*> на сетевом интерфейсе <*brN*>.

Некоторые провайдеры по умолчанию делегируют только префикс /64. Чтобы запросить определенный размер префикса, используйте эту опцию для запроса большего делегирования для данного экземпляра делегирования префикса <*id*>. Это значение находится в диапазоне от 32 до 64, поэтому вы можете запросить делегирование от /32 префикса (если ваш провайдер позволяет это сделать) до /64. Значение по умолчанию соответствует 64.

- `set interfaces bridge <brN> dhcpv6-options pd <id> interface <delegatee> address <address>`

Определяет адрес интерфейса <*address*>, используемый локально на интерфейсе <*delegatee*>, на который был делегирован префикс.

Он будет объединен с делегированным префиксом и *sla-id* для формирования полного адреса интерфейса. По умолчанию используется EUI-64 адрес интерфейса.

- `set interfaces bridge <brN> dhcpv6-options pd <id> interface <delegatee> sla-id <id>`

Задает значение идентификатора агрегатора уровня сайта SLA <*sla-id*> на интерфейсе <*delegatee*>. ID должен быть десятичным числом, большим 0, которое укладывается в длину идентификаторов SLA.

Настройка параметров интерфейса сетевого моста

- `set interfaces bridge <brN> aging <time>`

Устанавливает временной интервал <*time*>, по истечению которого MAC адрес интерфейса моста <*brN*> будет считаться устаревшим. Значение <*time*> задается в секундах. По умолчанию установлено 300 секунд.

- `set interfaces bridge <brN> max-age <time>`

Устанавливает временной интервал <*time*>, по истечению которого интерфейс моста <*brN*> перестает обмениваться информацией с соседними сетевыми мостами. Значение <*time*> задается в секундах. По умолчанию установлено 20 секунд.

- `set interfaces bridge <brN> igmp querier`

Включает механизм выбора *IGMP Querier* на интерфейсе моста <*brN*>.

- `set interfaces bridge <brN> stp`

Включает протокол *STP* (*Spanning Tree Protocol*) на интерфейсе моста. По умолчанию протокол STP отключен.

- `set interfaces bridge <brN> forwarding-delay <delay>`

Устанавливает время задержки пересылки <*delay*> для протокола STP на интерфейсе моста <*brN*>. Значение <*delay*> задается в секундах. По умолчанию установлено 15 секунд.

- `set interfaces bridge <brN> hello-time <interval>`

Устанавливает интервал отправки hello-пакетов <*interval*> для протокола STP на интерфейсе моста <*brN*>. Значение задается в секундах. По умолчанию установлено 2 секунды.

Мониторинг и эксплуатация интерфейса сетевого моста

- `show bridge`

Выводит на экран информацию обо всех настроенных интерфейсах моста

- `show bridge <name> spanning-tree`

Выводит на экран конфигурацию STP для интерфейса моста.

VLAN для интерфейса сетевого моста

IEEE 802.1q, часто называемый Dot1q, — это сетевой стандарт, поддерживающий виртуальные локальные сети VLAN в сетях Ethernet стандарта IEEE 802.3. Стандарт определяет систему для маркировки кадров Ethernet и сопутствующие процедуры, которые должны использоваться сетевыми устройствами при работе с такими кадрами. Стандарт также содержит положения о схеме приоритизации качества обслуживания, известной как IEEE 802.1p, и определяет протокол регистрации общих атрибутов.

В ПО **Факел** интерфейсы 802.1q представлены как виртуальные интерфейсы, подчиненные физическому интерфейсу. Для обозначения такого интерфейса используется термин *vif*.

> **Примечание**
>
> *Поскольку интерфейсы сетевого моста с поддержкой VLAN предполагают, что все немаркированные пакеты по умолчанию принадлежат VLAN 1 и что VLAN ID родительского интерфейса сетевого моста всегда равен 1, использование опции vif 1 для интерфейса сетевого моста недопустимо.*

Пример настройки интерфейса сетевого моста с поддержкой VLAN

Пример создания интерфейса сетевого моста с поддержкой VLAN выглядит следующим образом:

- *br100* - имя интерфейса сетевого моста
- Интерфейс *eth1* работает в режиме транк и пропускает через себя пакеты, помеченные тегом *VLAN 10*.

- Интерфейс *eth2* принадлежит *VLAN 10* и передает нетегированный трафик
- На интерфейсе сетевого моста *br100* настроен протокол STP
- *192.0.2.1/24* - IPv4 адрес интерфейса сетевого моста *br100*
- *2001:db8::ffff/64* - IPv6 адрес интерфейса сетевого моста *br100*

Список команд для настройки интерфейса сетевого моста с поддержкой VLAN:

```
set interfaces bridge br100 enable-vlan
set interfaces bridge br100 member interface eth1 allowed-vlan 10
set interfaces bridge br100 member interface eth2 native-vlan 10
set interfaces bridge br100 vif 10 address 192.0.2.1/24
set interfaces bridge br100 vif 10 address 2001:db8::ffff/64
set interfaces bridge br100 stp
```

Пример конфигурации интерфейса сетевого моста с поддержкой VLAN:

```
fakel@fakel:~$ show interfaces bridge br100
 enable-vlan
 member {
     interface eth1 {
         allowed-vlan 10
     }
     interface eth2 {
         native-vlan 10
     }
 }
 stp
 vif 10 {
     address 192.0.2.1/24
     address 2001:db8::ffff/64
 }
```

Основные настройки интерфейса сетевого моста с поддержкой VLAN

```
set interfaces bridge <brN> vif <vlan-id>
```

Создает новый интерфейс VLAN на интерфейсе <*brN*>, используя номер VLAN, указанный в поле <*vlan-id*>.

На одном физическом интерфейсе можно создать несколько интерфейсов VLAN. Диапазон идентификаторов VLAN составляет от 0 до 4094.

> **Примечание**
>
> *На Ethernet интерфейсах vif принимаются только пакеты с метками 802.1Q.*

- `set interfaces bridge <brN> vif <vlan-id> address <address|dhcp| dhcpv6>`

Задает адрес <*address* | *dhcp* | *dhcpv6*> для VLAN <*vlan-id*> интерфейса.

- address может быть указан несколько раз как IPv4 и/или IPv6-адрес, например, *192.0.2.1/24* и/или *2001:db8::1/64*
- dhcp адрес интерфейса получен по DHCP от DHCP-сервера в данном сегменте.
- dhcpv6 адрес интерфейса получен по протоколу DHCPv6 от сервера DHCPv6 в данном сегменте.

- `set interfaces bridge <brN> vif <vlan-id> description <text>`

Задает описательный псевдоним <*text*> для VLAN <*vlan-id*> интерфейса. Например, псевдоним используется командой show interfaces или средствами мониторинга на базе SNMP.

- `set interfaces bridge <brN> vif <vlan-id> disable`

Отключает VLAN <*vlan-id*> интерфейс. После отключения интерфейс будет переведен в состояние административного отключения (A/D).

- `set interfaces bridge <brN> vif <vlan-id> disable-link-detect`

С помощью этой команды можно настроить VLAN <*vlan-id*> интерфейс таким образом, чтобы он не обнаруживал изменений физического состояния канала связи, например, при отсоединении кабеля.

По умолчанию VLAN интерфейсы настроены на обнаружение изменения физического состояния канала.

- `set interfaces bridge <brN> vif <vlan-id> mac <address>`

Задает MAC адрес для VLAN <*vlan-id*> интерфейса.

- `set interfaces bridge <brN> vif <vlan-id> mtu <mtu>`

Устанавливает размер MTU <*mtu*> для VLAN <*vlan-id*> интерфейса.

- `set interfaces bridge <brN> vif <vlan-id> ip arp-cache-timeout <1-86400>`

Устанавливает тайм-аут записи в ARP-кэш в секундах <*1-86400*> для VLAN <vlan-id> интерфейса. По умолчанию это значение равно 30 секундам.

- `set interfaces bridge <brN> vif <vlan-id> ip disable-arp-filter`

Отключает фильтрацию ARP на VLAN <*vlan-id*> интерфейсе.

Если этот параметр установлен, то ядро может отвечать на запросы ARP адресами с других интерфейсов.

По умолчанию этот параметр отключен. Это позволяет иметь несколько сетевых интерфейсов в одной подсети, и ответ на ARP адреса каждого интерфейса зависит от того, будет ли ядро маршрутизировать пакет с ARP адреса через этот интерфейс. Для работы этой функции необходимо использовать маршрутизацию на основе источника.

- `set interfaces bridge <brN> vif <vlan-id> ip disable-forwarding`

Отключает переадресацию IP на VLAN <*vlan-id*> интерфейсе.

- `set interfaces bridge <brN> vif <vlan-id> ip enable-arp-accept`

Включает прием ARP пакетов на VLAN <*vlan-id*> интерфейсе.

Определяет поведение для ARP пакетов, IP записи которых еще не присутствует в ARP таблице. Если настройка включена, то создаются новые записи в ARP таблице.

Как ответы, так и запросы типа *gratuitous arp* будут вызывать обновление ARP таблицы, если эта настройка включена.

Если в ARP таблице уже содержится IP-адрес кадра *gratuitous arp*, то ARP таблица будет обновлена независимо от того, включена или выключена эта настройка.

- `set interfaces bridge <brN> vif <vlan-id> ip enable-arp-announce`

Включает анонсирование ARP пакетов на VLAN <*vlan-id*> интерфейсе.

- `set interfaces bridge <brN> vif <vlan-id> ip enable-arp-ignore`

Включает игнорирование ARP пакетов на VLAN <*vlan-id*> интерфейсе.

Этот параметр определяет режимы отправки ответов в ответ на полученные ARP запросы, разрешающие локальные целевые IP-адреса:

- Если параметр включен, ответ будет отправлен только в том случае, если целевой IP-адрес является локальным адресом, настроенным на входящем интерфейсе.
- Если параметр отключен (по умолчанию), ответ будет получен для любого локального целевого IP-адреса, настроенного на любом интерфейсе.

```
set interfaces bridge <brN> vif <vlan-id> ip enable-proxy-arp
```

Включает прокси ARP на VLAN <*vlan-id*> интерфейсе. Прокси ARP позволяет интерфейсу Ethernet отвечать своим MAC адресом на ARP запросы IP-адресов назначения в подсетях, подключенных к другим интерфейсам системы.

```
set interfaces bridge <brN> vif <vlan-id> ip proxy-arp-pvlan
```

Включает прокси ARP частного VLAN на VLAN <*vlan-id*> интерфейсе.

```
set interfaces bridge <brN> vif <vlan-id> ip source-validation
<strict|loose|disable>
```

Включает политику проверки источника по обратному пути, как указано в RFC 3704.

В настоящее время в RFC 3704 рекомендуется включать строгий режим для предотвращения подмены IP-адресов в результате DDos атак. При использовании асимметричной или другой сложной маршрутизации рекомендуется использовать свободный режим.

- **strict**: Каждый входящий пакет проверяется по FIB, и если интерфейс не является наилучшим обратным путем, то проверка пакета будет неудачной. По умолчанию неудачные пакеты отбрасываются.
- **loose**: Адрес источника каждого входящего пакета также проверяется по FIB, и если адрес источника не достижим ни через один интерфейс, проверка пакета будет неудачной.
- **disable**: Нет проверки источника.

```
set interfaces bridge <brN> vif <vlan-id> ipv6 address autoconf
```

Включает механизм получения IPv6-адреса с помощью *SLAAC* (*Stateless Address Auto-configuration*).

> **Примечание**
>
> *Этот метод автоматически отключает пересылку трафика IPv6 на сетевом интерфейсе.*

- `set interfaces bridge <brN> vif <vlan-id> ipv6 address eui64 <prefix>`

Устанавливает префикс EUI-64 на основе MAC адреса <*prefix*> для IPv6 адреса VLAN <*vlan-id*> интерфейса.

- `set interfaces bridge <brN> vif <vlan-id> ipv6 address no-default-link-local`

Удаляет локальный адрес IPv6 по умолчанию для VLAN <*vlan-id*> интерфейса.

- `set interfaces bridge <brN> vif <vlan-id> ipv6 disable-forwarding`

Отключает IP переадресацию на VLAN <*vlan-id*> интерфейсе.

- `set interfaces bridge <brN> vif <vlan-id> vrf <vrf>`

Размещает VLAN <*vlan-id*> интерфейс в указанном экземпляре VRF <*vrf*>.

Настройка получения адреса по DHCP для интерфейса сетевого моста с поддержкой VLAN

- `set interfaces bridge <brN> vif <vlan-id> dhcp-options client-id <text>`

Устанавливает идентификатор <*text*>, используемый клиентом для идентификации себя DHCP сервером

- `set interfaces bridge <brN> vif <vlan-id> dhcp-options host-name <hostname>`

Изменяет системное имя устройства <*hostname*>, передаваемое DHCP серверу.

- `set interfaces bridge <brN> vif <vlan-id> dhcp-options vendor-class-id <vendor-id>`

Определите тип производителя клиента <*vendor-id*> для DHCP сервера.

Опция *vendor-class-id* может быть использована для запроса у сервера определенного класса опций поставщика.

- `set interfaces bridge <brN> vif <vlan-id> dhcp-options no-default-route`

При активации параметра *no-default-route* у DHCP сервера запрашивается только адрес, но не запрашивается шлюз по умолчанию.

- `set interfaces bridge <brN> vif <vlan-id> dhcpv6-options duid <duid>`

Задает уникальный идентификатор DHCP (DUID) для VLAN <*vlan-id*> интерфейса.

Уникальный идентификатор DHCP (DUID) используется клиентом для получения IP-адреса от сервера DHCPv6. Он имеет 2-байтовое поле типа DUID и поле идентификатора переменной длины до 128 байт. Его фактическая длина зависит от типа. Сервер сравнивает DUID со своей базой данных и выдает клиенту данные (адрес, время аренды, DNS-серверы и т. д.).

- `set interfaces bridge <brN> vif <vlan-id> dhcpv6-options parameters-only`

Опция *parameters-only* определяет, что DHCPv6 должен обмениваться с серверами только информационными настройками. Примером таких параметров является список адресов DNS серверов. Этот параметр полезен, когда клиенту не нужны параметры с сохранением состояния, такие как IPv6 адреса или префиксы.

- `set interfaces bridge <brN> vif <vlan-id> dhcpv6-options rapid-commit`

Включает опцию *rapid-commit*, которая отвечает за получение настроек для VLAN интерфейса от DHCPv6 сервера.

- set `interfaces bridge <brN> vif <vlan-id> dhcpv6-options temporary`

При активации параметра *temporary* устройство запрашивает только временный адрес и не создавать партнерство *IANA* (*Identity Association for Non-temporary Addresses*).

Настройка делегирования DHCPv6 префикса для интерфейса сетевого моста с поддержкой VLAN

- `set` interfaces `bridge <brN> vif <vlan-id> dhcpv6-options pd <id> length <length>`

Задает размер делегируемого префикса <*length*> для экземпляра делегирования префикса <*id*> на VLAN <*vlan-id*> интерфейсе.

Некоторые провайдеры по умолчанию делегируют только префикс /64. Чтобы запросить определенный размер префикса, используйте эту опцию для запроса большего делегирования для данного экземпляра делегирования префикса <*id*>. Это значение находится в диапазоне от 32 до 64, поэтому вы можете запросить делегирование от /32 префикса (если ваш провайдер позволяет это сделать) до /64. Значение по умолчанию соответствует 64.

- `set interfaces bridge <brN> vif <vlan-id> dhcpv6-options pd <id> interface <delegatee> address <address>`

Определяет адрес интерфейса <*address*>, используемый локально на интерфейсе <*delegatee*>, на который был делегирован префикс.

Он будет объединен с делегированным префиксом и *sla-id* для формирования полного адреса интерфейса. По умолчанию используется EUI-64 адрес интерфейса.

- `set interfaces bridge <brN> vif <vlan-id> dhcpv6-options pd <id> interface <delegatee> sla-id <id>`

Задает значение идентификатора агрегатора уровня сайта SLA <*sla-id*> на интерфейсе <*delegatee*>. ID должен быть десятичным числом, большим 0, которое укладывается в длину идентификаторов SLA.

Зеркалирование интерфейса сетевого моста

Зеркалирование интерфейсов позволяет копировать входящий и исходящий трафик на указанный интерфейс, обычно подключенный к специальному оборудованию (система контроля поведения, система обнаружения вторжений или коллектор трафика) и копировать весь соответствующий трафик, полученный на этом интерфейсе. Преимущество зеркалирования трафика заключается в том, что устройства обрабатывают копию сетевого трафика, поэтому это не влияет на работу корпоративных сервисов и локальной сети организации.

Для настройки зеркалирования портов в ПО **Факел** используется опция *mirror*. Зеркалированный порт должен быть настроены для входящего и исходящего трафика.

Основные настройки зеркалирования интерфейса сетевого моста

- `set interfaces bridge <brN> mirror ingress <monitor-interface>`

Настраивает передачу копии входящего трафика от интерфейса <*brN*> на интерфейс <*monitor-interface*>.

- `set interfaces bridge <brN> mirror egress <monitor-interface>`

Настраивает передачу копии исходящего трафика от интерфейса <*brN*> на интерфейс <*monitor-interface*>.

Loopback интерфейс

Общая информация о Loopback интерфейсе

Интерфейс loopback — это логический интерфейс внутри маршрутизатора. Он не назначается физическому порту, поэтому его нельзя подключить к другому устройству.

Он считается программным интерфейсом, который автоматически переводится в состояние *up* (активен) во время работы маршрутизатора.

> **Примечание**
>
> *В системе может быть только один интерфейс loopback lo. Если требуется несколько интерфейсов, используйте тип интерфейса Dummy.*

Основные настройки Loopback интерфейса

- `set interfaces loopback lo address <address>`

Задает адрес <*address*> для Loopback интерфейса *lo*. Адрес может быть указан несколько раз как IPv4 и/или IPv6 адрес, например, *192.0.2.1/24* и/или *2001:db8::1/64*

- `set interfaces loopback lo description <description>`

Задает описательный псевдоним <*description*> для Loopback интерфейса *lo*. Например, псевдоним используется командой show interfaces или средствами мониторинга на базе SNMP.

Мониторинг и эксплуатация Loopback интерфейса

- `show interfaces loopback`

Выводит на экран краткую информацию о Loopback интерфейсе.

- `show interfaces loopback lo`

Выводит на экран подробную информацию о Loopback интерфейсе.

Dummy интерфейс

Общая информация о Dummy интерфейсе

В ПО **Факел** Dummy интерфейс работает аналогично интерфейсу Loopback. В отличие от Loopback интерфейса, который может быть только один, Dummy интерфейсов в операционной системе может быть несколько.

> **Примечание**
>
> *Dummy интерфейсы всегда остается в состоянии активен UP аналогично Loopback интерфейсу.*

Основные настройки Dummy интерфейса

- `set interfaces dummy <dumN> address <address>`

Задает адрес <*address*> для Dummy интерфейса <*dumN*>. Адрес может быть указан несколько раз как IPv4 и/или IPv6 адрес, например, *192.0.2.1/24* и/или *2001:db8::1/64*.

- `set interfaces dummy <dumN> description <description>`

Задает псевдоним <*description*> для Dummy интерфейса <*dumN*>. Например, псевдоним может использоваться командой *show interfaces* или средствами мониторинга на базе SNMP.

- `set interfaces dummy <dumN> disable`

Отключает Dummy интерфейс <*dumN*>. После отключения интерфейс будет переведен в состояние административного отключения (A/D).

- `set interfaces dummy <dumN> vrf <vrf>`

Размещает Dummy интерфейс <*dumN*> в указанном экземпляре VRF <*vrf*>.

Мониторинг и эксплуатация Dummy интерфейса

- `show interfaces dummy`

Выводит на экран краткую информацию обо всех Dummy интерфейсах.

- `show interfaces dummy <dumN>`

Выводит на экран подробную информацию об указанном Dummy интерфейсе <*dumN*>.

L2TPv3

Общая информация о L2TPv3

L2TPv3 (Layer Two Tunneling Protocol - Version 3) — это протокол мультиплексирования и установки туннелей на основе IP-пакетов. Он позволяет передавать различные протоколы и службы через IP-сети, создавая виртуальное соединение между двумя точками.

Основным преимуществом L2TPv3 является возможность передачи не только L2 кадров, но и любых других протоколов, таких, как IPv4, IPv6, IPX и т. д. Благодаря этому, данный протокол широко используется в сетях для объединения отдаленных локальных сетей, создания VPN-туннелей и обеспечения безопасности передачи данных.

Для использования L2TPv3 необходимо настроить две крайние точки туннеля — клиент и сервер. Клиентское устройство должно поддерживать протокол L2TPv3 и быть настроено для подключения к серверу. Сервер, в свою очередь, должен быть настроен для принятия соединений от клиентов и обеспечения генерации и обработки L2TPv3-пакетов.

Протокол L2TPv3 подробно описан в *RFC 3931*.

Основные настройки L2TPv3

- `set interfaces l2tpv3 <l2tpethN> address <address>`

Задает адрес <*address*> для сетевого интерфейса <*l2tpethN*>. Адрес может быть указан несколько раз как IPv4 и/или IPv6 адрес, например, *192.0.2.1/24* и/или *2001:db8::1/64*.

- `set interfaces l2tpv3 <l2tpethN> description <description>`

Задает псевдоним <*description*> для сетевого интерфейса <*l2tpethN*>. Например, псевдоним может использоваться командой *show interfaces* или средствами мониторинга на базе SNMP.

- `set interfaces l2tpv3 <l2tpethN> disable`

Отключает сетевой интерфейс <*l2tpethN*>. После отключения интерфейс будет переведен в состояние административного отключения (A/D).

- `set interfaces l2tpv3 <l2tpethN> disable-flow-control`

Отключает генерацию управления потоком Ethernet (кадров паузы).

Управление потоком Ethernet — это механизм временного прекращения передачи данных в компьютерных сетях семейства Ethernet. Цель этого механизма - обеспечить нулевую потерю пакетов в условиях перегрузки сети.

Станция отправитель (компьютер или сетевой коммутатор) может передавать данные быстрее, чем другой конец канала связи может их принять. Используя управление потоком, принимающая станция может подать сигнал отправителю с просьбой приостановить передачу до тех пор, пока получатель не поймает его.

- `set interfaces l2tpv3 <l2tpethN> disable-link-detect`

С помощью этой команды можно настроить сетевой интерфейс таким образом, чтобы он не обнаруживал изменений физического состояния канала связи, например, при отсоединении кабеля.

По умолчанию сетевые интерфейсы настроены на обнаружение изменения физического состояния канала.

- `set interfaces l2tpv3 <l2tpethN> mac <address>`

Задает MAC адрес *<address>* для сетевого интерфейса *<l2tpethN>*.

- `set interfaces l2tpv3 <l2tpethN> mac <address>`

Устанавливает размер MTU *<mtu>* для сетевого интерфейса *<l2tpethN>*.

- `set interfaces l2tpv3 <l2tpethN> ip arp-cache-timeout <1-86400>`

Устанавливает тайм-аут записи в ARP-кэш в секундах *<1-86400>* для сетевого интерфейса *<l2tpethN>*. По умолчанию это значение равно 30 секундам.

- `set interfaces l2tpv3 <l2tpethN> ip disable-arp-filter`

Отключает фильтрацию ARP на сетевом интерфейсе *<l2tpethN>*.

Если этот параметр установлен, то ядро может отвечать на запросы ARP адресами с других интерфейсов.

По умолчанию этот параметр отключен. Это позволяет иметь несколько сетевых интерфейсов в одной подсети, и ответ на ARP адреса каждого интерфейса зависит от того, будет ли ядро маршрутизировать пакет с ARP адреса через этот интерфейс. Для работы этой функции необходимо использовать маршрутизацию на основе источника.

- `set interfaces l2tpv3 <l2tpethN> ip disable-forwarding`

Отключает переадресацию IP на сетевом интерфейсе *<l2tpethN>*.

```
set interfaces l2tpv3 <l2tpethN> ip enable-arp-accept
```

Включает прием ARP пакетов на сетевом интерфейсе *<l2tpethN>*.

Определяет поведение для ARP пакетов, IP записи которых еще не присутствует в ARP таблице. Если настройка включена, то создаются новые записи в ARP таблице.

Как ответы, так и запросы типа *gratuitous arp* будут вызывать обновление ARP таблицы, если эта настройка включена.

Если в ARP таблице уже содержится IP-адрес кадра *gratuitous arp*, то ARP таблица будет обновлена независимо от того, включена или выключена эта настройка.

```
set interfaces l2tpv3 <l2tpethN> ip enable-arp-announce
```

Включает анонсирование ARP пакетов на сетевом интерфейсе *<l2tpethN>*.

```
set interfaces l2tpv3 <l2tpethN> ip enable-arp-ignore
```

Включает игнорирование ARP пакетов на сетевом интерфейсе *<l2tpethN>*.

Этот параметр определяет режимы отправки ответов в ответ на полученные ARP запросы, разрешающие локальные целевые IP-адреса:

- Если параметр включен, ответ будет отправлен только в том случае, если целевой IP-адрес является локальным адресом, настроенным на входящем интерфейсе.
- Если параметр отключен (по умолчанию), ответ будет получен для любого локального целевого IP-адреса, настроенного на любом интерфейсе.

```
set interfaces l2tpv3 <l2tpethN> ip enable-proxy-arp
```

Включает прокси ARP на сетевом интерфейсе *<l2tpethN>*. Прокси ARP позволяет интерфейсу Ethernet отвечать своим MAC адресом на ARP запросы IP-адресов назначения в подсетях, подключенных к другим интерфейсам системы.

```
set interfaces l2tpv3 <l2tpethN> ip proxy-arp-pvlan
```

Включает прокси ARP частного VLAN на сетевом интерфейсе *<l2tpethN>*.

```
set interfaces l2tpv3 <l2tpethN> ip source-validation <strict|
   loose|disable>
```

Включает политику проверки источника по обратному пути, как указано в RFC 3704.

В настоящее время в RFC 3704 рекомендуется включать строгий режим для предотвращения подмены IP-адресов в результате DDos атак. При использовании асимметричной или другой сложной маршрутизации рекомендуется использовать свободный режим.

- **strict**: Каждый входящий пакет проверяется по FIB, и если интерфейс не является наилучшим обратным путем, то проверка пакета будет неудачной. По умолчанию неудачные пакеты отбрасываются.
- **loose**: Адрес источника каждого входящего пакета также проверяется по FIB, и если адрес источника не достижим ни через один интерфейс, проверка пакета будет неудачной.
- **disable**: Нет проверки источника

```
set interfaces l2tpv3 <l2tpethN> ipv6 address autoconf
```

Включает механизм получения IPv6-адреса с помощью *SLAAC* (*Stateless Address Auto-configuration*).

> **Примечание**
>
> *Этот метод автоматически отключает пересылку трафика IPv6 на сетевом интерфейсе.*

```
set interfaces l2tpv3 <l2tpethN> ipv6 address eui64 <prefix>
```

Устанавливает префикс EUI-64 на основе MAC адреса *<prefix>* для IPv6 адреса сетевого интерфейса *<l2tpethN>*.

```
set interfaces l2tpv3 <l2tpethN> ipv6 address no-default-link-
   local
```

Удаляет локальный адрес IPv6 по умолчанию для сетевого интерфейса *<l2tpethN>*.

```
set interfaces l2tpv3 <l2tpethN> ipv6 disable-forwarding
```

Отключает IP переадресацию на сетевом интерфейсе *<l2tpethN>*.

```
set interfaces l2tpv3 <l2tpethN> vrf <vrf>
```

Размещает сетевой интерфейс *<l2tpethN>* в указанном экземпляре VRF *<vrf>*.

Настройка параметров L2TPv3

```
set interfaces l2tpv3 <l2tpethN> encapsulation <udp|ip>
```

Устанавливает тип инкапсуляции *<udp|ip>* для туннеля. По умолчанию используется значение udp.

```
set interfaces l2tpv3 <l2tpethN> source-address <address>
```

Задает IP-адрес локального устройства *<address>*, который будет использоваться для построения туннеля.

Этот адрес должен принадлежать одному из интерфейсов маршрутизатора. Он может быть указан как IPv4 адрес или IPv6 адрес.

```
set interfaces l2tpv3 <l2tpethN> remote <address>
```

Задает IP-адрес удаленного устройства *<address>*, который будет использоваться для построения туннеля. Он может быть указан как IPv4 адрес или IPv6 адрес.

```
set interfaces l2tpv3 <l2tpethN> session-id <id>
```

Устанавливает локальный идентификатор сеанса *<id>*, который представляет собой 32-разрядное целочисленное значение. Используемое значение должно совпадать со значением *peer_session_id*.

```
set interfaces l2tpv3 <l2tpethN> peer-session-id <id>
```

Устанавливает удаленный идентификатор сеанса <*id*>, который представляет собой 32-битное целочисленное значение, присваиваемое сеансу со стороны устройства. Используемое значение должно совпадать со значением *session_id*.

```
set interfaces l2tpv3 <l2tpethN> tunnel-id <id>
```

Задает локальный идентификатор туннеля <*id*>, который представляет собой 32-разрядное целочисленное значение. Идентификатор определяет туннель, в котором будет создана сессия.

```
set interfaces l2tpv3 <l2tpethN> peer-tunnel-id <id>
```

Задает удаленный идентификатор туннеля, который представляет собой 32-разрядное целочисленное значение. Идентификатор определяет туннель, в котором будет создана сессия.

Туннельный интерфейс

Общая информация о туннельном интерфейсе

Подобно VPN, IP-туннель напрямую соединяет две сети через третью сеть, например Интернет. Однако не все туннельные протоколы поддерживают шифрование.

Маршрутизаторы в обеих сетях, создающие туннель, должны иметь как минимум два интерфейса:

- один интерфейс, подключенный к локальной сети;
- один интерфейс, подключенный к сети, через которую создается туннель.

Для создания туннеля на обоих маршрутизаторах создается виртуальный интерфейс с IP-адресом из удаленной подсети.

ПО **Факел** поддерживает следующие типы IP-туннелей:

- IPv4 поверх IPv4 (IPIP);
- IPv4 поверх IPv6 (IPIP6);
- IPv6 поверх IPv6 (IP6IP6);
- Simple Internet Transition (SIT);
- Generic Routing Encapsulation (GRE);
- Generic Routing Encapsulation over IPv6 (IP6GRE);
- Generic Routing Encapsulation Terminal Access Point (GRETAP);
- Generic Routing Encapsulation Terminal Access Point over IPv6 (IP6GRETAP);
- Virtual Tunnel Interfaces (VTI).

В зависимости от типа эти туннели действуют либо на втором, либо на третьем уровне модели OSI.

Основные настройки туннельного интерфейса

- `set interfaces tunnel <tunN> address <address|dhcp|dhcpv6>`

Задает адрес <*address|dhcp|dhcpv6*> для сетевого интерфейса <*tunN*>. Адрес может быть указан несколько раз как IPv4 и/или IPv6 адрес, например, *192.0.2.1/24* и/или *2001:db8::1/64*.

- `set interfaces tunnel <tunN> description <description>`

Задает псевдоним <*description*> для сетевого интерфейса <*tunN*>. Например, псевдоним может использоваться командой ***show interfaces*** или средствами мониторинга на базе SNMP.

- `set interfaces tunnel <tunN> disable`

Отключает сетевой интерфейс <*tunN*>. После отключения интерфейс будет переведен в состояние административного отключения (A/D).

- `set interfaces tunnel <tunN> disable-flow-control`

Отключает генерацию управления потоком Ethernet (кадров паузы).

Управление потоком Ethernet — это механизм временного прекращения передачи данных в компьютерных сетях семейства Ethernet. Цель этого механизма - обеспечить нулевую потерю пакетов в условиях перегрузки сети.

Станция отправитель (компьютер или сетевой коммутатор) может передавать данные быстрее, чем другой конец канала связи может их принять. Используя управление потоком, принимающая станция может подать сигнал отправителю с просьбой приостановить передачу до тех пор, пока получатель не поймает его.

- `set interfaces tunnel <tunN> disable-link-detect`

С помощью этой команды можно настроить сетевой интерфейс таким образом, чтобы он не обнаруживал изменений физического состояния канала связи, например, при отсоединении кабеля.

По умолчанию сетевые интерфейсы настроены на обнаружение изменения физического состояния канала.

- `set interfaces tunnel <tunN> mtu <mtu>`

Устанавливает размер MTU <*mtu*> для сетевого интерфейса <*tunN*>.

- `set interfaces tunnel <tunN> ip arp-cache-timeout <1-86400>`

Устанавливает тайм-аут записи в ARP-кэш в секундах *<1-86400>* для сетевого интерфейса *<tunN>*. По умолчанию это значение равно 30 секундам.

- `set interfaces tunnel <tunN> ip disable-arp-filter`

Отключает фильтрацию ARP на сетевом интерфейсе *<tunN>*.

Если этот параметр установлен, то ядро может отвечать на запросы ARP адресами с других интерфейсов.

По умолчанию этот параметр отключен. Это позволяет иметь несколько сетевых интерфейсов в одной подсети, и ответ на ARP адреса каждого интерфейса зависит от того, будет ли ядро маршрутизировать пакет с ARP адреса через этот интерфейс. Для работы этой функции необходимо использовать маршрутизацию на основе источника.

- `set interfaces tunnel <tunN> ip disable-forwarding`

Отключает переадресацию IP на сетевом интерфейсе *<tunN>*.

- `set interfaces tunnel <tunN> ip enable-arp-accept`

Включает прием ARP пакетов на сетевом интерфейсе *<tunN>*.

Определяет поведение для ARP пакетов, IP записи которых еще не присутствует в ARP таблице. Если настройка включена, то создаются новые записи в ARP таблице.

Как ответы, так и запросы типа *gratuitous arp* будут вызывать обновление ARP таблицы, если эта настройка включена.

Если в ARP таблице уже содержится IP-адрес кадра *gratuitous arp*, то ARP таблица будет обновлена независимо от того, включена или выключена эта настройка.

- `set interfaces tunnel <tunN> ip enable-arp-announce`

Включает анонсирование ARP пакетов на сетевом интерфейсе *<tunN>*.

- `set interfaces tunnel <tunN> ip enable-arp-ignore`

Включает игнорирование ARP пакетов на сетевом интерфейсе *<tunN>*.

Этот параметр определяет режимы отправки ответов в ответ на полученные ARP запросы, разрешающие локальные целевые IP-адреса:

- Если параметр включен, ответ будет отправлен только в том случае, если целевой IP-адрес является локальным адресом, настроенным на входящем интерфейсе.

- Если параметр отключен (по умолчанию), ответ будет получен для любого локального целевого IP-адреса, настроенного на любом интерфейсе.

```
set interfaces tunnel <tunN> ip enable-proxy-arp
```

Включает прокси ARP на сетевом интерфейсе <*tunN*>. Прокси ARP позволяет интерфейсу Ethernet отвечать своим MAC адресом на ARP запросы IP-адресов назначения в подсетях, подключенных к другим интерфейсам системы.

```
set interfaces tunnel <tunN> ip proxy-arp-pvlan
```

Включает прокси ARP частного VLAN на сетевом интерфейсе <*tunN*>.

```
set interfaces tunnel <tunN> ip source-validation
<strict|loose| disable>
```

Включает политику проверки источника по обратному пути, как указано в RFC 3704.

В настоящее время в RFC 3704 рекомендуется включать строгий режим для предотвращения подмены IP-адресов в результате DDos атак. При использовании асимметричной или другой сложной маршрутизации рекомендуется использовать свободный режим.

- **strict**: Каждый входящий пакет проверяется по FIB, и если интерфейс не является наилучшим обратным путем, то проверка пакета будет неудачной. По умолчанию неудачные пакеты отбрасываются.
- **loose**: Адрес источника каждого входящего пакета также проверяется по FIB и, если адрес источника не достижим ни через один интерфейс, проверка пакета будет неудачной.
- **disable**: Нет проверки источника

```
set interfaces tunnel <tunN> ipv6 address autoconf
```

Включает механизм получения IPv6-адреса с помощью *SLAAC* (*Stateless Address Auto-configuration*).

> **Примечание**
>
> *Этот метод автоматически отключает пересылку трафика IPv6 на сетевом интерфейсе.*

```
set interfaces tunnel <tunN> ipv6 address eui64 <prefix>
```

Устанавливает префикс EUI-64 на основе MAC адреса <*prefix*> для IPv6 адреса сетевого интерфейса <*tunN*>.

```
set interfaces tunnel <tunN> ipv6 address no-default-link-local
```

Удаляет локальный адрес IPv6 по умолчанию для сетевого интерфейса <*tunN*>.

- `set interfaces tunnel <tunN> ipv6 disable-forwarding`

Отключает IP переадресацию на сетевом интерфейсе <*tunN*>.

- `set interfaces tunnel <tunN> vrf <vrf>`

Размещает сетевой интерфейс <*tunN*> в указанном экземпляре VRF <*vrf*>.

IPv4 туннель

Туннель IPIP, как можно понять из его названия — это туннель, работающий в режиме «IP over IP» RFC 2003.

Заголовок пакета туннеля IPIP выглядит следующим образом:

Такие туннели обычно используются для соединения двух внутренних IPv4-подсетей через общедоступную IPv4-сеть (интернет). Применение IPIP создаёт минимальную дополнительную нагрузку на систему, но по такому туннелю можно выполнять только однонаправленную передачу данных (unicast). То есть, построив подобный туннель, нельзя будет использовать его для групповой передачи данных (multicast).

IPIP-туннели поддерживают режимы «IP over IP» и «MPLS over IP».

> **Примечание**
>
> *Когда загружен модуль ipip, или когда впервые создано IPIP-устройство, ядро Linux создаст в каждом пространстве имён устройство по умолчанию tunl0 с атрибутами local=any и remote=any. Получая IPIP-пакеты, ядро, в определённых случаях, будет перенаправлять их на tunl0 как на устройство, используемое по умолчанию. Это происходит тогда, когда ядро не может найти другого устройства, атрибуты local/remote которого более точно соответствуют адресам источника и приёмника пакетов.*

Пример настройки туннеля IPv4

Пример настройки IPIP туннеля между двумя маршрутизаторами.

Для настройки IPIP туннеля будут использованы следующие параметры:

- *192.0.2.10* - адрес для построения туннеля на 1-м маршрутизаторе;
- *203.0.113.20* - адрес для построения туннеля на 2-м маршрутизаторе;

- *192.168.100.200/24* - локальный адрес туннельного интерфейса на 1-м маршрутизаторе;
- *10.16.200.200/24* - локальный адрес туннельного интерфейса на 2-м маршрутизаторе.

Список команд для настройки на 1-м маршрутизаторе:

```
set interfaces tunnel tun0 encapsulation ipip
set interfaces tunnel tun0 source-address 192.0.2.10
set interfaces tunnel tun0 remote 203.0.113.20
set interfaces tunnel tun0 address 192.168.100.200/24
```

Список команд для настройки на 2-м маршрутизаторе:

```
set interfaces tunnel tun0 encapsulation ipip
set interfaces tunnel tun0 source-address 203.0.113.20
set interfaces tunnel tun0 remote 192.0.2.10
set interfaces tunnel tun0 address 10.16.200.200/24
```

Основные настройки туннеля IPv4

```
set interfaces tunnel <tunN> encapsulation ipip
```

Устанавливает тип инкапсуляции IPIP для указанного туннельного интерфейса <*tunN*>.

> **Примечание**
>
> *Полный список команд для настройки туннельного интерфейса представлен в разделе* ***Основные настройки туннельного интерфейса.***

IPv6 туннель

Туннельный интерфейсы может работает в режиме «IPv4/IPv6 over IPv6». Он похож на IPv6 версию туннеля SIT.

Заголовок пакета IPv6 туннеля выглядит следующим образом:

Туннельный интерфейс IPv6 поддерживает режимы работы

- *IP6IP6* - IPv6 аналог туннеля IPIP. Режим IP6IP6 представлен схемой «IPv6 over IPv6».
- *IPIP6* - IPv4 вложенный в IPv6. Режим IPIP6 представлен схемой «IPv4 over IPv6».

Пример настройки туннеля IPv6

IP6IP6

Пример настройки IP6IP6 туннеля между двумя маршрутизаторами.

Для настройки IP6IP6 туннеля будут использованы следующие параметры:

- *2001:db8:aa::1* - адрес для построения туннеля на 1-м маршрутизаторе;
- *2001:db8:aa::2* - адрес для построения туннеля на 2-м маршрутизаторе;
- *2001:db8:bb::1/64* - локальный адрес туннельного интерфейса на 1-м маршрутизаторе;
- *2001:db8:bb::2/64* - локальный адрес туннельного интерфейса на 2-м маршрутизаторе.

Список команд для настройки на 1-м маршрутизаторе:

```
set interfaces tunnel tun0 encapsulation ip6ip6
set interfaces tunnel tun0 source-address 2001:db8:aa::1
set interfaces tunnel tun0 remote 2001:db8:aa::2
set interfaces tunnel tun0 address 2001:db8:bb::1/64
```

Список команд для настройки на 2-м маршрутизаторе:

```
set interfaces tunnel tun0 encapsulation ip6ip6
set interfaces tunnel tun0 source-address 2001:db8:aa::1
set interfaces tunnel tun0 remote 2001:db8:aa::2
set interfaces tunnel tun0 address 2001:db8:bb::1/64
```

IPIP6

Пример настройки IPIP6 туннеля между двумя маршрутизаторами.

Для настройки IPIP6 туннеля будут использованы следующие параметры:

- *2001:db8:aa::1* - адрес для построения туннеля на 1-м маршрутизаторе;
- *2001:db8:aa::2* - адрес для построения туннеля на 2-м маршрутизаторе;
- *192.168.70.80/24* - локальный адрес туннельного интерфейса на 1-м маршрутизаторе;

- *10.16.170.80/24* - локальный адрес туннельного интерфейса на 2-м маршрутизаторе.

Список команд для настройки на 1-м маршрутизаторе:

```
set interfaces tunnel tun0 encapsulation ipip6
set interfaces tunnel tun0 source-address 2001:db8:aa::1
set interfaces tunnel tun0 remote 2001:db8:aa::2
set interfaces tunnel tun0 address 192.168.70.80/24
```

Список команд для настройки на 2-м маршрутизаторе:

```
set interfaces tunnel tun0 encapsulation ipip6
set interfaces tunnel tun0 source-address 2001:db8:aa::2
set interfaces tunnel tun0 remote 2001:db8:aa::1
set interfaces tunnel tun0 address 10.16.170.80/24
```

Основные настройки туннеля IPv6

```
set interfaces tunnel <tunN> encapsulation ip6ip6
```

Устанавливает тип инкапсуляции IP6IP6 для указанного туннельного интерфейса <*tunN*>.

```
set interfaces tunnel <tunN> encapsulation ipip6
```

Устанавливает тип инкапсуляции IPIP6 для указанного туннельного интерфейса <*tunN*>.

> **Примечание**
>
> *Полный список команд для настройки туннельного интерфейса представлен в разделе **Основные настройки туннельного интерфейса.***

SIT туннель

SIT (Simple Internet Transition) — это технология создания туннелей, главной целью существования которой является соединение изолированных IPv6 сетей через интернет с использованием протокола IPv4.

Изначально технология SIT могла работать лишь в режиме туннелирования «IPv6 over IPv4». Однако за годы развития она приобрела поддержку ещё нескольких режимов. В частности, это ipip (то же самое было с IPIP туннелем), ip6ip, mplsip и any.

Режим any используется для работы с IPv4 и IPv6 трафиком, что может оказаться полезным в некоторых ситуациях. SIT туннели также поддерживают ISATAP.

Заголовок SIT пакета выглядит следующим образом:

Пример настройки SIT туннеля

Пример настройки SIT туннеля между двумя маршрутизаторами.

Для настройки SIT туннеля будут использованы следующие параметры:

- *192.0.2.10* - адрес для построения туннеля на 1-м маршрутизаторе;
- *192.0.2.20* - адрес для построения туннеля на 2-м маршрутизаторе;
- *2001:db8:bb::1/64* - локальный адрес туннельного интерфейса на 1-м маршрутизаторе;
- *2001:db8:aa::1/64* - локальный адрес туннельного интерфейса на 2-м маршрутизаторе.

Список команд для настройки на 1-м маршрутизаторе:

```
set interfaces tunnel tun0 encapsulation sit
set interfaces tunnel tun0 source-address 192.0.2.10
set interfaces tunnel tun0 remote 192.0.2.20
set interfaces tunnel tun0 address 2001:db8:bb::1/64
```

Список команд для настройки на 2-м маршрутизаторе:

```
set interfaces tunnel tun0 encapsulation sit
set interfaces tunnel tun0 source-address 192.0.2.20
set interfaces tunnel tun0 remote 192.0.2.10
set interfaces tunnel tun0 address 2001:db8:aa::1/64
```

Основные настройки SIT туннеля

```
set interfaces tunnel <tunN> encapsulation sit
```

Устанавливает тип инкапсуляции SIT для указанного туннельного интерфейса <*tunN*>.

Примечание

*Полный список команд для настройки туннельного интерфейса представлен в разделе **Основные настройки туннельного интерфейса.***

GRE туннель

Протокол GRE

Технология GRE (Generic Routing Encapsulation) описана в *RFC 2784*. При GRE туннелировании между заголовками внутреннего и внешнего IP пакета добавляется дополнительный заголовок GRE.

GRE инкапсулирует пакеты протокола 3-го уровня с допустимым Ethernet типом. Это отличает технологию GRE от технологии IPIP, которая поддерживает лишь инкапсуляцию IP пакетов. Основным преимуществом туннеля GRE является возможность передачи нескольких протоколов внутри одного туннеля. GRE поддерживает многоадресный трафик и поддерживает протоколы маршрутизации, использующие многоадресную рассылку для формирования соседних связей.

Заголовок пакета при использовании технологии GRE выглядит следующим образом:

Примечание

Туннели GRE позволяют выполнять групповую передачу данных и поддерживают IPv6.

Кроме того, GRE позволяет создавать несколько туннелей с одним и тем же источником и назначением благодаря поддержке туннельных ключей. Несмотря на свое название, эта функция не имеет никакого отношения к безопасности: это просто идентификатор, позволяющий маршрутизаторам отличать один туннель от другого.

В ПО **Факел** туннель GRE может передавать IPv4 и IPv6 трафик.

Протокол GRE6

Протокол GRE6 — это IPv6 эквивалент протокола GRE. Туннели GRE6 позволяют инкапсулировать любые протоколы 3 уровня в IPv6.

Заголовок пакета при использовании технологии GRE6 выглядит следующим образом:

Протокол GRETAP и GRE6TAP

В то время как туннели GRE и GRE6 работают на 3 уровне модели OSI, туннели GRETAP и GRE6TAP работают на 2 уровне модели OSI. Это означает, что одними из внутренних заголовков соответствующих пакетов являются Ethernet-заголовки.

Заголовок пакета при использовании технологии GRE выглядит следующим образом:

В туннелях GRE6TAP, как и в туннелях GRETAP, среди внутренних заголовков пакета есть и Ethernet-заголовки:

Пример настройки GRE туннеля

Пример настройки GRE туннеля между двумя маршрутизаторами. Для настройки GRE туннеля между двумя маршрутизаторами требуется указать адрес внешнего интерфейса локального устройства, адрес внешнего интерфейса удаленного устройства, тип инкапсуляции и адрес внутреннего интерфейса локального устройства.

Для настройки GRE туннеля будут использованы следующие параметры:

- *198.51.100.2* - адрес для построения туннеля на 1-м маршрутизаторе
- *203.0.113.10* - адрес для построения туннеля на 2-м маршрутизаторе
- *10.0.55.1/30* - локальный адрес туннельного интерфейса на 1-м маршрутизаторе
- *192.168.55.2/30* - локальный адрес туннельного интерфейса на 2-м маршрутизаторе

Список команд для настройки на 1-м маршрутизаторе:

```
set interfaces tunnel tun100 address 10.0.55.1/30
set interfaces tunnel tun100 encapsulation gre
set interfaces tunnel tun100 source-address 198.51.100.2
set interfaces tunnel tun100 remote 203.0.113.10
```

Список команд для настройки на 2-м маршрутизаторе:

```
set interfaces tunnel tun100 address 10.0.55.1/30
set interfaces tunnel tun100 encapsulation gre
```

```
set interfaces tunnel tun100 source-address 198.51.100.2
set interfaces tunnel tun100 remote 203.0.113.10
```

Основные настройки GRE туннеля

```
set interfaces tunnel <tunN> encapsulation gre
```

Устанавливает тип инкапсуляции GRE для указанного туннельного интерфейса **<tunN>**.

```
set interfaces tunnel <tunN> encapsulation ip6gre
```

Устанавливает тип инкапсуляции GRE для указанного туннельного интерфейса **<tunN>**.

```
set interfaces tunnel <tunN> encapsulation gretap
```

Устанавливает тип инкапсуляции GRETAP для указанного туннельного интерфейса **<tunN>**.

```
set interfaces tunnel <tunN> encapsulation gre6tap
```

Устанавливает тип инкапсуляции GRE6TAP для указанного туннельного интерфейса **<tunN>**.

```
set interfaces tunnel <tunN> parameters ip key <key>
```

Задет туннельный ключ **<key>** для указанного туннельного интерфейса **<tunN>**.

Примечание

Туннели GRE позволяют выполнять групповую передачу данных и поддерживают IPv6.

VTI интерфейс

VTI — это виртуальный интерфейс туннеля, который используется для установки виртуальных частных сетей VPN на основе IPsec. Он обеспечивает абстракцию уровня интерфейса для настройки и управления IPsec-туннелями между устройствами.

VTI интерфейсы позволяют передавать зашифрованный трафик между сетевыми устройствами через общедоступные или ненадежные сети, обеспечивая тем самым конфиденциальность, целостность и аутентификацию данных. Они также облегчают настройку и управление IPsec-туннелями, предоставляя единый интерфейс для управления VPN соединениями.

В целом, VTI туннели работают почти так же, как туннели IPIP или SIT. Исключением является то, что они задействуют *fwmark* и формирование/разбор данных IPsec.

> **! Предупреждение**
>
> *При настройке Site-to Site IPsec туннеля с использованием VTI интерфейсов, обязательно отключите автоматическую установку маршрута: set vpn ipsec options disable-route-autoinstal*

Основные настройки VTI интерфейса

- `set interfaces vti <vtiN> address <address>`

Задает адрес <*address*> для сетевого интерфейса <*vtiN*>. Адрес может быть указан несколько раз как IPv4 и/или IPv6 адрес, например, *192.0.2.1/24* и/или *2001:db8::1/64*.

- `set interfaces vti <vtiN> description <text>`

Задает псевдоним <*text*> для сетевого интерфейса <*vtiN*>. Например, псевдоним может использоваться командой show interfaces или средствами мониторинга на базе SNMP.

- `set interfaces vti <vtiN> disable`

Отключает сетевой интерфейс <*vtiN*>. После отключения интерфейс будет переведен в состояние административного отключения (A/D).

- `set interfaces vti <vtiN> mtu <68-9000>`

Устанавливает размер MTU <*mtu*> для сетевого интерфейса <*vtiN*>.

- `set interfaces vti <vtiN> redirect <text>`

Устанавливает интерфейс <*text*> для пересылки входящих на VTI интерфейс <*vtiN*> пакетов.

WireGuard интерфейс

Общая информация о WireGuard интерфейсе

WireGuard — коммуникационный протокол и бесплатное программное обеспечение с открытым исходным кодом, который реализует зашифрованные виртуальные частные сети (VPN). По производительности он значительно превосходит IPSec и OpenVPN. Изначально он был выпущен для ядра Linux, но в настоящее время поддерживается несколькими платформами (Windows, macOS, BSD, iOS, Android) и широко распространен. В настоящее время он находится в стадии активной разработки, но уже

сейчас его можно считать наиболее безопасным, простым и удобным в использовании VPN решением в отрасли.

Преимущества WireGuard над другими VPN решениями:

- Простой в использовании;
- Использует современную криптографию: Noise protocol framework, Curve25519, ChaCha20, Poly1305, BLAKE2, SipHash24, HKDF и т.д.;
- Компактный читаемый код, проще исследовать на уязвимости;
- Высокая производительность;
- Четкая и проработанная спецификация.

Принцип работы WireGuard:

1. Инициализация и обмен ключами:
    - Каждое устройство, желающее подключиться к VPN, создает пару ключей: закрытый и открытый ключи.
    - Устройства обмениваются своими открытыми ключами.
    - При получении открытого ключа другого устройства каждое устройство может создать общий секретный ключ, используя свой закрытый ключ и открытый ключ удаленного устройства.
2. Установление туннеля:
    - После установления общего секретного ключа между двумя устройствами начинается установление защищенного туннеля.
    - Каждое устройство создает виртуальный сетевой интерфейс (например, wg0), который является точкой входа для зашифрованных данных.
3. Шифрование и передача данных:
    - Данные, отправляемые через VPN, шифруются с использованием общего секретного ключа и упаковываются в зашифрованные пакеты.
    - Зашифрованные пакеты отправляются через защищенный туннель.
4. Расшифровка и обработка данных на удаленном конце:
    - Удаленное устройство получает зашифрованные пакеты, расшифровывает их с помощью общего секретного ключа и обрабатывает, как обычные сетевые пакеты.
5. Пересылка данных в локальную сеть:
    - Расшифрованные данные передаются в локальную сеть удаленного устройства и далее к целевым устройствам, которым они предназначены.

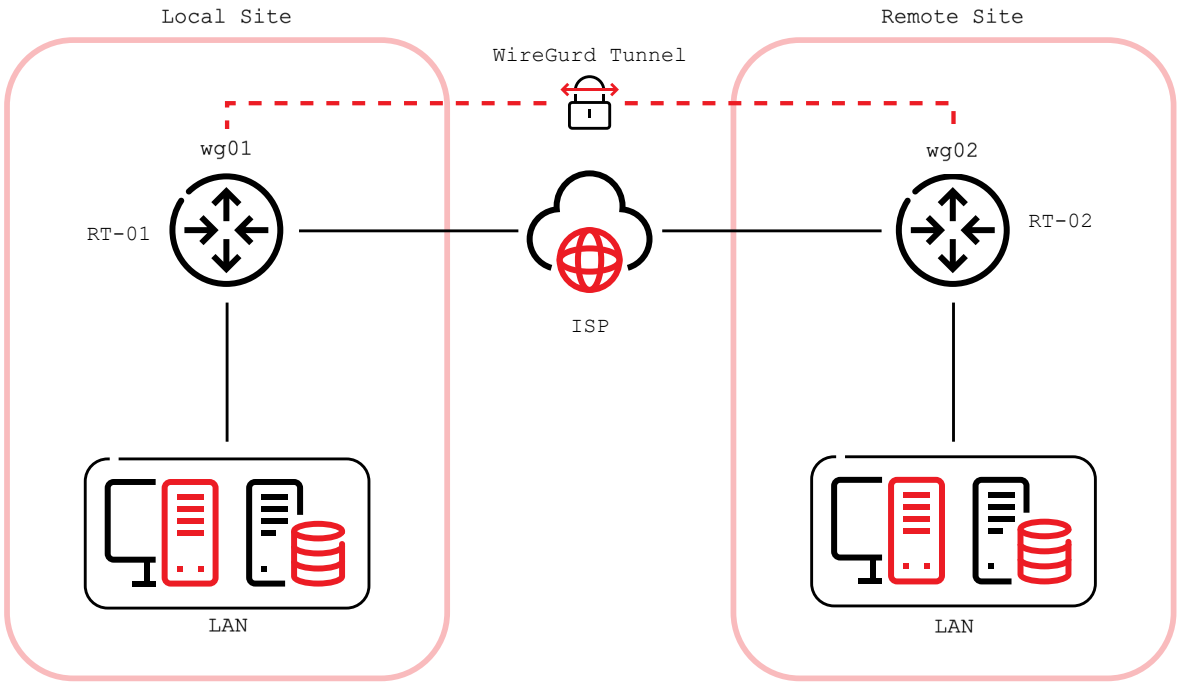
6. Обратный путь:
    - Подобные операции выполняются в обратном направлении для отправки ответов и данных обратно.

WireGuard работает на более низком уровне, чем традиционные VPN-протоколы, такие как IPsec, что делает его более эффективным и быстрым. Он также обеспечивает высокую безопасность благодаря использованию современных криптографических примитивов и простоте своей архитектуры.

> **Примечание**
>
> *Максимальное преимущество в производительности (по сравнению с OpenVPN и IPSec) будет заметно на Linux системах, так как там WireGuard реализован в виде модуля ядра. Кроме этого, поддерживаются macOS, Android, iOS, FreeBSD и OpenBSD, но в них WireGuard выполняется в userspace со всеми вытекающими последствиями для производительности.*

Пример настройки

Пример настройки WireGuard туннеля между двумя площадками.



Создание пары ключей

Для настройки WireGuard туннеля требуется создание пары ключей, которая включает в себя закрытый ключ для расшифровки входящего трафика и открытый ключ для передачи шифрованного трафика.

Список команд для создания пары ключей на локальном и удаленном устройствах:

```
fakel@RT01# generate wireguard default-keypair
fakel@RT02# generate wireguard default-keypair
```

Пара ключей будет использоваться по умолчанию на любом настроенном интерфейсе WireGuard, даже если настраивается несколько интерфейсов.

Для получения информации об открытом ключе используйте команду: ***show wireguard keypairs pubkey default***.

Команда ***show wireguard keypairs pubkey default*** выводит информацию об открытом ключе, который будет передан другому устройству для построения туннеля WireGuard. Ваше устройство будет шифровать весь трафик, передаваемый в туннеле, используя этот открытый ключ.

> **Примечание**
>
> *Если настраивается несколько интерфейсов WireGuard, то каждый из них может иметь свою собственную пару ключей. Для создания такой пары ключей используются именованные пары ключей. Именованные пары ключей могут быть созданы для каждого отдельного сетевого интерфейса.*

Настройка туннельных интерфейсов

Следующим шагом является настройка локальной стороны, а также доверенных адресов назначения на основе политики. Если ваше устройство только инициирует соединение, то настраивать порт прослушивания необязательно. Однако, если ваше устройство выступает в роли сервера и конечные точки инициируют соединения с вашей системой, то необходимо определить порт, к которому могут подключаться клиенты. Если порт будет выбран случайно это может вызвать проблемы при подключении удаленных устройств. Для настройки WireGuard туннеля требуется дополнительная настройка правил межсетевого экрана. Поскольку при каждой перезагрузке устройства порт прослушивания может быть разным, настроенные правила межсетевого экрана для WireGuard туннеля будут не актуальны.

Для настройки туннеля WireGuard потребуется открытый ключ удаленного устройства, а также сеть (сети), которую вы хотите пропускать через туннель *allowed-ips*.

Перед настройкой WireGuard интерфейсов на устройствах, между которыми будет строиться туннель, необходимо определить закрытий и открытый ключи для этих интерфейсов.

Список команд для определения закрытого и открытого ключей WireGuard интерфейса локального RT-01 и удаленного RT-02 устройств:

```
set interfaces wireguard wg01 private-key KP01
run show wireguard keypairs pubkey KP01
EKY0dxRrSD98QHjfHOK13mZ5PJ7hnddRZt5woB3szyw=
```

```
set interfaces wireguard wg01 private-key KP02
run show wireguard keypairs pubkey KP02
XMrlPykaxhdAAiSjhtPlvi30NVkvLQliQuKP7AI7CyI
```

После определения открытых ключей для всех устройств, между которыми будет строиться WireGuard туннель, можно переходить к настройкам интерфейсов.

Параметры для настройки WireGuard туннеля на локальном устройстве RT-01:

- *10.1.0.1/30* – IP-адрес WireGuard интерфейса *wg01* на локальном устройстве *RT-01*.
- *192.168.2.0/24* - адрес сети внутри туннеля WireGuard на локальном устройстве *RT-01*.
- *192.0.2.1* – IP-адрес удаленного устройства *RT-02*.
- *51820* - номер порта удаленного устройства *RT-02*.
- *XMrlPykaxhdAAiSjhtPlvi30NVkvLQliQuKP7AI7CyI=* - открытый ключ удаленного устройства *RT-02*.
- *51820* - номер порта для интерфейса *wg01*, который будет использовать удаленное устройства *RT-02* для инициации соединения с локальным устройством *RT-01*.
- Весь трафик для сети *192.168.2.0/24* будет направлен на интерфейс *wg01*.

Список команд для настойки WireGuard интерфейса на локальном устройстве RT-01:

```
set interfaces wireguard wg01 address '10.1.0.1/30'
set interfaces wireguard wg01 description 'VPN-to-wg02'
set interfaces wireguard wg01 peer to-RT-02 allowed-ips
'192.168.2.0/24'
set interfaces wireguard wg01 peer to-RT-02 address '192.0.2.1'
set interfaces wireguard wg01 peer to-RT-02 port '51820'
set interfaces wireguard wg01 peer to-RT-02 pubkey
'XMrlPykaxhdAAiSjhtPlvi30NVkvLQliQuKP7AI7CyI='
set interfaces wireguard wg01 port '51820'
set protocols static interface-route 192.168.2.0/24 next-hop-
interface wg01
```

Последним шагом является определение интерфейсного маршрута сети *192.168.2.0/24* для прохождения через интерфейс *wg01*.

Предупреждение

Нельзя назначить один и тот же адрес параметра *allowed-ips* для нескольких узлов WireGuard.

Параметры для настройки WireGuard туннеля на локальном устройстве RT-02:

- *10.1.0.2/30* – IP-адрес WireGuard интерфейса wg01 на удаленном устройстве *RT-02*;
- *192.168.1.0/24* - адрес сети внутри туннеля WireGuard на удаленном устройстве *RT-02*;
- *192.0.2.2* – IP-адрес локального устройства *RT-01*;
- *51820* - номер порта локального устройства *RT-01*;
- *EKY0dxRrSD98QHjfHOK13mZ5PJ7hnddRZt5woB3szyw=* - открытый ключ локального устройства *RT-01*;
- *51820* - номер порта для интерфейса *wg01*, который будет использовать локальное устройства *RT-01* для инициации соединения с удаленным устройством *RT-02*;
- Весь трафик для сети *192.168.1.0/24* будет направлен на интерфейс *wg01.*

Список команд для настойки WireGuard интерфейса на удаленном устройстве RT-02:

```
set interfaces wireguard wg01 address '10.1.0.2/30'
set interfaces wireguard wg01 description 'VPN-to-wg01'
set interfaces wireguard wg01 peer to-RT-01 allowed-ips
'192.168.1.0/24'
set interfaces wireguard wg01 peer to-RT-01 address '192.0.2.2'
set interfaces wireguard wg01 peer to-RT-01 port '51820'
set interfaces wireguard wg01 peer to-RT-01 public-key
'EKY0dxRrSD98QHjfHOK13mZ5PJ7hnddRZt5woB3szyw='
set interfaces wireguard wg01 port '51820'
set protocols static route 192.168.1.0/24 interface wg01
```

Настройка исключений для межсетевого экрана

Чтобы трафик WireGuard проходил через внешний интерфейс, необходимо добавить исключение в правилах межсетевого экрана на всех устройствах, между которыми будет строиться WireGuard туннель.

Список команд для настойки правил межсетевого экрана:

```
set firewall name OUTSIDE_LOCAL rule 10 action accept
set firewall name OUTSIDE_LOCAL rule 10 description 'Allow established/related'
set firewall name OUTSIDE_LOCAL rule 10 state established enable
set firewall name OUTSIDE_LOCAL rule 10 state related enable
set firewall name OUTSIDE_LOCAL rule 20 action accept
set firewall name OUTSIDE_LOCAL rule 20 description WireGuard_IN
set firewall name OUTSIDE_LOCAL rule 20 destination port 51820
set firewall name OUTSIDE_LOCAL rule 20 log enable
set firewall name OUTSIDE_LOCAL rule 20 protocol udp
set firewall name OUTSIDE_LOCAL rule 20 source
```

Также необходимо убедиться, что политика межсетевого экрана *OUTISDE_LOCAL* привязана к внешнему сетевому интерфейсу для локального трафика интерфейса.

```
set interfaces ethernet eth0 firewall local name 'OUTSIDE-LOCAL'
```

> **! Предупреждение**
>
> *Настройки правил межсетевого экрана будут аналогичны для локального RT-01 и удаленного RT-02 устройств.*

После настройки правил для межсетевого экрана убедитесь, что настроенные правила разрешают трафик WireGuard туннеля:

```
fakel@RT01# ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=1.16 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=1.77 ms

fakel@RT02# ping 192.168.2.1
PING 192.168.2.1 (192.168.2.1) 56(84) bytes of data.
64 bytes from 192.168.2.1: icmp_seq=1 ttl=64 time=4.40 ms
```

```
64 bytes from 192.168.2.1: icmp_seq=2 ttl=64 time=1.02 ms
```

Настройка симметричного ключа

Поверх асимметричной криптографии может быть использован дополнительный уровень криптографии с симметричными ключами.

Для настройки симметричного ключа, выполните команду ***run generate wireguard preshared-key*** на локальном устройстве *RT-01*:

```
fakel@RT01# run generate wireguard preshared-key

rvVDOoc2IYEnV+k5p7TNAmHBMEGTHbPU8Qqg8c/sUqc=
```

Затем необходимо скопировать полученный ключ и настроить его в качестве предварительно распределённого ключа на локальном *RT-01* и удаленном *RT-02* устройствах.

> **! Предупреждение**
>
> *Поскольку это симметричный ключ, его содержимое должно быть известно только вам и вашему удаленному устройству. Обеспечьте безопасное распространение ключа.*

Список команд для настройки симметричного ключа шифрования на локальном RT-01 и удаленном RT-02 устройствах:

- `fakel@RT01# set interfaces wireguard wg01 peer to-RT-02 preshared-key 'rvVDOoc2IYEnV+k5p7TNAmHBMEGTHbPU8Qqg8c/sUqc='`
- `fakel@RT02# set interfaces wireguard wg01 peer to-RT-01 preshared-key 'rvVDOoc2IYEnV+k5p7TNAmHBMEGTHbPU8Qqg8c/sUqc='`

Основные настройки WireGuard интерфейса

- `generate wireguard default-keypair`

Создает пару ключей, включающую открытый и закрытый ключи. Пара ключей будет использоваться по умолчанию на любом настроенном интерфейсе WireGuard, даже если настраивается несколько интерфейсов.

- `generate wireguard named-keypairs <name>`

Создает именованную пару ключей <*name*>.

- `set interfaces wireguard <wgN> address <address>`

Задает адрес <*address*> для WireGuard интерфейса <*wgN*>. Адрес может быть указан несколько раз как IPv4 и/или IPv6 адрес, например, *192.0.2.1/24* и/или *2001:db8::1/64*.

- `set interfaces wireguard <wgN> description <text>`

Задает псевдоним **<text>** для сетевого интерфейса <*wgN*>. Например, псевдоним может использоваться командой show interfaces или средствами мониторинга на базе SNMP.

- `set interfaces wireguard <wgN> disable`

Отключает сетевой интерфейс <*wgN*>. После отключения интерфейс будет переведен в состояние административного отключения (A/D).

- `set interfaces wireguard <wgN> redirect <text>`

Устанавливает назначение <*text*> для пересылки входящих на интерфейс WireGuard <*wgN*> пакетов.

- `set interfaces wireguard <wgN> vrf <vrf>`

Размещает сетевой интерфейс <*wgN*> в указанном экземпляре VRF <*vrf*>.

- `set interfaces wireguard <wgN> peer <text> allowed-ips <network>`

Определяет сеть <*network*> за локальным устройством, которая будет передаваться через WireGuard интерфейс <*wgN*> на удаленное устройство <*text*>.

- `set interfaces wireguard <wgN> peer <text> address <address>`

Устанавливает IP-адрес <*address*> удаленного устройства <*text*>, с которым локальное устройство будет строить туннель через WireGuard интерфейс <*wgN*>.

- `set interfaces wireguard <wgN> peer <text> port <port>`

Определяет номер порта <*port*> удаленного устройства **<text>**, с которым локальное устройство будет строить туннель через WireGuard интерфейс <*wgN*>.

- `set interfaces wireguard <wgN> peer <text> pubkey <pubkey>`

Задает открытый ключ в кодировке base64 <*pubkey*> удаленного устройства <*text*>, с которым локальное устройство будет строить туннель через WireGuard интерфейс <*wgN*>.

- `set interfaces wireguard <wgN> port <port>`

Определяет номер порта <*port*>, по которому локальное устройство будет доступно для построения туннеля через интерфейс WireGuard <*wgN*>.

- `set interfaces wireguard <wgN> private-key <name>`

Устанавливает закрытый ключ *<name>* для использования на WireGuard интерфейсе *<wgN>*.

- `set interfaces wireguard <wgN> peer <text> persistent-keepalive <1-65535>`

Устанавливает интервал отправки сообщений проверки соединения *<1-65535>* для удаленного устройства *<text>*, с которым локальное устройство будет строить туннель через WireGuard интерфейс *<wgN>*. Значение для интервала задается в секундах.

- `set interfaces wireguard <wgN> peer <text> preshared key <preshared-key>`

Задает предварительно распределённый ключ в кодировке base64 *<preshared-key>* для удаленного устройства *<text>*, с которым локальное устройство будет строить туннель через WireGuard интерфейс *<wgN>*.

Мониторинг и эксплуатация WireGuard интерфейса

- `show interfaces wireguard`

Выводит на экран список всех интерфейсов WireGuard.

- `show interfaces wireguard <wgN>`

Выводит на экран общую информацию о конкретном интерфейсе WireGuard *<wgN>*.

- `show wireguard keypair pubkey <name>`

Выводит на экран открытую часть для указанного ключа. Это может быть как ключ по умолчанию, так и любая другая именованная пара ключей.

- `delete wireguard keypair pubkey <name>`

Удаляет пару ключей. Это может быть как ключ по умолчанию, так и любая другая именованная пара ключей.

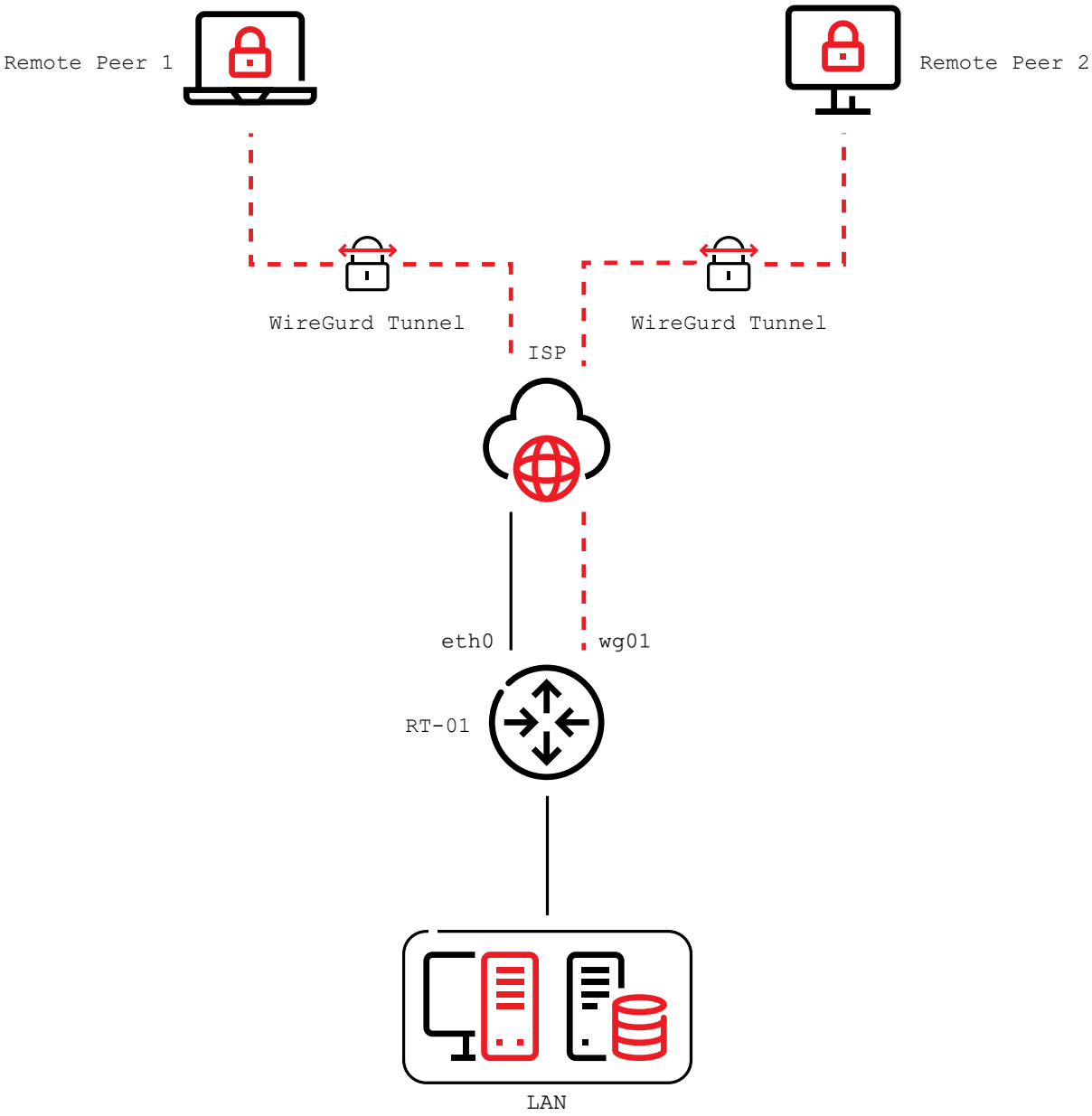
WireGuard для удаленных пользователей

Помимо построения защищенного канала между площадками типа «Site-to-Site» протокол WireGuard также используется для обеспечения безопасного доступ типа «Remote Access» удаленных узлов к сервисам в корпоративной сети.

В данном разделе описана настройка защищенной сети для безопасного доступа к ресурсам в корпоративной сети удаленных узлов с помощью протокола WireGuard.

Пример настройки WireGuard для удаленных пользователей

В качестве примера приведена настройка защищенной сети для безопасного доступа к ресурсам в корпоративной сети для двух удаленных узлов:



Параметры, используемые для настройки:

- *wg01* - WireGuard интерфейс на маршрутизаторе, который используется для подключения удаленных узлов
- *eth0* - Внешний WAN интерфейс маршрутизатора
- *89.65.211.178* - Адрес внешнего интерфейса eth0
- *172.16.155.1/24* - Адрес WireGuard интерфейса wg0

- *172.16.155.21/32* - Адрес, который будет присвоен удаленному узлу 1 при подключении по защищенному каналу
- *172.16.155.22/32* - Адрес, который будет присвоен удаленному узлу 2 при подключении по защищенному каналу
- *15 секунд* - Значение параметра проверки доступности удаленных узлов «persistent-keepalive»
- *2224* - Номер порта, по которому будут подключаться удаленные узлы WireGuard
- *RIbtUTCfgzNjnLNPQ/ulkGnnB2vMWHm7l2H/xUfbyjc=* - Публичный ключ маршрутизатора Факел
- *QQ+yJff8805ldoUsrAde1WGhF9V1a8JAqbgeCOWd6Cw=* - Публичный ключ удаленного узла 1
- *L87ytwQQMwpA+R96SH5L4voPLVxFViTRz5fCAtgnvkQ=* - Публичный ключ удаленного узла 2

Процедура настройки WireGuard для Remote Access VPN:

Создать пару ключей (Публичный + приватный):

```
generate wireguard default-keypair
```

Создать сетевой интерфейс WireGuard:

```
set interfaces wireguard wg01 address 172.16.155.1/24
```

Добавить описание для сетевого интерфейса WireGuard (опционально):

```
set interfaces wireguard wg01 description "Remote_Access_VPN"
```

Задать номер порта, по которому будут подключаться удаленные узлы WireGuard (В данном примере используется 2224 порт, но также можно указать любой другой порт по желанию):

```
set interfaces wireguard wg01 port 2224
```

Создать профиль для удаленного узла 1:

```
set interfaces wireguard wg01 peer Remote_Peer_01
```

Задать адрес, который будет присвоен удаленному узлу 1 при подключении через WireGuard:

```
set interfaces wireguard wg01 peer Remote_Peer_01 allowed-ips
172.16.155.21/32
```

Для параметра проверки доступности удаленного узла 1 задать значение равное 15 секундам:

```
set interfaces wireguard wg01 peer Remote_Peer_01 persistent-keepalive 15
```

Добавить публичный ключ для удаленного узла:

```
set interfaces wireguard wg01 peer Remote_Peer_01 pubkey QQ+yJff8805ldoUsrAde1WGhF9V1a8JAqbgeCOWd6Cw=
```

Создать профиль для удаленного узла 2:

```
set interfaces wireguard wg01 peer Remote_Peer_02
```

Задать адрес, который будет присвоен удаленному узлу 2 при подключении через WireGuard:

```
set interfaces wireguard wg01 peer Remote_Peer_02 allowed-ips 172.16.155.22/32
```

Для параметра проверки доступности удаленного узла 1 задать значение равное 15 секундам:

```
set interfaces wireguard wg01 peer Remote_Peer_02 persistent-keepalive 15
```

Добавить публичный ключ для удаленного узла:

```
set interfaces wireguard wg01 peer Remote_Peer_02 pubkey L87ytwQQMwpA+R96SH5L4voPLVxFViTRz5fCAtgnvkQ=
```

После создания интерфейса WireGuard и добавления профилей для удаленных узлов, настройте правила для Межсетевого Экрана. Для корректной работы WireGuard создайте два правила. Первое правило *Rule 10* включает состояние Established и Related для сессий, которые устанавливаются с маршрутизатором при подключении удаленных узлов. Второе правило *Rule 20* разрешает UDP трафик, который приходит на *2224* порт для установки защищенного канала между маршрутизатором и удаленными узлами: Оба правила создаются в политике, которая отвечает за фильтрацию трафика, приходящего непосредственно на внешний интерфейс маршрутизатора *WAN_LOCAL*.

Процедура настройки правил межсетевого экрана для WireGuard Remote Access VPN:

Создать правило *Rule 10*, которое включает состояние *Established* и *Related* для сессий, которые устанавливаются с маршрутизатором при подключении удаленных узлов:

```
set firewall name WAN_LOCAL rule 10 action accept
```

Добавить описание для правила *Rule 10*:

```
set firewall name WAN_LOCAL rule 10 description 'Allow
established / related'
```

Включить состояние *Established* для сессий, которые устанавливаются с маршрутизатором при подключении удаленных узлов:

```
set firewall name WAN_LOCAL rule 10 state established enable
```

Включить состояние *Related* для сессий, которые устанавливаются с маршрутизатором при подключении удаленных узлов:

```
set firewall name WAN_LOCAL rule 10 state related enable
```

Создать правило *Rule 20*, разрешающее UDP трафик, который приходит на *2224* порт для установки защищенного канала между маршрутизатором и удаленными узлами:

```
set firewall name WAN_LOCAL rule 20 action accept
```

Добавить описание для правила *Rule 20*:

```
set firewall name WAN_LOCAL rule 20 description 'WireGuard_IN'
```

Задать номер порта, по которому будут подключаться удаленные клиенты WireGuard:

```
set firewall name WAN_LOCAL rule 20 destination port 2224
```

Разрешить прохождение UDP трафика по данному правилу:

```
set firewall name WAN_LOCAL rule 20 protocol udp
```

Привязать политику *WAN_LOCAL* к внешнему сетевому интерфейсу (WAN) для цепочки, которая отвечает за трафик, приходящий непосредственно на адрес внешнего интерфейса:

```
set interfaces ethernet eth0 firewall local name WAN_LOCAL
```

После завершения настроек на маршрутизаторе, настройте WireGuard на удаленных узлах.

Пример настроек для узла 1:

```
[Interface]
```

```
PrivateKey = ARAKLSDJsadlkfjasdfiowqeruriowqeuasdf=
Address = 172.16.155.21/24
DNS = 172.18.0.10
[Peer]
PublicKey = RIbtUTCfgzNjnLNPQ/ulkGnnB2vMWHm7l2H/xUfbyjc=
AllowedIPs = 0.0.0.0/0
Endpoint = 89.65.211.178:2224
PersistentKeepalive = 25
```

Пример настроек для узла 2:

```
[Interface]
PrivateKey = 8Iasdfweirousd1EVGUk5XsT+wYFZ9mhPnQhmjzaJE6Go=
Address = 172.16.155.22/24
DNS = 172.18.0.10
[Peer]
PublicKey = RIbtUTCfgzNjnLNPQ/ulkGnnB2vMWHm7l2H/xUfbyjc=
AllowedIPs = 0.0.0.0/0
Endpoint = 89.65.211.178:2224
PersistentKeepalive = 25
```

PPPoE интерфейс

Общая информация о PPPoE интерфейсе

PPPoE — это сетевой протокол, используемый для установки соединений типа, точка-точка между двумя узлами в сети Ethernet. PPPoE обеспечивает механизм аутентификации пользователей, управления сеансами и маршрутизации данных через Ethernet-сеть.

Протокол PPPoE добавляет кадры PPP поверх Ethernet кадров. Каждый кадр PPPoE состоит из заголовка PPPoE и полезной нагрузки PPP. Заголовок PPPoE содержит информацию, необходимую для управления и маршрутизации кадров PPP, а также метаданные для управления сеансом PPPoE.

Основные характеристики протокола PPPoE

Основные характеристики протокола PPPoE включают:

- **Аутентификация и безопасность:** PPPoE обеспечивает механизм аутентификации для контроля доступа к сети. Это позволяет провайдерам услуг

Интернета обеспечить безопасный доступ для своих клиентов, требуя аутентификации с помощью уникальных учетных данных.

- **Масштабируемость и гибкость:** PPPoE позволяет обслуживать большое количество пользователей и предоставлять различные уровни услуг, включая широкополосный доступ к Интернету и специализированные сетевые сервисы.
- **Управление сеансами:** Протокол предоставляет механизм управления сеансами, позволяющий установить, поддерживать и завершать соединение по запросу клиента или провайдера услуг.
- **Динамическое присвоение IP-адресов:** PPPoE поддерживает динамическое присвоение IP-адресов клиентам при установлении сеанса. Это позволяет оптимизировать использование адресного пространства и облегчить администрирование сети.
- **Поддержка различных типов сетей:** PPPoE может использоваться в различных типах сетей, включая Ethernet, Wi-Fi и другие технологии передачи данных. Это делает его универсальным средством подключения к Интернету для различных устройств и сетевых конфигураций.
- **Ретрансляция данных:** Протокол включает в себя механизмы ретрансляции данных, который обеспечивает надежную передачу информации в случае возникновения ошибок или потерь пакетов в сети.
- **Поддержка шифрования и защиты данных:** PPPoE может использовать различные методы шифрования и защиты данных для обеспечения конфиденциальности и целостности передаваемой информации, что делает его подходящим для использования в безопасных сетевых средах.

Принцип работы PPPoE

Процесс работы протокола PPPoE может быть разбит на несколько этапов:

- **Инициализация:** Клиентский узел (обычно маршрутизатор или компьютер) инициирует соединение с сервером PPPoE (например, у поставщика услуг интернета).
- **Отправка запроса на соединение (PADI):** Клиент отправляет широковещательное сообщение PADI, чтобы найти доступные серверы PPPoE.
- **Ответ сервера (PADO):** Сервер, получивший запрос, отправляет сообщение PADO с предложением соединения клиенту.
- **Запрос на подключение (PADR):** Клиент выбирает сервер из предложенных и отправляет запрос на подключение PADR.
- **Подтверждение (PADS):** Сервер, получив запрос на подключение, отправляет подтверждение PADS клиенту.
- **Аутентификация:** Происходит обмен данными для аутентификации клиента. Это может включать в себя ввод имени пользователя и пароля.

- **Установление сеанса PPP:** После успешной аутентификации устанавливается сеанс протокола PPP между клиентом и сервером.
- **Передача данных:** После установления сеанса PPP между клиентом и сервером начинается передача данных. Это может включать передачу IP-пакетов и других сетевых данных.
- **Мониторинг сеанса:** Во время сеанса обе стороны могут отслеживать его состояние, проверяя статус подключения и обмен сетевыми пакетами.
- **Завершение сеанса:** По окончании использования клиент может завершить сеанс, отправив соответствующий запрос серверу PPPoE.
- **Завершение соединения:** После завершения сеанса сервер освобождает ресурсы, занятые для этой сессии, и закрывает соединение.
- **Обработка ошибок и ретрансляция данных:** В случае возникновения ошибок или потерь пакетов в сети, PPPoE может осуществлять ретрансляцию (повторную отправку) пакетов для гарантированной доставки информации.

Этот процесс обеспечивает эффективное и безопасное управление соединением между клиентом и сервером PPPoE, обеспечивая надежный доступ к Интернету и другим сетевым ресурсам.

Сценарии использования

Протокол PPPoE используется в различных сценариях и сетевых средах. Вот несколько примеров его применения:

Домашние сети

PPPoE часто используется в домашних сетях для подключения к Интернету через DSL, где сигнал передается по телефонной линии, или через другие технологии, такие как Ethernet.

Корпоративные сети

В корпоративных сетях PPPoE может быть использован для управления доступом сотрудников к корпоративному интернету или внутренним ресурсам, обеспечивая аутентификацию и контроль доступа.

Провайдеры услуг Интернета

ISP используют PPPoE для управления и предоставления доступа к интернет-соединениям своим клиентам. Он позволяет провайдерам контролировать и отслеживать подключения, а также предоставлять услуги аутентификации и маршрутизации.

Общественные сети доступа к Интернету

В общественных местах, таких как кафе, аэропорты или гостиницы, PPPoE может быть использован для предоставления доступа к сети Интернет через общедоступные точки доступа Wi-Fi или проводные сети.

Индустриальные и специализированные сети

PPPoE может использоваться в различных промышленных и специализированных сетях, где требуется управление доступом, аутентификация пользователей и обмен данными посредством Ethernet.

Таким образом, PPPoE широко применяется в различных сетевых средах, где необходимо управление доступом, безопасность и эффективная передача данных через Ethernet.

Пример настройки PPPoE интерфейса

Пример настройки PPPoE соединения между провайдером и устройством на базе ПО **Факел**.

Параметры для настройки PPPoE:

- Сетевой интерфейс вашего устройства на базе ПО **Факел** подключен к сетевому оборудованию провайдера.
- Для связи с провайдером не требуется настройка VLAN для сетевого интерфейса устройства на базе ПО **Факел**.
- Для настройки необходимо получить учетные данные PPPoE от вашего DSL провайдера.
- Максимальный размер MTU, который можно использовать в DSL, составляет 1492 байта. Если вы переходите от провайдера, использующего DHCP, например, кабельного, то имейте в виду, что для работы в рамках этого ограничения может потребоваться корректировка размера MTU, например, для VPN-каналов.
- Если для параметра default-route установлено значение auto, то ПО **Факел** добавит в таблицу маршрутизации шлюз по умолчанию, полученный от DSL-провайдера только в том случае, если у вас нет других WAN подключений. Если вы хотите использовать двойное WAN соединение, измените значение параметра default-route на force. Можно также установить статический интерфейс маршрут и задать для параметра default-route значение none.
- Если опция name-server установлена в значение none, то ПО **Факел** будет игнорировать серверы имен, которые присылает вам провайдер, и будет использовать те, которые вы настроили статически.

Список команд для настройки протокола PPPoE:

```
set interfaces pppoe pppoe0 default-route 'auto'
set interfaces pppoe pppoe0 mtu 1492
set interfaces pppoe pppoe0 authentication user 'userid'
set interfaces pppoe pppoe0 authentication password 'secret'
```

```
set interfaces pppoe pppoe0 source-interface 'eth0'
```

В приведенную выше конфигурацию также следует добавить настройку межсетевого экрана для работы на интерфейсе PPPoE. Для этого нужно назначить политики межсетевого экрана для всех направлений трафика на интерфейс pppoe0.

Список команд для настройки работы межсетевого экрана на интерфейсе PPPoE:

```
set interfaces pppoe pppoe0 firewall in name NET-IN
set interfaces pppoe pppoe0 firewall local name NET-LOCAL
set interfaces pppoe pppoe0 firewall out name NET-OUT
```

Примечание

Политики межсетевого экрана настраиваются индивидуально, в зависимости от типа трафика, который присутствует в вашей сети. Подробное описание настройки правил межсетевого экрана представлено в разделе Межсетевой экран.

PPPoE соединение через VLAN интерфейс

Некоторые современные провайдеры требуют построения PPPoE соединения через VLAN интерфейс. Операционная система может легко создать PPPoE сессию через VLAN-интерфейс. В следующей конфигурации PPPoE-соединение будет осуществляться через VLAN7.

Список команд для настройки построения PPPoE соединение через VLAN интерфейс:

```
set interfaces pppoe pppoe0 default-route 'auto'
set interfaces pppoe pppoe0 mtu 1492
set interfaces pppoe pppoe0 authentication user 'userid'
set interfaces pppoe pppoe0 authentication password 'secret'
set interfaces pppoe pppoe0 source-interface 'eth0.7'
```

Основные настройки PPPoE интерфейса

```
set interfaces pppoe <pppoeN> description <text>
```

Задает псевдоним <*text*> для сетевого интерфейса <*pppoeN*>. Например, псевдоним может использоваться командой ***show interfaces*** или средствами мониторинга на базе SNMP.

```
set interfaces pppoe <pppoeN> disable
```

Отключает сетевой интерфейс <*pppoeN*>. После отключения интерфейс будет переведен в состояние административного отключения (A/D).

```
set interfaces pppoe <pppoeN> vrf <vrf>
```

Размещает сетевой интерфейс <*pppoeN*> в указанном экземпляре VRF <*vrf*>.

Настройка параметров PPPoE интерфейса

```
set interfaces pppoe <pppoeN> access-concentrator <name>
```

Определяет сервер доступа <*name*> для PPPoE соединения через сетевой интерфейс <*pppoeN*>.

```
set interfaces pppoe <pppoeN> authentication user <username>
```

Устанавливает имя пользователя для аутентификации с удаленной конечной точкой для PPPoE соединения через сетевой интерфейс <*pppoeN*>.

```
set interfaces pppoe <pppoeN> authentication password <passwor>
```

Устанавливает пароль для аутентификации с удаленной конечной точкой для PPPoE соединения через сетевой интерфейс <*pppoeN*>.

```
set interfaces pppoe <pppoeN> connect-on-demand
```

Активирует режим *dial-on-demand* для PPPoE интерфейса <*pppoeN*>.

```
set interfaces pppoe <pppoeN> default-route [auto|force|none]
```

Определяет значение параметра *default-route*, который отвечает за автоматическое добавление маршрута по умолчанию для PPPoE интерфейса <*pppoeN*>:

- **default**: Маршрут по умолчанию к удаленной конечной точке добавляется автоматически при появлении соединения.
- **auto**: Маршрут по умолчанию добавляется, если в системе не существует другого маршрута по умолчанию (из любого источника).
- **force**: Маршрут по умолчанию добавляется после удаления всех существующих маршрутов по умолчанию.
- **none**: Маршрут по умолчанию не устанавливается.

Примечание

Маршрут по умолчанию добавляется только в том случае, если в системе не существует другого маршрута по умолчанию.

- `set interfaces pppoe <pppoeN> mru <mru>`

Устанавливает размер MRU <*mru*> для PPPoE интерфейса <*pppoeN*>. Значение MRU должно находиться в диапазоне от 128 до 16384. По умолчанию используется значение 1492 байта.

Примечание

При использовании протокола IPv6 размер MTU должно быть не менее 1280 байт.

- `set interfaces pppoe <pppoeN> idle-timeout <time>`

Устанавливает интервал ожидания <*time*> для PPPoE интерфейса <*pppoeN*>. При установке соединения по запросу канал активируется только при отправке трафика и отключается, если канал простаивает в течение указанного интервала.

Если этот параметр не задан или равен 0, то соединение по запросу будет оставаться включенным всегда.

- `set interfaces pppoe <pppoeN> local-address <address>`

Задает адрес локального устройства <*address*> для сеанса PPPoE через PPPoE интерфейс <*pppoeN*>. Если адрес локального устройства не задан, то он будет согласован в процессе инициализации PPPoE сессии.

- `set interfaces pppoe <pppoeN> mtu <mtu>`

Устанавливает размер MTU для PPPoE интерфейса <*pppoeN*>.

- `set interfaces pppoe <pppoeN> no-peer-dns`

Отключает использование внешних DNS серверов. При активации данного параметра используется только локальный DNS сервер.

- `set interfaces pppoe <pppoeN> remote-address <address>`

Устанавливает адрес удаленного устройства <*address*> для сеанса PPPoE через PPPoE интерфейс <*pppoeN*>. Если адрес удаленного устройства не задан, то он будет согласован в процессе инициализации PPPoE сессии.

- `set interfaces pppoe <pppoeN> service-name <name>`

Задает имя сервиса <*name*>, по которому локальный интерфейс PPPoE может выбирать сервера доступа для подключения.

```
set interfaces pppoe <pppoeN> source-interface <source-interface>
```

Определяет физический интерфейс <*source-interface*> для привязки PPPoE соединения к этому физическому интерфейсу. Каждое PPPoE соединение должно быть установлено через физический интерфейс.

Настройка параметром для протокола IPv6 на PPPoE интерфейсе

```
set interfaces pppoe <pppoeN> ipv6 address autoconf
```

Активирует механизм автоматической настройки *Stateless autoconfig* (*SLAAC*) для получения IPv6 адреса.

```
set interfaces pppoe <pppoeN> dhcpv6-options pd <id> length <length>
```

Определяет размер префикса IPv6 <*length*>, запрашиваемого у провайдера. Значение размера запрашиваемого префикса находится в диапазоне от 32 до 64. Значение по умолчанию соответствует 64.

```
set interfaces pppoe <pppoeN> dhcpv6-options pd <id> interface <delegatee>address <address>
```

Устанавливает адрес <*address*> для интерфейса <*delegatee*>, на который был делегирован префикс.

```
set interfaces pppoe <pppoeN> dhcpv6-options pd <id> interface <delegatee> sla-id<id>
```

Устанавливает значение идентификатора *Site-Level Aggregator* (*SLA*) на интерфейсе. Значение идентификатора должно быть десятичным числом, большим 0, которое укладывается в длину идентификаторов SLA.

Мониторинг и эксплуатация PPPoE интерфейса

```
show interfaces pppoe <pppoeN>
```

Выводит на экран подробную информацию об указанном интерфейсе PPPoE <*pppoeN*>.

```
show interfaces pppoe <pppoeN> queue
```

Выводит на экран информацию об очередях для указанного интерфейса PPPoE <*pppoeN*>.

- `disconnect interface <pppoeN>`

Отключает указанный интерфейс PPPoE <*pppoeN*>.

- `connect interface <pppoeN>`

Включает указанный интерфейс PPPoE <*pppoeN*>.

Маршрутизация

В ПО **Факел** маршрутизация реализована следующими механизмами:

- Статические маршруты;
- OSPF;
- IS-IS;
- RIP;
- Групповая рассылка;
- MPLS;
- Политика маршрутизации;
- Виртуальная маршрутизация.

Статические маршруты

Общая информация о статических маршрутах

Статические маршруты представляет собой маршруты, заданные вручную, которые по большей части не обновляются автоматически на основании информации о топологии сети, полученной ПО **Факел** по результатам работы других протоколов маршрутизации. Однако если в канале возникает сбой, система удалит соответствующие маршруты, включая статические маршруты из таблицы RIB, которые используются данным интерфейсом для обеспечения достижимости следующего узла. В основном статические маршруты используются для обеспечения связности в относительно простых топологиях сети или для того, чтобы переопределить результаты работы протоколов динамической маршрутизации при небольшом количестве маршрутов. Список всех маршрутов, вне зависимости от того, заданы ли они вручную или определены динамически в результате работы протоколов динамической маршрутизации, хранится в таблице RIB. Одноадресные (unicast) маршруты используются непосредственно для определения содержимого таблицы FIB, используемой при принятии решения о перенаправлении одноадресных (unicast) пакетов.

Основные настройки для статических маршрутов

```
set protocols static route <subnet> next-hop <address>
```

Создает статический маршрут IPv4 до подсети <*subnet*>, которая доступна через узел с адресом <*address*>.

```
set protocols static route <subnet> next-hop <address> disable
```

Выключает активный статический маршрут IPv4 до подсети <*subnet*>, которая доступна через узел с адресом <*address*>.

```
set protocols static route <subnet> next-hop <address> distance
<distance>
```

Задает административную дистанцию <*distance*> для статического маршрута IPv4 до подсети <*subnet*>, которая доступна через узел с адресом <*address*>.

```
set protocols static route6 <subnet> next-hop <address>
```

Создает статический маршрут IPv6 до подсети <*subnet*>, которая доступна через узел с адресом <*address*>.

```
set protocols static route6 <subnet> next-hop <address> disable
```

Выключает активный статический маршрут IPv6 до подсети <*subnet*>, которая доступна через узел с адресом <*address*>.

```
set protocols static route6 <subnet> next-hop <address>
distance <distance>
```

Задает административную дистанцию <*distance*> для статического маршрута IPv6 до подсети <*subnet*>, которая доступна через узел с адресом <*address*>.

Примечание

Для маршрута можно задать административную дистанцию (1 - 255, по умолчанию - 1). Маршруты с наименьшей административной дистанцией обрабатываются в приоритете по отношению к маршрутам с наибольшей административной дистанцией. Маршруты с административной дистанцией 255 фактический определяются операционной системой Факел как отключенные и не устанавливаются в ядро.

Настройка маршрутов на основе интерфейсов

- `set protocols static interface-route <subnet> next-hop-interface <interface>`

Создает статический маршрут IPv4 на основе интерфейса до подсети <*subnet*>, которая доступна через интерфейс с именем <*interface*>.

- `set protocols static interface-route <subnet> next-hop-interface <interface> disable`

Выключает активный статический маршрут IPv4 на основе интерфейса до подсети <*subnet*>, которая доступна через интерфейс с именем <*interface*>.

- `set protocols static interface-route <subnet> next-hop-interface <interface> distance <distance>`

Задает административную дистанцию <*distance*> для статического маршрута IPv4 на основе интерфейса до подсети <*subnet*>, которая доступна через интерфейс с именем <*interface*>.

- `set protocols static interface-route6 <subnet> next-hop-interface <interface>`

Создает статический маршрут IPv6 на основе интерфейса до подсети <*subnet*>, которая доступна через интерфейс с именем <*interface*>.

- `set protocols static interface-route6 <subnet> next-hop-interface <interface> disable`

Выключает активный статический маршрут IPv6 на основе интерфейса до подсети <*subnet*>, которая доступна через интерфейс с именем <*interface*>.

- `set protocols static interface-route6 <subnet> next-hop-interface <interface> distance <distance>`

Задает административную дистанцию <*distance*> для статического маршрута IPv6 на основе интерфейса до подсети <*subnet*>, которая доступна через интерфейс с именем <*interface*>.

Примечание

Для маршрута можно задать административную дистанцию (1 - 255, по умолчанию - 1). Маршруты с наименьшей административной дистанцией обрабатываются в приоритете по отношению к маршрутам с наибольшей административной дистанцией. Маршруты с административной дистанцией 255 фактический определяются операционной системой Факел как отключенные и не устанавливаются в ядро.

Настройка тупиковых маршрутов

Тупиковые (blackhole) маршруты используются ПО **Факел** для отбрасывания подошедших пакетов без каких-либо уведомлений об этом их отправителя. Использование тупиковых маршрутов предотвращает утечку данных о подсетях через внешние интерфейсы, но при этом не блокирует к ним доступ из локальной сети посредством точечных маршрутов.

- `set protocols static route <subnet> blackhole`

Создает тупиковый (blackhole) маршрут IPv4 до подсети <*subnet*>.

- `set protocols static route <subnet> blackhole distance <distance>`

Задает административную дистанцию <*distance*> для тупикового (blackhole) маршрута IPv4 до подсети <*subnet*>.

- `set protocols static route6 <subnet> blackhole`

Создает тупиковый (blackhole) маршрут IPv6 до подсети <*subnet*>.

- `set protocols static route6 <subnet> blackhole distance <distance>`

Задает административную дистанцию <*distance*> для тупикового (blackhole) маршрута IPv6 до подсети <*subnet*>.

> **Примечание**
>
> *Для маршрута можно задать административную дистанцию (1 - 255, по умолчанию - 1). Маршруты с наименьшей административной дистанцией обрабатываются в приоритете по отношению к маршрутам с наибольшей административной дистанцией. Маршруты с административной дистанцией 255 фактический определяются операционной системой Факел как отключенные и не устанавливаются в ядро.*

Динамическая маршрутизация

В ПО **Факел** динамическая маршрутизация реализована следующими механизмами:

- **OSPF** - Open Shortest Path First;
- **BGP** - Border Gateway Protocol;
- **IS-IS** - Intermediate System to Intermediate System;
- **RIP** - Routing Information Protocol;

- **Групповая рассылка**;
- **BFD** - Bidirectional Forwarding Detection protocol;
- **MPLS** - Multiprotocol Label Switching;
- **VRF** - Virtual routing and forwarding.

OSPF - Open Shortest Path First

Протокол OSPF является протоколом с контролем состояния соединения из категории IGP протоколов для маршрутизации внутри локальной сети, оперирующий внутри отдельной автономной системы. В настоящий момент в спецификации RFC 2328 определен протокол OSPF версии 2 (OSPFv2) для работы в сетях на базе протокола IPv4, а в спецификации RFC 5340 - протокол OSPF версии 3 (OSPFv3) для работы в сетях на базе протокола IPv6. Протокол OSPF поддерживает использование модели адресации CIDR.

OSPFv2

Общая информация о протоколе OSPFv2

OSPFv2 - протокол контроля состояния соединений, который широко применяется в сетях. Как и в любом другом протокол контроля состояния соединений, маршрутизаторы в OSPFv2 строят топологию всей сети вообще. Каждый маршрутизатор хранит копию базы знаний по топологии всей сети и запускает свой собственный алгоритм поиска кратчайшего пути по этой базе. База носит название Link State Database или LSDB. Представляет она из себя описание состояния каналов сети. Протокол OSPFv2 работает поверх протокола IPv4.

Пример настройки OSPFv2

Пример типичной настройки с использованием двух узлов, перераспределением loopback-адреса и отправкой узлом 1 маршрута по умолчанию:

Список команд для настройки первого узла:

```
set interfaces loopback lo address 10.1.1.1/32
set protocols ospf area 0 network 192.168.0.0/24
set protocols ospf default-information originate always
set protocols ospf default-information originate metric 10
set protocols ospf default-information originate metric-type 2
set protocols ospf log-adjacency-changes
set protocols ospf parameters router-id 10.1.1.1
set protocols ospf redistribute connected metric-type 2
set protocols ospf redistribute connected route-map CONNECT
```

```
set policy route-map CONNECT rule 10 action permit
set policy route-map CONNECT rule 10 match interface lo
```

Список команд для настройки второго узла:

```
set interfaces loopback lo address 10.2.2.2/32
set protocols ospf area 0 network 192.168.0.0/24
set protocols ospf log-adjacency-changes
set protocols ospf parameters router-id 10.2.2.2
set protocols ospf redistribute connected metric-type 2
set protocols ospf redistribute connected route-map CONNECT
set policy route-map CONNECT rule 10 action permit
set policy route-map CONNECT rule 10 match interface lo
```

Основные настройки протокола OSPFv2

ПО **Факел** не располагает специальной командой для запуска процесса OSPF. Процесс запускается автоматически при настройке протокола OSPF на интерфейсах.

```
set protocols ospf area <number> network <x.x.x.x/x>
```

Активирует протокол OSPF на определенном интерфейсе в случае, если его адрес находится в указанном диапазоне адресов <*x.x.x.x/x*>, после чего маршрутизатор может обмениваться информацией о топологии сети с другими устройствами, которые также являются участниками OSPF процесса и доступны через этот интерфейс. Номер области <*number*> может быть указан целым числом в диапазоне от 0 до 4294967295 или числом с точкой в формате, похожем на IP-адрес.

```
set protocols ospf auto-cost reference-bandwidth <number>
```

Задает значение полосы пропускания <*number*> в Мбит/сек для расчета веса каналов. Полоса пропускания указывается числом в диапазоне от 1 до 4294967. Значение полосы пропускания по умолчанию - 100, что соответствует весу 1. Значения меньше 100 будут также соответствовать весу 1 путем округления.

```
set protocols ospf parameters router-id <rid>
```

Задает идентификатор маршрутизатора <*rid*> для процесса OSPF. В качестве идентификатора может быть указан IP-адрес или произвольным 32-разрядным числом, но идентификатор должен быть указан обязательно. Вместе с тем идентификатор для данного маршрутизатора должен быть уникальным в рамках всего OSPF домена. В противном случае возможно возникновение ошибок в работе процесса OSPF.

Дополнительные настройки протокола OSPFv2

- `set protocols ospf default-information originate [always] [metric <number>] [metric-type <1|2>] [route-map <name>]`

Распространяет сообщение *AS-External* (type-5), содержащее маршрут по умолчанию, во все внешние маршрутизируемые области с указанием определенного значения и типа метрики. Если указан параметр always, сообщение распространяется всегда, даже если маршрут по умолчанию отсутствует в таблице маршрутизации. Параметр *route-map* позволяет указать карту маршрутов, при совпадении с которой будет распространяться сообщение с маршрутом по умолчанию.

- `set protocols ospf distance global <distance>`

Изменяет значение административной дистанции глобально для всего процесса OSPF в диапазоне от 1 до 255.

- `set protocols ospf distance ospf <external|inter-area|intra-area> <distance>`

Изменяет значение административной дистанции для определенных маршрутов: внешних маршрутов (external), маршрутов между областями (inter-area), маршрутов внутри области (intra-area). Значение административной дистанции задается в диапазоне от 1 до 255.

- `set protocols ospf log-adjacency-changes [detail]`

Регистрирует изменения в состоянии смежности. С дополнительным аргументом *detail* отображаются все изменения в состоянии смежности. Без аргумента *detail* показываются только изменения в полную или регрессию.

- `set protocols ospf max-metric router-lsa <administrative|on-shutdown <seconds>|on-startup <seconds>`

Включает поддержку *RFC 3137*, когда процесс OSPF описывает свои транзитные каналы в своем LSA как имеющие бесконечное расстояние, чтобы другие маршрутизаторы не вычисляли транзитные пути через маршрутизатор, сохраняя при этом возможность доступа к сетям через маршрутизатор. Эта поддержка может быть включена административно (и неограниченно) с помощью команды administrative. Она также может быть включена условно. Условное включение max-metric router-lsas может осуществляться на период в несколько секунд после запуска с помощью команды on-startup <seconds> и/или на период в несколько секунд до выключения с помощью команды on-shutdown <seconds>. Диапазон времени составляет от 5 до 86400.

- `set protocols ospf parameters abr type <cisco|ibm|shortcut|standard>`

Определяет модель ABR. Маршрутизатор OSPF поддерживает четыре модели ABR:

- **cisco** – маршрутизатор будет считаться ABR, если он имеет несколько настроенных связей с сетями в разных областях, одна из которых является магистральной. Причем связь с магистральной областью должна быть активной (рабочей).
- **ibm** – идентична модели «cisco», но в этом случае канал магистральной зоны может быть не активен.
- **standard** – маршрутизатор имеет несколько активных связей с различными областями.
- **shortcut** – идентична «стандартной», но в этой модели маршрутизатору разрешено использовать топологию связанных областей без привлечения магистральной области для межобластных соединений.

Примечание

Подробную информацию о различиях моделей cisco и ibm можно найти в RFC 3509. Модель shortcut позволяет ABR создавать маршруты между областями, основываясь на топологии областей, подключенных к данному маршрутизатору, но не используя магистральную область в случае, если не магистральный маршрут будет дешевле. Более подробная информация о модели shortcut приведена в файле ospf-shortcut-abr-02.txt.

```
set protocols ospf parameters rfc1583-compatibility
```

Включает поддержку *RFC 1583*.

```
set protocols ospf passive-interface <interface>
```

Определяет интерфейс <*interface*> как пассивный. Пассивный интерфейс рекламирует свой адрес, но не выполняет протокол OSPF (смежности не формируются и пакеты hello не генерируются).

```
set protocols ospf passive-interface default
```

Определяет все интерфейсы как пассивные по умолчанию. Поскольку данная команда изменяет логику конфигурации на пассивную по умолчанию, поэтому интерфейсы, на которых ожидается наличие смежных маршрутизаторов, должны быть настроены с помощью команды *passive-interface exclude*.

```
set protocols ospf passive-interface-exclude <interface>
```

Исключает интерфейс из пассивного состояния. Данная команда используется, если была настроена команда *passive-interface default*.

```
set protocols ospf refresh timers <seconds>
```

Маршрутизатор автоматически обновляет информацию о состоянии связей со своими соседями. Обновляется только та устаревшая информация, возраст которой превысил определенный порог. Данный параметр изменяет пороговое значение, которое по умолчанию составляет 1800 секунд (полчаса). Значение применяется ко всему маршрутизатору OSPF. Диапазон значений таймера составляет от 10 до 1800.

```
set protocols ospf timers throttle spf <delay|initial-holdtime
|max-holdtime> <seconds>
```

Задает начальную задержку, начальное время задержки и максимальное время задержки между моментом расчета SPF и событием, вызвавшим расчет. Время задается в миллисекундах и должно находиться в диапазоне от 0 до 600000 миллисекунд.

- **delay** задает начальную задержку расписания SPF в миллисекундах. Значение по умолчанию равно 200 мс.
- **initial-holdtime** задает минимальное время удержания между двумя последовательными вычислениями SPF. Значение по умолчанию равно 1000 мс.
- **max-holdtime** задает максимальное время ожидания между двумя последовательными SPF-расчетами. Значение по умолчанию равно 10000 мс.

Настройки областей OSPFv2

```
set protocols ospf area <number> area-type stub
```

Определяет область <*number*> как Stub Area. То есть область, в которой ни один маршрутизатор не прокладывает внешние для OSPF маршруты, и, следовательно, все внешние маршруты проходят через ABR. Следовательно, ABR для такой области не нужно передавать в нее AS-External LSAs (тип-5) или ASBR-Summary LSAs (тип-4). Они должны передавать в такую область только Network-Summary (тип-3) LSA вместе с резюме маршрута по умолчанию.

```
set protocols ospf area <number> area-type stub no-summary
```

Определяет область <*number*> как Totally Stub Area. В дополнение к ограничениям области stub этот тип области не позволяет ABR инжектировать Network-Summary (тип-3) LSA в указанную область stub. Разрешен только суммарный маршрут по умолчанию.

```
set protocols ospf area <number> area-type stub default-cost
<number>
```

Устанавливает стоимость LSA по умолчанию, анонсируемых в stubby-областях. Диапазон стоимости составляет от 0 до 16777215.

```
set protocols ospf area <number> area-type nssa
```

Определяет область как Not So Stubby Area. Информация о внешней маршрутизации импортируется в NSSA в виде LSA типа 7. LSA типа 7 аналогичны LSA типа 5 AS-external, за исключением того, что они могут быть переданы только в NSSA. Для дальнейшего распространения внешней информации NSSA LSA Type-7 должен быть транслирован в Type-5 AS-external-LSA на NSSA ABR.

- `set protocols ospf area <number> area-type nssa no-summary`

Определяет область как NSSA Totally Stub Area. ABR для такой области не должны передавать в нее LSA Network-Summary (тип-3) (кроме суммарного маршрута по умолчанию), ASBR-Summary LSA (тип-4) и AS-External LSA (тип-5). Однако LSA типа-7, преобразованные в тип-5 на ABR NSSA, разрешены.

- `set protocols ospf area <number> area-type nssa default-cost <number>`

Устанавливает стоимость по умолчанию для LSA, анонсируемых в области NSSA. Диапазон стоимости составляет от 0 до 16777215.

- `set protocols ospf area <number> area-type nssa translate <always |candidate|never>`

Указывает, будет ли данный пограничный маршрутизатор NSSA безусловно транслировать LSA типа 7 в LSA типа 5. Если роль - *Always*, LSA типа 7 транслируются в LSA типа 5 независимо от состояния транслятора других пограничных маршрутизаторов NSSA. Если роль - *Candidate*, то этот маршрутизатор участвует в выборах транслятора, чтобы определить, будет ли он выполнять обязанности транслятора. Когда роль равна *Never*, этот маршрутизатор никогда не будет транслировать LSA типа 7 в LSA типа 5.

- `set protocols ospf area <number> authentication plaintext-password`

Определяет, что для данной области должна использоваться аутентификация с помощью простого пароля. Пароль также должен быть настроен для каждого интерфейса.

- `set protocols ospf area <number> authentication md5`

Указывает, что пакеты OSPF должны осуществлять аутентификацию с помощью MD5 HMAC в пределах заданной области. Ключевой материал также должен быть сконфигурирован на основе каждого интерфейса.

- `set protocols ospf area <number> range <A.B.C.D/M> [cost <number>]`

Суммирует пути из указанной зоны в один summary-LSA (Type-3), анонсированный в другие зоны. Эта команда может быть использована только в ABR, и суммировать

можно только маршрутные-LSA (тип-1) и сетевые-LSA (тип-2) (т. е. LSA с областью действия). AS-external-LSA (Type-5) не могут быть суммированы - их область видимости - AS. Необязательный аргумент cost задает метрику агрегированного соединения. Диапазон значений метрики - от 0 до 16777215.

- `set protocols ospf area <number> range <A.B.C.D/M> not-advertise`

Фильтрует внутриобластные пути - т. е. внутриобластные пути из этого диапазона не рекламируются в другие области. Эта команда имеет смысл только в ABR.

- `set protocols ospf area <number> range <A.B.C.D/M> substitute <E.F.G.H/M>`

Анонсирует в магистральную область один summary-LSA типа 3 с информацией о маршрутизации *<E.F.G.H/M>*, если определенная область содержит хотя бы одну внутриобластную сеть (т. е. описанную с помощью router-LSA или network-LSA) из диапазона *<A.B.C.D/M>*. Данная команда имеет смысл только в ABR.

- `set protocols ospf area <number> shortcut <default|disable|enable>`

Сокращает маршруты (не магистральные) для межрайонных маршрутов. Для сокращения маршрутов существует три режима:

- **default** – эта область будет использоваться для короткого замыкания только в том случае, если ABR не имеет связи с магистральной областью или эта связь была потеряна.
- **enable** – область будет использоваться для сокращения пути каждый раз, когда проходящий через нее маршрут будет дешевле.
- **disable** – эта область никогда не используется ABR для сокращения маршрутов.

- `set protocols ospf area <number> virtual-link <A.B.C.D>`

Обеспечивает когерентность магистральной области за счет создания виртуальных каналов связи.

В общем случае протокол OSPF требует, чтобы магистральная область (область 0) была целостной и полностью связанной. Т. е. любой маршрутизатор магистральной области должен иметь маршрут к любому другому маршрутизатору магистральной области. Более того, каждый ABR должен иметь связь с магистральной областью. Однако не всегда возможно иметь физическую связь с магистральной областью. В этом случае между двумя ABR (один из которых имеет связь с магистральной областью) в области (не stub area) организуется виртуальная связь.

- *<number>* – идентификатор области, через которую проходит виртуальный канал связи.

- *<A.B.C.D>* – ABR router-id, с которым устанавливается виртуальная связь. Виртуальная связь должна быть настроена на обоих маршрутизаторах.

Формально виртуальный канал выглядит как сеть «точка-точка», соединяющая два ABR из одной области, один из которых физически подключен к магистральной области. Эта виртуальная сеть считается принадлежащей магистральной области.

Настройки интерфейсов для OSPFv2

- `set interfaces <inttype> <interface> ip ospf authentication plaintext-password <text>`

Устанавливает ключ аутентификации OSPF в виде простого пароля. После установки все пакеты OSPF будут аутентифицированы. Длина ключа может составлять до 8 символов.

Аутентификация с помощью простого текстового пароля небезопасна и отменена в пользу аутентификации с помощью MD5 HMAC.

- `set interfaces <inttype> <interface> ip ospf authentication md5 key-id <id> md5-key<text>`

Указывает, что на данном интерфейсе должна использоваться аутентификация MD5 HMAC. Она устанавливает ключ аутентификации OSPF в виде криптографического пароля. Key-id определяет секретный ключ, используемый для создания дайджеста сообщения. Этот идентификатор является частью протокола и должен быть одинаковым для всех маршрутизаторов на линии. Ключ может иметь длину до 16 символов (более длинные строки будут усечены) и ассоциируется с заданным key-id.

- `set interfaces <inttype> <interface> ip ospf bandwidth <number>`

Задает пропускную способность интерфейса для расчета стоимости, где пропускная способность может быть в диапазоне от 1 до 100000, указана в Мбит/с.

- `set interfaces <inttype> <interface> ip ospf cost <number>`

Устанавливает стоимость соединения для указанного интерфейса. Значение стоимости устанавливается в поле метрики маршрутизатора-LSA и используется для расчета SPF. Диапазон значений стоимости составляет от 1 до 65535.

- `set interfaces <inttype> <interface> ip ospf dead-interval <number>`

Задает количество секунд для значения таймера Dead Interval маршрутизатора, используемого для таймера ожидания и таймера бездействия. Это значение должно быть одинаковым для всех маршрутизаторов, подключенных к общей сети. Значение по умолчанию составляет 40 секунд. Диапазон интервалов составляет от 1 до 65535.

- `set interfaces <inttype> <interface> ip ospf hello-interval <number>`

Устанавливает количество секунд для значения таймера Hello Interval. При установке этого значения пакет Hello будет отправляться на указанный интерфейс каждые несколько секунд по таймеру. Это значение должно быть одинаковым для всех маршрутизаторов, подключенных к общей сети. Значение по умолчанию составляет 10 секунд. Диапазон значений интервала составляет от 1 до 65535.

- `set interfaces <inttype> <interface> ip ospf mtu-ignore`

Отключает проверку значения MTU в пакетах OSPF DBD. Таким образом, использование этой команды позволяет OSPF-связке достичь состояния FULL, даже если между двумя OSPF-маршрутизаторами имеется несоответствие MTU интерфейса.

- `set interfaces <inttype> <interface> ip ospf network <type>`

Указывает тип распределения для сети, подключенной к данному интерфейсу:

- **broadcast** – распределение широковещательных IP-адресов.
- **non-broadcast** – распределение адресов в топологии сетей NBMA.
- **point-to-multipoint** – распределение адресов в сетях «точка-многоточечное соединение.
- **point-to-point** – распределение адресов в сетях «точка-точка.

- `set interfaces <inttype> <interface> ip ospf priority <number>`

Устанавливает целочисленное значение *Router Priority*. Маршрутизатор с наивысшим приоритетом будет иметь больше шансов стать назначенным маршрутизатором. Если установить значение 0, то маршрутизатор не будет иметь права стать назначенным маршрутизатором. Значение по умолчанию равно 1. Диапазон интервалов от 0 до 255.

- `set interfaces <inttype> <interface> ip ospf retransmit-interval <number>`

Задает количество секунд для значения таймера *RxmtInterval*. Это значение используется при повторной передаче пакетов Database Description и Link State Request, если подтверждение не было получено. Значение по умолчанию равно 5 секундам. Диапазон интервалов составляет от 3 до 65535.

- `set interfaces <inttype> <interface> ip ospf transmit-delay <number>`

Задает количество секунд для значения *InfTransDelay*. Она позволяет установить и настроить для каждого интерфейса интервал задержки перед началом процесса

синхронизации базы данных маршрутизатора со всеми соседями. Значение по умолчанию составляет 1 секунду. Диапазон значений интервала составляет от 3 до 65535.

Настройка OSPFv2 соседства вручную

Обычно устройства маршрутизации OSPF обнаруживают своих соседей динамически, прослушивая широковещательные или многоадресные пакеты hello в сети. Поскольку сеть NBMA не поддерживает широковещание (или многоадресную рассылку), устройство не может обнаружить своих соседей динамически, поэтому все соседи должны быть настроены статически.

```
set protocols ospf neighbor <A.B.C.D>
```

Задает IP-адрес *<A.B.C.D>* соседнего устройства.

```
set protocols ospf neighbor <A.B.C.D> poll-interval <seconds>
```

Устанавливает продолжительность времени *<seconds>*, в течение которого устройство маршрутизации посылает пакеты hello с интерфейса до установления смежности с соседом. Диапазон значений составляет от 1 до 65535 секунд. Значение по умолчанию - 60 секунд.

```
set protocols ospf neighbor <A.B.C.D> priority <number>
```

Задает значение приоритета маршрутизатора для соседа, связанного с указанным IP-адресом. По умолчанию это значение равно 0. Это ключевое слово не применяется к интерфейсам типа «точка-многоточечное соединение».

Настройка редистрибуции для OSPFv2

```
set protocols ospf redistribute <route source>
```

Перераспределяет информацию о маршрутизации из заданного источника маршрута в процесс OSPF. Для источника маршрута доступно пять режимов: bgp, connected, kernel, rip, static.

```
set protocols ospf default-metric <number>
```

Задает значение метрики по умолчанию для перераспределенных маршрутов. Диапазон значений метрики составляет от 0 до 16777214.

```
set protocols ospf redistribute <route source> metric <number>
```

Задает метрику для перераспределенных маршрутов из заданного источника маршрута. Для источника маршрута доступно пять режимов: bgp, connected, kernel, rip, static. Диапазон значений метрики составляет от 1 до 16777214.

- `set protocols ospf redistribute <route source> metric-type <1|2>`

Задает тип метрики для перераспределенных маршрутов. Разница между двумя типами метрик заключается в том, что метрика типа 1 — это метрика, которая «соизмерима» с внутренними связями OSPF. При расчете метрики до внешнего пункта назначения полная метрика пути вычисляется как сумма метрик пути маршрутизатора, который рекламировал эту связь, плюс метрика связи. Таким образом, будет выбран маршрут с наименьшей суммарной метрикой. Если внешняя связь рекламируется с метрикой типа 2, то выбирается путь через маршрутизатор, рекламировавший эту связь с наименьшей метрикой несмотря на то, что внутренний путь к этому маршрутизатору длиннее (с большей стоимостью). Однако если два маршрутизатора рекламируют внешнюю связь и с метрикой типа 2, то предпочтение отдается пути, проходящему через маршрутизатор с более коротким внутренним путем. Если два разных маршрутизатора рекламируют два канала к одному и тому же внешнему пункту назначения, но с разным типом метрики, то предпочтение отдается маршрутизатору с метрикой типа 1. Если тип метрики не определен, то маршрутизатор будет считать, что эти внешние ссылки по умолчанию имеют метрику типа 2.

- `set protocols ospf redistribute <route source> route-map <name>`

Устанавливает route map <*name*> для фильтрации перераспределенных маршрутов из заданного источника маршрута. Для источника маршрута доступно пять режимов: bgp, connected, kernel, rip, static.

Мониторинг состояния OSPFv2

- `show ip ospf neighbor`

Выводит информацию о состояние соседей.

- `show ip ospf neighbor detail`

Выводит информацию о соседях в подробном виде.

- `show ip ospf neighbor <A.B.C.D>`

Выводит в подробном виде информацию для конкретного соседа <*A.B.C.D*>.

- `show ip ospf neighbor <interface>`

Выводит в подробном виде информацию для соседа на указанном интерфейсе <*interface*>.

- `show ip ospf interface <interface>`

Выводит информацию о состоянии и конфигурации OSPF указанного интерфейса <*interface*> или всех интерфейсов, если интерфейс не указан.

- `show ip ospf route`

Выводит таблицу маршрутизации OSPF, определенную по последнему расчету SPF.

- `show ip ospf border-routers`

Выводит таблицу путей к маршрутизаторам границы области и границы автономной системы.

- `show ip ospf database`

Выводит сводную таблицу с содержимым базы данных (LSA).

- `show ip ospf database <type> [A.B.C.D] [adv-router <A.B.C.D>|self-originate]`

Выводит содержимое базы данных для определенного типа анонсируемых соединений.

Тип может быть следующим: asbr-summary, external, network, nssa-external, opaque-area, opaque-as, opaque-link, router, ummary.

- `show ip ospf database max-age`

Выводит информацию об LSA в списке MaxAge.

OSPFv3

Общая информация о протоколе OSPFv3

Для работы в сетях IPv6 разработан бесклассовый протокол маршрутизации по состоянию канала OSPFv3, большинство параметров которого аналогичны протоколу OSPFv2, работающему в сетях IPv4. Оба протокола работают на основе алгоритма SPF. При обмене маршрутной информацией рассылаются те же типы пакетов, что и в версии протокол OSPFv6. Рассылка маршрутной информации в OSPFv3 производится с использованием группового адреса FF02::5 или FF02::6 и маршрутизаторов DR и BDR. Кроме того, для рассылки внутри локального канала используются индивидуальные локальные адреса источника и назначения. OSPFv3 использует аутентификацию IPv6 по протоколу IPSec.

Пример настройки OSPFv3

Пример типичной конфигурация OSPFv3 с использованием двух узлов.

Список команд для настройки первого узла:

```
set protocols ospfv3 area 0.0.0.0 interface eth1
set protocols ospfv3 area 0.0.0.0 range 2001:db8:1::/64
set protocols ospfv3 parameters router-id 192.168.1.1
set protocols ospfv3 redistribute connected
```

Список команд для настройки второго узла:

```
set protocols ospfv3 area 0.0.0.0 interface eth1
set protocols ospfv3 area 0.0.0.0 range 2001:db8:2::/64
set protocols ospfv3 parameters router-id 192.168.2.1
set protocols ospfv3 redistribute connected
```

Чтобы получить информацию о редистрибуции маршрутов, используйте команду ***show ipv6 ospfv3 redistribute***.

> **Примечание**
>
> *Чтобы перераспределить маршруты IPv6 через OSPFv3 на канале интерфейса WireGuard, настройте вручную адреса link-local на интерфейсах WireGuard*

Пример настройки OSPFv3 для интерфейсов WireGuard

Список команд для настройки первого узла:

```
set interfaces wireguard wg01 address
'fe80::216:3eff:fe51:fd8c/64'
set interfaces wireguard wg01 address '192.168.0.1/24'
set interfaces wireguard wg01 peer ospf02 allowed-ips '::/0'
set interfaces wireguard wg01 peer ospf02 allowed-ips
'0.0.0.0/0'
set interfaces wireguard wg01 peer ospf02 endpoint
'10.1.1.101:12345'
set interfaces wireguard wg01 peer ospf02 pubkey 'ie3...='
set interfaces wireguard wg01 port '12345'
set protocols ospfv3 parameters router-id 192.168.1.1
set protocols ospfv3 area 0.0.0.0 interface 'wg01'
set protocols ospfv3 area 0.0.0.0 interface 'lo'
```

Список команд для настройки второго узла:

```
set interfaces wireguard wg01 address 'fe80::216:3eff:fe0a:7ada/64'
set interfaces wireguard wg01 address '192.168.0.2/24'
set interfaces wireguard wg01 peer ospf01 allowed-ips '::/0'
set interfaces wireguard wg01 peer ospf01 allowed-ips '0.0.0.0/0'
set interfaces wireguard wg01 peer ospf01 endpoint '10.1.1.100:12345'
set interfaces wireguard wg01 peer ospf01 pubkey 'NHI...='
set interfaces wireguard wg01 port '12345'
set protocols ospfv3 parameters router-id 192.168.1.2
set protocols ospfv3 area 0.0.0.0 interface 'wg01'
set protocols ospfv3 area 0.0.0.0 interface 'lo'
```

Для проверки состояния между соседствующими узлами, используйте команду ***show ipv6 ospfv3 neighbor***.

Пример вывода информации о состоянии соседствующих узлов:

```
fakel@ospf01:~$ show ipv6 ospfv3 neighbor

Neighbor ID     Pri    DeadTime    State/IfState         Duration I/F[State]
192.168.0.2       1    00:00:37     Full/PointToPoint    00:18:03 wg01[PointToPoint]


fakel@ospf02# run show ipv6 ospfv3 neighbor

Neighbor ID     Pri    DeadTime    State/IfState         Duration I/F[State]
192.168.0.1       1    00:00:39     Full/PointToPoint    00:19:44 wg01[PointToPoint]
```

Основные настройки OSPFv3

В ПО **Факел** нет специальной команды для запуска процесса OSPFv3. Процесс OSPFv3 запускается при настройке первого интерфейса с поддержкой ospf.

```
set protocols ospfv3 area <number> interface <interface>
```

Активирует протокол OSPFv3 на определенном интерфейс <*interface*>. Номер области <*number*> может быть указан в десятичной системе счисления в диапазоне от 0 до 4294967295. Или он может быть указан в десятичной системе счисления, аналогично ip-адресу.

```
set protocols ospfv3 parameters router-id <rid>
```

Задает идентификатор маршрутизатора <*rid*> процесса OSPFv3. Идентификатор маршрутизатора может быть IP-адресом маршрутизатора, но не обязательно — это может быть любое произвольное 32-битное число.

Дополнительные настройки OSPFv3

```
set protocols ospfv3 distance global <distance>
```

Изменяет глобальное значение расстояния <*distance*> в протоколе OSPFv3. Диапазон расстояний составляет от 1 до 255.

```
set protocols ospfv3 distance ospfv3 <external|inter-
area|intra-area> <distance>
```

Изменяет значение расстояния <*distance*> в протоколе OSPFv3. Аргументами являются значения расстояния для внешних маршрутов <*external*>, маршрутов, распространяемых между зонами <*inter-area*> и маршрутов, распространяемых внутри одной зоны <*intra-area*> соответственно. Диапазон значений расстояния составляет от 1 до 255.

Настройки областей OSPFv3

```
set protocols ospfv3 area <number> range <prefix>
```

Суммирует маршруты, распространяемые внутри одной зоны в один маршрут, распространяемый между зонами с префиксным типом LSA Type-3, анонсированный в другие зоны. Данная команда может быть использована только в ABR.

```
set protocols ospfv3 area <number> range <prefix> not-advertise
```

Фильтрует внутриобластные пути - т. е. внутриобластные пути из этого диапазона не рекламируются в другие области. Эта команда имеет смысл только в ABR.

```
set protocols ospfv3 area <number> range <prefix> not-advertise
```

Настройки интерфейсов для OSPFv3

```
set interfaces <inttype> <interface> ipv6 ospfv3 cost <number>
```

Устанавливает стоимость соединения <*number*> для указанного интерфейса <*interface*>. Значение стоимости устанавливается в поле метрики маршрутизатора-LSA

и используется для расчета SPF. Диапазон значений стоимости составляет от 1 до 65535.

- `set interfaces <inttype> <interface> ipv6 ospfv3 dead-interval <number>`

Задайте количество секунд для значения таймера Dead Interval маршрутизатора, используемого для таймера ожидания и таймера бездействия. Это значение должно быть одинаковым для всех маршрутизаторов, подключенных к общей сети. Значение по умолчанию составляет 40 секунд. Диапазон интервалов составляет от 1 до 65535.

- `set interfaces <inttype> <interface> ipv6 ospfv3 hello-interval <number>`

Установка количества секунд для значения таймера Hello Interval **<number>**. При установке этого значения пакет Hello будет отправляться на указанный интерфейс каждые несколько секунд по таймеру. Это значение должно быть одинаковым для всех маршрутизаторов, подключенных к общей сети. Значение по умолчанию составляет 10 секунд. Диапазон значений интервала составляет от 1 до 65535.

- `set interfaces <inttype> <interface> ipv6 ospfv3 mtu-ignore`

Отключает проверку значения MTU в пакетах OSPF DBD. Таким образом, использование этой команды позволяет OSPF-связке достичь состояния FULL, даже если между двумя OSPF-маршрутизаторами имеется несоответствие MTU интерфейса.

- `set interfaces <inttype> <interface> ipv6 ospfv3 network <type>`

Определяет тип распределения для сети <type>, подключенной к определенному интерфейсу *<inttype>*:

- *broadcast – распределение широковещательных IP-адресов;*
- *point-to-point – распределение адресов в сетях «точка-точка.*

- `set interfaces <inttype> <interface> ipv6 ospfv3 priority <number>`

Задает целочисленное значение Router Priority *<number>*. Маршрутизатор с наивысшим приоритетом будет иметь больше шансов стать назначенным маршрутизатором. Если установить значение 0, то маршрутизатор не будет иметь права стать назначенным маршрутизатором. Значение по умолчанию равно 1. Диапазон интервалов от 0 до 255.

- `set interfaces <inttype> <interface> ipv6 ospfv3 passive`

Определяет интерфейс *<interface>* как пассивный. Пассивный интерфейс рекламирует свой адрес, но не выполняет протокол OSPF (смежности не формируются и пакеты hello не генерируются).

- `set interfaces <inttype> <interface> ipv6 ospfv3 retransmit-interval <number>`

Задает количество секунд для значения таймера *RxmtInterval* <*number*>. Это значение используется при повторной передаче пакетов Database Description и Link State Request, если подтверждение не было получено. Значение по умолчанию равно 5 секундам. Диапазон интервалов составляет от 3 до 65535.

- `set interfaces <inttype> <interface> ipv6 ospfv3 transmit-delay <number>`

Задает количество секунд для значения *InfTransDelay* <*number*>. Позволяет установить и настроить для каждого интерфейса интервал задержки перед началом процесса синхронизации базы данных маршрутизатора со всеми соседями. Значение по умолчанию составляет 1 секунду. Диапазон значений интервала составляет от 3 до 65535.

Настройка редистрибуции для OSPFv3

- `set protocols ospfv3 redistribute <route source>`

Перераспределяет информацию о маршрутизации из заданного источника маршрута в процесс OSPF. Для источника маршрута доступно пять режимов: bgp, connected, kernel, rip, static.

- `set protocols ospf redistribute <route source> route-map <name>`

Устанавливает route map <*name*> для фильтрации перераспределенных маршрутов из заданного источника маршрута <*route source*>. Для источника маршрута доступно пять режимов: bgp, connected, kernel, rip, static.

Мониторинг состояния OSPFv3

- `show ipv6 ospfv3 neighbor`

Выводит информацию о состоянии соседей.

- `show ipv6 ospfv3 neighbor detail`

Выводит информацию о соседях в подробном виде.

- `show ipv6 ospfv3 neighbor <A.B.C.D>`

Выводит в подробном виде информацию для конкретного соседа <*A.B.C.D*>.

- `show ipv6 ospfv3 neighbor <interface>`

Выводит в подробном виде информацию для соседа на указанном интерфейсе *<interface>*.

- `show ipv6 ospfv3 interface [prefix]|[<interface> [prefix]]`

Выводит информацию о состоянии и настройках OSPF для указанного интерфейса *<interface>* или для всех интерфейсов, если конкретный интерфейс не указан. С аргументом prefix эта команда также выводит информацию о префиксах.

- `show ipv6 ospfv3 route`

Выводит таблицу маршрутизации OSPF, определенную по последнему расчету SPF.

- `show ipv6 ospfv3 border-routers`

Выводит таблицу путей к маршрутизаторам, граничащим с областью и автономной системой.

- `show ipv6 ospfv3 database`

Выводит на экран сводную таблицу с содержимым базы данных (LSA).

- `show ipv6 ospfv3 database <type> [A.B.C.D] [adv-router <A.B.C.D> |self-originate]`

Выводит содержимое базы данных для определенного типа анонсированного канала.

- `show ipv6 ospfv3 redistribute`

Выводит информацию о редистрибуции маршрутов в OSPFv3

BGP - Border Gateway Protocol

Общая информация о протоколе BGP

BGP (Border Gateway Protocol) — это основной протокол динамической маршрутизации, который используется в Интернете.

Маршрутизаторы, использующие протокол BGP, обмениваются информацией о доступности сетей. Вместе с информацией о сетях передаются различные атрибуты этих сетей, с помощью которых BGP выбирает лучший маршрут и настраиваются политики маршрутизации.

ПО **Факел** использует для реализации протокола BGP механизм FRR.

Примечание

Примечание

В ПО Факел нет специальной команды для запуска процесса, реализующего протокол BGP. Процесс запускается при настройке первого BGP соседа.

Автономные системы

Из спецификации *RFC 1930* следует, что автономная система (далее - АС) представляет собой связанную группу из одного или нескольких IP префиксов, управляемую одним или несколькими провайдерами, которая имеет единую и четко определенную политику маршрутизации.

Каждая АС имеет связанный с ней идентификационный номер, обозначаемый как ASN. Это двухбайтовое значение от 1 до 65535. Номера АС с 64512 по 65535 определяются как частные номера АС. Частные номера АС не должны анонсироваться в глобальной сети Интернет. Диапазон двухбайтовых номеров АС исчерпан. Четырехбайтные номера АС определены в спецификации *RFC 6793* и предоставляют собой пул из 4294967296 номеров.

ASN является одним из основных элементов протокола BGP. Протокол BGP является протоколом дистанционно-векторной маршрутизации, а механизм AS-Path обеспечивает метрику для дистанционно-векторной маршрутизации, а также обнаружение петель в протоколе BGP.

Выбор маршрутов

Процесс выбора маршрута, используемый в реализации протокола BGP механизмом FRR, использует следующие критерии принятия решения, начиная с верхней части списка и двигаясь к нижней до тех пор, пока один из критериев не будет определен как подходящий:

1. **Проверка веса маршрутов** - выбор в пользу маршрутов с большим локальным весом.
2. **Проверка локальных предпочтений** - выбор в пользу маршрутов с большими локальными предпочтениями.
3. **Проверка локальных маршрутов** - выбор в пользу локальных маршрутов (статических, агрегированных, перераспределенных) вместо полученных.
4. **Проверка длины пути до АС** - выбор в пользу меньшего числа пересылок на пути до АС.
5. **Проверка источника** - выбор в пользу маршрута с наименьшего типом источника. То есть предпочтение отдается маршрутам IGP, а не EGP и неполным маршрутам.
6. **Проверка атрибута MED** - если маршруты с атрибутом MED были получены из одной и той же АС, предпочтение отдается маршруту с наименьшим значением атрибута MED.
7. **Проверка внешних источников маршрутов** - выбор в пользу маршрута, полученного от внешнего eBGP соседа, вместо маршрутов, полученных от других типов соседей.

8. **Проверка IGP стоимости** - выбор в пользу маршрута с меньшей IGP стоимостью.
9. **Проверка множественных путей** - если включено использование множественных путей, то проверяется, можно ли считать равными маршруты, еще не различающиеся по предпочтениям. Если установлено значение параметра bgp bestpath as-path multipath-relax, то все такие маршруты считаются равными, в противном случае равными считаются маршруты, полученные через iBGP с одинаковыми AS-Path, или маршруты, полученные от соседей eBGP в той же АС.
10. **Проверка уже выбранного внешнего маршрута** - если оба маршрута получены от eBGP соседей, то предпочтение отдается тому маршруту, который уже выбран. Необходимо учесть, что эта проверка не применяется, если настроен параметр bgp bestpath compare-routerid. Эта проверка может предотвратить некоторые случаи колебаний.
11. **Проверка параметра идентификатора маршрутизатора** - предпочтение отдается маршруту с наименьшим идентификатором router-ID. Если маршрут имеет атрибут ORIGINATOR_ID, полученный через механизм отражения маршрутов iBGP, то используется идентификатор данного маршрутизатора, в противном случае используется идентификатор router-ID соседа, от которого был получен маршрут.
12. **Проверка длины списка кластеризации** - используется маршрут с наименьшей длиной списка кластеризации. Список кластеризации определяет путь отражения маршрута в iBGP.
13. **Проверка адреса соседа** - предпочтение отдается маршруту, полученному от соседа с более высоким адресом транспортного уровня, в качестве критерия при прочих равных результатах.

Согласование возможностей

При добавлении функции обмена маршрутной информации по протоколу IPv6 в контексте протокола BGP было несколько предложений. IETF IDR приняла предложение под названием Multiprotocol Extension for BGP. Описание данного подхода представлено в спецификации RFC 2283. В спецификации не определяются новые протоколы. Вместо этого определяются новые атрибуты для существующей реализации протокола BGP. Когда протокол BGP используется для обмена маршрутной информацией по протоколу IPv6, он обозначается как BGP-4+. Когда он используется для обмена информацией о многоадресной маршрутизации, он обозначается как MBGP.

Сервис bgpd в составе ПО **Факел** поддерживает мультипротокольное расширение для BGP. Таким образом, если удаленный сосед поддерживает этот протокол, сервис bgpd может обмениваться маршрутной информации по протоколу IPv6 и/или с помощью многоадресной рассылке.

Стандартная реализация протокола BGP не предусматривала возможности определения функций удаленного соседа: например, может ли он работать с префиксами других типов, кроме одноадресных маршрутов IPv4. Это было большой проблемой при использовании Multiprotocol Extension for BGP в реальных сетях. В спецификации RFC 2842 была принята функция, называемую Capability Negotiation. Сервис bgpd использует эту функцию для определения функций удаленного соседа.

Если сосед сконфигурирован только как узел с одноадресными коммуникациями по протоколу IPv4, сервис bgpd не посылает пакеты Capability Negotiation до тех пор, пока не используются другие дополнительные функции протокола BGP, которые не требуют согласования возможностей.

По умолчанию механизм FRR в составе ПО **Факел** устанавливает соединение с соседями, используя минимальный набор общих функций для обеих сторон. Например, если локальный маршрутизатор использует функцию одноадресной и многоадресной рассылки, а удаленный маршрутизатор использует только одноадресную рассылку, то локальный маршрутизатор установит соединение с возможностью только одноадресной рассылки. При отсутствии общих функциях для обеих сторон коммуникации механизм FRR возвращает ошибку Unsupported Capability и затем сбрасывает соединение.

Пример настройки BGP

Для протокола IPv4

Пример простой конфигурации eBGP между двумя устройствами.

Список команд для настройки первого узла:

```
set protocols bgp 65534 neighbor 192.168.0.2 ebgp-multihop '2'
set protocols bgp 65534 neighbor 192.168.0.2 remote-as '65535'
set protocols bgp 65534 neighbor 192.168.0.2 update-source
'192.168.0.1'
set protocols bgp 65534 address-family ipv4-unicast network
'172.16.0.0/16'
set protocols bgp 65534 parameters router-id '192.168.0.1'
set protocols static route 172.16.0.0/16 blackhole distance
'254'
```

Список команд для настройки второго узла:

```
set protocols bgp 65535 neighbor 192.168.0.1 ebgp-multihop '2'
set protocols bgp 65535 neighbor 192.168.0.1 remote-as '65534'
set protocols bgp 65535 neighbor 192.168.0.1 update-source
'192.168.0.2'
set protocols bgp 65535 address-family ipv4-unicast network
'172.17.0.0/16'
set protocols bgp 65535 parameters router-id '192.168.0.2'
```

```
set protocols static route 172.17.0.0/16 blackhole distance
'254'
```

По умолчанию префикс, объявленный в операторе network не обязательно должен существовать в таблице маршрутизации, однако для предотвращения возникновения петель маршрутизации его обычно добавляют.

Наиболее правильный способ сделать это - создать статический маршрут:

Для протокола IPv6

Пример простой конфигурации eBGP с использованием протокола IPv6.

Список команд для настройки первого узла:

```
set protocols bgp 65534 neighbor 2001:db8::2 ebgp-multihop '2'
set protocols bgp 65534 neighbor 2001:db8::2 remote-as '65535'
set protocols bgp 65534 neighbor 2001:db8::2 update-source
'2001:db8::1'
set protocols bgp 65534 neighbor 2001:db8::2 address-family
ipv6-unicast
set protocols bgp 65534 address-family ipv6-unicast network
'2001:db8:1::/48'
set protocols bgp 65534 parameters router-id '10.1.1.1'
set protocols static route6 2001:db8:1::/48 blackhole distance
'254'
```

Список команд для настройки второго узла:

```
set protocols bgp 65535 neighbor 2001:db8::1 ebgp-multihop '2'
set protocols bgp 65535 neighbor 2001:db8::1 remote-as '65534'
set protocols bgp 65535 neighbor 2001:db8::1 update-source
'2001:db8::2'
set protocols bgp 65535 neighbor 2001:db8::1 address-family
ipv6-unicast
set protocols bgp 65535 address-family ipv6-unicast network
'2001:db8:2::/48'
set protocols bgp 65535 parameters router-id '10.1.1.2'
set protocols static route6 2001:db8:2::/48 blackhole distance
'254'
```

По умолчанию префикс, объявленный в операторе network не обязательно должен существовать в таблице маршрутизации, однако для предотвращения возникновения петель маршрутизации его обычно добавляют. Наиболее правильный способ сделать это - создать статический маршрут.

Настройка фильтрации маршрутов

К маршрутам может быть применен фильтр с помощью карты маршрутов (route-map):

Список команд для настройки первого узла:

```
set policy prefix-list AS65535-IN rule 10 action 'permit'
set policy prefix-list AS65535-IN rule 10 prefix '172.16.0.0/16'
set policy prefix-list AS65535-OUT rule 10 action 'deny'
set policy prefix-list AS65535-OUT rule 10 prefix '172.16.0.0/16'
set policy prefix-list6 AS65535-IN rule 10 action 'permit'
set policy prefix-list6 AS65535-IN rule 10 prefix '2001:db8:2::/48'
set policy prefix-list6 AS65535-OUT rule 10 action 'deny'
set policy prefix-list6 AS65535-OUT rule 10 prefix '2001:db8:2::/48'
set policy route-map AS65535-IN rule 10 action 'permit'
set policy route-map AS65535-IN rule 10 match ip address prefix-list 'AS65535-IN'
set policy route-map AS65535-IN rule 10 match ipv6 address prefix-list 'AS65535-IN'
set policy route-map AS65535-IN rule 20 action 'deny'
set policy route-map AS65535-OUT rule 10 action 'deny'
set policy route-map AS65535-OUT rule 10 match ip address prefix-list 'AS65535-OUT'
set policy route-map AS65535-OUT rule 10 match ipv6 address prefix-list 'AS65535-OUT'
set policy route-map AS65535-OUT rule 20 action 'permit'
set protocols bgp 65534 neighbor 2001:db8::2 address-family ipv4-unicast route-map export 'AS65535-OUT'
set protocols bgp 65534 neighbor 2001:db8::2 address-family ipv4-unicast route-map import 'AS65535-IN'
```

```
set protocols bgp 65534 neighbor 2001:db8::2 address-family ipv6-unicast route-map export 'AS65535-OUT'
set protocols bgp 65534 neighbor 2001:db8::2 address-family ipv6-unicast route-map import 'AS65535-IN'
```

Список команд для настройки второго узла:

```
set policy prefix-list AS65534-IN rule 10 action 'permit'
set policy prefix-list AS65534-IN rule 10 prefix '172.17.0.0/16'
set policy prefix-list AS65534-OUT rule 10 action 'deny'
set policy prefix-list AS65534-OUT rule 10 prefix '172.17.0.0/16'
set policy prefix-list6 AS65534-IN rule 10 action 'permit'
set policy prefix-list6 AS65534-IN rule 10 prefix '2001:db8:1::/48'
set policy prefix-list6 AS65534-OUT rule 10 action 'deny'
set policy prefix-list6 AS65534-OUT rule 10 prefix '2001:db8:1::/48'
set policy route-map AS65534-IN rule 10 action 'permit'
set policy route-map AS65534-IN rule 10 match ip address prefix-list 'AS65534-IN'
set policy route-map AS65534-IN rule 10 match ipv6 address prefix-list 'AS65534-IN'
set policy route-map AS65534-IN rule 20 action 'deny'
set policy route-map AS65534-OUT rule 10 action 'deny'
set policy route-map AS65534-OUT rule 10 match ip address prefix-list 'AS65534-OUT'
set policy route-map AS65534-OUT rule 10 match ipv6 address prefix-list 'AS65534-OUT'
set policy route-map AS65534-OUT rule 20 action 'permit'
set protocols bgp 65535 neighbor 2001:db8::1 address-family ipv4-unicast route-map export 'AS65534-OUT'
set protocols bgp 65535 neighbor 2001:db8::1 address-family ipv4-unicast route-map import 'AS65534-IN'
set protocols bgp 65535 neighbor 2001:db8::1 address-family ipv6-unicast route-map export 'AS65534-OUT'
```

```
set protocols bgp 65535 neighbor 2001:db8::1 address-family
ipv6-unicast route-map import 'AS65534-IN'
```

Можно также расширить данный пример запретом локальных соединений и многоадресной рассылки в правиле с номеров 20 с действием action deny.

Настройка BGP соседства

Определение BGP соседей

```
set protocols bgp <asn> neighbor <address|interface> remote-as
<nasn>
```

Данная команда позволяет создать нового соседа, для которого значение параметра *remote-as* равно *<nasn>*. Адрес соседа может быть IPv4 адресом, IPv6 адресом или интерфейсом, который будет использоваться для соединения. Команда применима как для отдельного соседа, так и для группы соседей.

```
set protocols bgp <asn> neighbor <address|interface> remote-as
internal
```

При выполнении данной команды создание соседа происходит так же, как и при указании ASN, за исключением того, что если ASN соседа отличается от собственного, указанного в команде ***set protocols bgp <asn>***, то в соединении будет отказано.

```
set protocols bgp <asn> neighbor <address|interface> remote-as
external
```

При выполнении данной команды создание соседа происходит так же, как и при указании ASN, за исключением того, что если ASN соседа совпадает с собственным, указанным в команде ***set protocols bgp <asn>***, то в соединении будет отказано.

```
set protocols bgp <asn> neighbor <address|interface> shutdown
```

Данная команда позволяет отключить соседа или группу соседей. Для повторной активации соседа необходимо использовать контекст delete для данной команды.

```
set protocols bgp <asn> neighbor <address|interface>
description <text>
```

Данная команда позволяет задать описание для соседа или группы соседей.

```
set protocols bgp <asn> neighbor <address|interface> update-
source <address|interface>
```

Данная команда позволяет указать адрес отправителя, который будет использоваться для BGP сессии с этим соседом. Адрес может быть указан как IPv4 адрес или как имя интерфейса.

Согласование возможностей BGP соседства

- `set protocols bgp <asn> neighbor <address|interface> capability dynamic`

Данная команда позволяет динамически обновлять список доступных функций через установленную BGP сессию.

- `set protocols bgp <asn> neighbor <address|interface> capability extended-nexthop`

Данная команда позволяет разрешить для протокола BGP согласование функции extended-nexthop со своим соседом. Если подключение к соседу осуществляется по протоколу IPv6 с использованием локального адреса Link-Local, то эта возможность включается автоматически. Если подключение к соседу осуществляется по глобальному адресу IPv6, то включение этой команды позволит протоколу BGP устанавливать маршруты IPv4 с указанием адресов следующих узлов по протоколу IPv6, но только в случае, если на интерфейсах не настроен IPv4 адрес.

- `set protocols bgp <asn> neighbor <address|interface> disable-capability-negotiation`

Данная команда запрещает отправку соседу сообщения Capability Negotiation в качестве необязательного параметра сообщения для установления соседства. Команда действует только в случае, если для соседа задана конфигурация, отличная от одноадресной конфигурации IPv4.

Если удаленный сосед не имеет возможности согласования функций, то он не будет посылать список таковых. В этом случае протокол BGP выполнит настройку соседа в соответствии с заданными функциями.

В некоторых случаях может потребоваться использования большего числа функций, чем было согласовано между соседями, даже если удаленный сосед передает определенный список функций. В этом случае, если сосед сконфигурирован с помощью параметра override-capability, то операционная система Факел игнорирует полученный список функций, а затем заменяет согласованные функции сконфигурированными.

Также необходимо иметь в виду, что данная функция принципиально лишает возможности использовать широко распространенные функции протокола BGP, такие как: использование ненумерованных интерфейсов, поддержка имен хостов, использование четырехбайтных номеров AC, задание нескольких путей для одного и того же префикса, обновление маршрутов, фильтрация маршрутной информации, динамический список функций, плавный перезапуск.

- `set protocols bgp <asn> neighbor <address|interface> override-capability`

Данная команда позволяет переопределить результат согласования функций с помощью локальной конфигурации. В этом случае список функций удаленного соседа игнорируется.

- `set protocols bgp <asn> neighbor <address|interface> strict-capability-match`

Данная команда принуждает к строгому сравнению списка доступных функций удаленного и локального соседей. Если списка функций отличается, формируется ошибка Unsupported Capability, а затем происходит сброс соединения.

Можно также запретить отправку параметра Capability Negotiation в сообщении для установления соседства, если удаленный сосед не поддерживается возможности согласования функций. Для отключения необходимо использовать команду disable-capability-negotiation.

Настройка параметров для BGP соседей

- `set protocols bgp <asn> neighbor <address|interface> address-family <ipv4-unicast|ipv6-unicast> allowas-in number <number>`

Данная команда принимает входящие маршруты с путем до АС, содержащим номер АС с тем же значением и АС данной системы. Такой подход используется в случаях, когда необходимо использовать один и тот же номер АС для разных сегментов, соединить которые напрямую не представляется возможным.

Параметр *<number>* позволяет настроить количество допустимых повторений системного номера АС в пути до АС.

Допустимо использование данной команды только для отдельных eBGP соседей и не допустимо для групп соседей.

- `set protocols bgp <asn> neighbor <address|interface> address-family <ipv4-unicast|ipv6-unicast> as-override`

Данная команда переопределяет номер АС маршрутизатора-источника локальным номером АС.

Обычно такая конфигурация применяется на оконечном оборудовании провайдера (Provider Edge) для замены входящего номера АС клиента таким образом, чтобы оконечное клиентское оборудование (Customer Edge) могло использовать тот же номер АС. Это позволяет клиенту сети провайдера использовать один и тот же номер АС во всех сегментах.

Допустимо использование данной команды только для отдельных eBGP соседей.

- `set protocols bgp <asn> neighbor <address|interface> address-family <ipv4-unicast|ipv6-unicast> attribute-unchanged <as-path|med|next-hop>`

Данная команда задает список атрибутов, которые должны быть неизменными при анонсах соседу или группе соседей.

- `set protocols bgp <asn> neighbor <address|interface> address-family <ipv4-unicast|ipv6-unicast> maximum-prefix <number>`

Данная команда задает максимальное количество префиксов, которое можно получить от определенного соседа. Если это количество будет превышено, то BGP сессия будет сброшена. Диапазон значений составляет от 1 до 4294967295.

- `set protocols bgp <asn> neighbor <address|interface> address-family <ipv4-unicast|ipv6-unicast> nexthop-self`

Данная команда принуждает BGP соседа, являющегося источником анонсов, сообщать о себе как о следующем узле для анонсируемого в адрес своего соседа маршрута.

- `set protocols bgp <asn> neighbor <address|interface> address-family <ipv4-unicast|ipv6-unicast> remove-private-as`

Данная команда удаляет частные номера ASN для маршрутов, которые анонсируются сконфигурированному соседу. Ее использование приводит к удалению только частных номеров ASN для маршрутов, анонсируемых eBGP соседям.

Если параметр AS-Path содержит только частные номера ASN для маршрута, то все эти номера удаляются.

Если параметр AS-Path содержит частные номера ASN наряду с публичными номерами ASN, то считается, что так задумано, и частные номера ASN не удаляются.

- `set protocols bgp <asn> neighbor <address|interface> address-family <ipv4-unicast|ipv6-unicast> soft-reconfiguration inbound`

Внесение изменений в политики BGP требуют сброса соответствующих сессий. Сброс оказывает негативное влияние на работу сети. Плавная внесение изменений в конфигурацию позволяет генерировать входящие обновления от соседа, изменять и активировать политики BGP без сброса сессий.

Данная команда определяет, что обновления маршрутов, полученные от определенного соседа, будут сохраняться без изменений вне зависимости от политики входящих соединений. Если включено плавное изменение конфигурации входящих маршрутов, то сохраненные обновления обрабатываются новой конфигурацией политики для создания новых входящих обновлений.

Примечание

> *Для хранения обновлений маршрутов используется оперативная память. Если включить плавное внесение изменений в конфигурацию входящих маршрутов для нескольких соседей, объем используемой памяти может значительно возрасти.*

- `set protocols bgp <asn> neighbor <address|interface> address-family <ipv4-unicast|ipv6-unicast> weight <number>`

Данная команда задает значение веса по умолчанию для маршрутов соседа. Диапазон значений составляет от 1 до 65535.

- `set protocols bgp <asn> neighbor <address|interface> advertisement-interval <seconds>`

Данная команда задает минимальный интервал между анонсами маршрутов для соседа. Значение интервала составляет от 0 до 600 секунд, по умолчанию интервал между анонсами равен 0.

- `set protocols bgp <asn> neighbor <address|interface> disable-connected-check`

Данная команда позволяет соединять между собой напрямую подключенных eBGP соседей с использованием Loopback адресов и без корректировка времени жизни (TTL), по умолчанию значение которого равно 1.

- `set protocols bgp <asn> neighbor <address|interface> disable-send-community <extended|standard>`

Данная команда определяет, что атрибут Community не должен передаваться в обновлениях маршрутов, адресованных соседу. По умолчанию атрибут Community передается.

- `set protocols bgp <asn> neighbor <address|interface> ebgp-multihop <number>`

Данная команда позволяет устанавливать соединения с eBGP соседями, даже если они находятся на удалении в несколько пересылок между узлами сети. Если сосед не имеет прямого соединения и данная команда не используется, соединение не будет установлено. Диапазон количества пересылок между узлами сети составляет от 1 до 255. Эта команда является взаимоисключающей по отношению к команде ttl-security hops.

- `set protocols bgp <asn> neighbor <address|interface> local-as <asn> [no-prepend] [replace-as]`

Данная команда позволяет указать альтернативную AC для данного BGP процесса при взаимодействии с указанным соседом или группой соседей. Без дополнительных атрибутов AC, указанная в параметре local-as, добавляется к входящему значению AS-

Path при получении обновлений маршрутов от соседа и к исходящему значению AS-Path (после обработки локальной АС) при передаче локальных маршрутов соседу.

Если указан атрибут no-prepend, то заданное значение параметра local-as не добавляется к входящему значению AS-Path.

Если указан атрибут replace-as, то при передаче обновлений локальных маршрутов соседу к значению AS-Path добавляется только заданное значение параметра local-as.

> **Примечание**
>
> *Данная команда применима только для eBGP соседей.*

- `set protocols bgp <asn> neighbor <address|interface> passive`

Данная команда настраивает узел в качестве источника BGP анонсов таким образом, чтобы он принимал только входящие соединения, но не инициировал исходящие соединения с соседом или группой соседей.

- `set protocols bgp <asn> neighbor <address|interface> password <text>`

Данная команда задает пароль для TCP сокета, кодированный алгоритмом MD5 и используемый для соединения с удаленным соседом.

- `set protocols bgp <asn> neighbor <address|interface> ttl-security hops <number>`

Данная команда реализует механизм безопасности GTSM согласно спецификации RFC 5082. С помощью этой команды статус соседей разрешается присваивать только узлам, находящимся на расстоянии заданного числа пересылок между узлами. Диапазон числа пересылок между узлами составляет от 1 до 254. Команда является взаимоисключающей по отношению к команде ebgp-multihop.

Настройка группы BGP соседей

Группы соседей используются для улучшения масштабирования за счет объединения общих параметров и информации об обновлениях для всех участников группы. Это означает, что маршруты, созданные участником группы, будут отправляться обратно в адрес источника с атрибутом Originator Identifier, значение которого указывает на соседа-источника маршрутной информации. Соседи, не связанные ни с одной группой, воспринимаются как принадлежащие к группе соседей по умолчанию, и будут использовать общие обновления.

- `set protocols bgp <asn> peer-group <name>`

Данная команда определяет новую группу соседей. Для группы можно указать те же параметры, что я для отдельных соседей.

Примечание

Если изменения применяются к IP-адресу отдельного соседа, они переопределяют изменения, внесенные для группы соседей, в которую входит данный IP-адрес.

```
set protocols bgp <asn> neighbor <address|interface> peer-group
<name>
```

Данная команда связывает конкретную группу соседей с заданным именем.

Настройка анонсов BGP

```
set protocols bgp <asn> address-family <ipv4-unicast|ipv6-
unicast> network <prefix>
```

Данная команда используется для анонсирования сетей IPv4 или IPv6.

Примечание

По умолчанию префикс BGP анонсируется, даже если он отсутствует в таблице маршрутизации.

```
set protocols bgp <asn> parameters network-import-check
```

Данная конфигурация позволяет изменить поведение оператора network. При такой настройке базовая сеть должна существовать в таблице маршрутизации.

```
set protocols bgp <asn> neighbor <address|interface> address-
family <ipv4-unicast|ipv6-unicast> default-originate [route-map
<name>]
```

По умолчанию в ПО **Факел** не анонсируется маршрут по умолчанию (0.0.0.0/0), даже если он есть в таблице маршрутизации. Данную команду следует использовать в случае, если необходимо анонсировать маршрут по умолчанию. При использовании дополнительного аргумента route-map, можно передать маршрут по умолчанию заданному соседу только в том случае, если выполняются условия заданной аргументом route-map карты маршрутов.

Настройка агрегации маршрутов

```
set protocols bgp <asn> address-family <ipv4-unicast|ipv6-
unicast> aggregate-address <prefix>
```

Данная команда задает агрегированный адрес. Маршрутизатор также будет анонсировать более длинные префиксы внутри агрегированного адреса.

- `set protocols bgp <asn> address-family <ipv4-unicast|ipv6-unicast> aggregate-address <prefix> as-set`

Данная команда задает агрегированный адрес с математическим набором AC. Команда суммирует атрибуты AS-Path для всех отдельных маршрутов.

- `set protocols bgp <asn> address-family <ipv4-unicast|ipv6-unicast> aggregate-address <prefix> summary-only`

Данная команда задает агрегированный адрес и обеспечивает подавление более длинных префиксов внутри агрегированного адреса перед отправкой обновлений BGP соседям.

- `set protocols bgp <asn> neighbor <address|interface> address-family <ipv4-unicast|ipv6-unicast> unsuppress-map <name>`

Данная команда применяет определенную карту маршрутов для выборочного исключения префиксов, подавленных в процессе суммирования.

Настройка редистрибуции BGP

- `set protocols bgp <asn> address-family <ipv4-unicast|ipv6-unicast> redistribute <route source>`

Данная команда осуществляет редистрибуцию маршрутной информации от заданного источника в процесс BGP. Для источника маршрута доступны следующие режимы: connected, kernel, ospf, rip, static, table.

- `set protocols bgp <asn> address-family <ipv4-unicast|ipv6-unicast> redistribute <route source> metric <number>`

Данная команда задает метрику MED для маршрутов, подлежащих редистрибуции. Диапазон значений метрики составляет от 0 до 4294967295. Для источника маршрутной информации доступны следующие режимы: connected, kernel, ospf, rip, static, table.

- `set protocols bgp <asn> address-family <ipv4-unicast|ipv6-unicast> redistribute <route source> route-map <name>`

Данная команда позволяет использовать определенную карту маршрутов для фильтрации маршрутов, подлежащих редистрибуции. Для источника маршрутной информации доступны следующие режимы: connected, kernel, ospf, rip, static, table.

Базовые настройки BGP

Общие параметры BGP

- `set protocols bgp <asn> parameters router-id <id>`

Данная команда задает идентификатор маршрутизатора router-ID. Если идентификатор не указан, то в качестве него будет использоваться IP-адрес интерфейса с наибольшим значением.

```
set protocols bgp <asn> address-family <ipv4-unicast|ipv6-unicast> maximum-paths <ebgp|ibgp> <number>
```

Данная команда определяет максимальное количество параллельных маршрутов, которое может поддерживаться процессом BGP. Для того, чтобы процесс BGP использовал второй путь, необходимо, чтобы совпадали следующие атрибуты: вес, локальные предпочтения, путь до AC (номер AC и длина пути), код источника маршрутной информации, атрибут MED, метрика IGP. Кроме того, адрес следующего узла для каждого пути должен быть разным.

```
set protocols bgp <asn> parameters default no-ipv4-unicast
```

Данная команда позволяет отключить по умолчанию установление соседства по протоколу IPv4.

```
set protocols bgp <asn> parameters log-neighbor-changes
```

Данная команда позволяет активирует запись в журнал события, которые связаны изменения статуса соседей (доступен / не доступен) и причины сброса.

```
set protocols bgp <asn> parameters no-client-to-client-reflection
```

Данная команда отключает отражение маршрутов между клиентами. По умолчанию клиенты не должны быть объединены в полносвязную сеть, и маршруты от определенного клиента отражаются всем остальным. Если клиенты все же объединены в полносвязную сеть, отражение маршрутов не требуется. В этом случае необходимо использовать команду no-client-to-client-reflection, чтобы отключить отражение маршрутов между клиентами.

```
set protocols bgp <asn> parameters no-fast-external-failover
```

Данная команда отключает немедленный сброс сессий при отказе канала связи с соседом.

Настройка административной дистанции для протокола BGP

```
set protocols bgp <asn> parameters distance global <external|internal|local> <distance>
```

Данная команда изменяет значение административной дистанции в BGP. В качестве аргументов указываются значения административной дистанции для внешних, внутренних и локальных маршрутов соответственно. Диапазон значений административной дистанции составляет от 1 до 255.

- `set protocols bgp <asn> parameters distance prefix <subnet> distance <distance>`

Данная команда задает административную дистанцию для определенного маршрута. Диапазон значений административной дистанции составляет от 1 до 255.

> **Примечание**
>
> *Маршруты с административной дистанцией 255 фактически отключены и не используются модулями ядра при принятии решений о маршрутизации трафика.*

Настройка таймеров

- `set protocols bgp <asn> timers holdtime <seconds>`

Данная команда задает время удержания маршрутов в секундах. Диапазон значений такого таймера составляет от 4 до 65535. По умолчанию используется значение 180 секунд. Если установить значение 0, то ПО **Факел** не будет удерживать маршруты.

- `set protocols bgp <asn> timers keepalive <seconds>`

Данная команда задает время жизни маршрутов в секундах. Диапазон значений такого таймера составляет от 4 до 65535. По умолчанию используется значение 60 секунд.

- `set protocols bgp <asn> timers keepalive <seconds>`

Настройка BGP Route Dampening

Когда маршрут выходит из строя, выполняется обновление маршрутизации для исключения данного маршрута из таблиц маршрутизации сети. При восстановлении маршрута изменение его доступности также объявляется. Маршрут, который неоднократно выходит из строя и восстанавливается, вызывает значительный сетевой трафик для обновления статуса маршрута в сети.

Технология Route Dampening (подавление маршрута), описанная в RFC 2439, позволяет выявлять маршруты, которые повторно выходят из строя и восстанавливаются. При активации Route Dampening нестабильный маршрут накапливает штрафные очки каждый раз, когда происходит его отказ и последующее восстановление. Если суммарные штрафные очки превышают установленный порог, маршрут перестает анонсироваться. Это и есть подавление маршрута. Маршруты, которые были подавлены, могут быть вновь добавлены в таблицу маршрутизации только после того, как величина их штрафных очков снизится ниже установленного порога.

За каждый сбой маршрута начисляется штраф в 1000 очков. Когда сумма штрафов достигает заранее определенного порога (пороговое значение для подавления), маршрутизатор прекращает анонсировать данный маршрут.

После начисления штрафа маршруту, величина штрафа уменьшается в два раза по прошествии определенного времени. Когда суммарные штрафы уменьшаются до

заранее установленного порога (пороговое значение для повторного использования), маршрут перестает быть подавленным и возвращается в таблицу маршрутизации BGP.

Подавление маршрута не является бессрочным. Максимальное время подавления устанавливает предельное время, в течение которого маршрут может быть подавлен, прежде чем он будет снова анонсирован.

- `set protocols bgp <asn> parameters dampening half-life <minutes>`

Данная команда определяет время в минутах, по истечении которого примененный штраф уменьшается вдвое. Диапазон значений составляет от 10 до 45 минут.

- `set protocols bgp <asn> parameters dampening re-use <seconds>`

Данная команда определяет накопленную сумму штрафа, при которой маршрут повторно анонсируется. Диапазон значений составляет от 1 до 20000.

- `set protocols bgp <asn> parameters dampening start-suppress-time <seconds>`

Данная команда определяет накопленную сумму штрафов, при которой маршрут подавляется. Диапазон значений составляет от 1 до 20000.

- `set protocols bgp <asn> parameters dampening max-suppress-time <seconds>`

Данная команда определяет максимальное время в минутах, в течение которого маршрут может оставаться подавленным, прежде чем произойдет его повторное анонсирование. Диапазон значений составляет от 1 до 255 минут.

Настройка выбора маршрутов

- `set protocols bgp <asn> parameters always-compare-med`

Данная команда позволяет сравнивать значения атрибута MED маршрутов, даже если они были полученных из разных соседних АС. Использование данной опции делает порядок предпочтения маршрутов более определенным и должно исключить возможные колебания, вызванные MED.

- `set protocols bgp <asn> parameters bestpath as-path confed`

Данная команда определяет, что длина наборов и последовательностей разделенных путей (confederation path) должна учитываться в процессе принятия решения протоколом BGP о выборе оптимального пути.

- `set protocols bgp <asn> parameters bestpath as-path multipath-relax`

Данная команда предписывает в процессе принятия протоколом BGP решений о выборе оптимального пути рассмотрение путей одинаковой длины AS-Path в качестве кандидатов на расчет множественности путей. Без использования этой команды при расчете соответствующих этой концепции маршрутов требуется совпадение всего значения AS-Path.

- `set protocols bgp <asn> parameters bestpath as-path ignore`

Данная команда предписывает игнорирование длины AS-Path при выборе маршрута.

- `set protocols bgp <asn> parameters bestpath compare-routerid`

При сравнении маршрутов необходимо убедиться, что если они равны по большинству метрик, включая локальные предпочтения, длину пути до AC, стоимость IGP, значение атрибута MED, равенство в конечном счете определяется по идентификатору маршрутизатора router-ID.

Если данная опция включена, то проверка уже выбранных маршрутов, при которой предпочтение отдается уже выбранным маршрутам eBGP, пропускается.

Если маршрут имеет атрибут ORIGINATOR_ID в результате отражения, то будет использоваться значение атрибута ORIGINATOR_ID. В противном случае будет использоваться идентификатор маршрутизатора того соседа, от которого был получен маршрут.

Преимуществом такого подхода является то, что выбор маршрута на данном этапе будет более детерминированным. Недостатком является то, что из-за такой проверки один или несколько маршрутизаторов или один маршрутизатор с наименьшим идентификатором могут перетянуть весь трафик на другие равные пути. Это может увеличить вероятность возникновения колебаний MED или IGP, если не были приняты другие меры для их предотвращения. Точное поведение будет зависеть от топологии iBGP и процесса отражения маршрутов.

- `set protocols bgp <asn> parameters bestpath med confed`

Данная команда определяет, что протокол BGP учитывает атрибут MED при сравнении маршрутов, источниками которых являются разные дочерние AC в пределах области разделения, к которой принадлежит данный источник анонсов по протоколу BGP. Состоянием по умолчанию является состояние, при котором атрибут MED не учитывается.

- `set protocols bgp <asn> parameters bestpath med missing-as-worst`

Данная команда определяет, что маршрут с атрибутом MED всегда считается маршрутом с самым высоким приоритетом, чем без него, определяя для отсутствующего атрибута MED фактически значение бесконечности. Состоянием по умолчанию является состояние, при котором атрибут MED не учитывается.

- `set protocols bgp <asn> parameters default local-pref <local-pref value>`

Данная команда задает значение локального предпочтения по умолчанию. Диапазон значений локального предпочтения составляет от 0 до 4294967295.

- `set protocols bgp <asn> parameters deterministic-med`

Данная команда позволяет сравнивать различные значения атрибута MED, анонсируемые соседями в одной АС, при выборе маршрутов. Когда эта команда включена, маршруты из одной АС группируются вместе, сравниваются лучшие записи из каждой группы.

- `set protocols bgp <asn> address-family ipv4-unicast network <prefix> backdoor`

Данная команда предписывает маршрутизатору предпочтение маршрут с указанным префиксом, полученным путем использования протокола IGP по через backdoor соединение, вместо маршрута к тому же префиксу, полученному путем использования протокола eBGP.

Настройка фильтрации маршрутов

Для контроля и модификации маршрутной информации, которой обмениваются соседи, можно использовать карту маршрутов, список заданных фильтров и префиксов, список дистрибуции.

Для входящих обновлений предпочтение отдается следующим образом:

- карта маршрутов (route-map);
- список заданных фильтров (filter-list);
- список заданных префиксов, список дистрибуции (prefix-list, distribute-list).

Для исходящих обновлений предпочтение отдается следующим образом:

- список заданных префиксов, список дистрибуции (prefix-list, distribute-list);
- список заданных фильтров (filter-list);
- карта маршрутов (route-map).

Примечание

Атрибуты prefix-list и distribute-list являются взаимоисключающими, и только одна команда (distribute-list или prefix-list) может быть применена к каждому входящему или исходящему направлению для конкретного соседа.

- `set protocols bgp <asn> neighbor <address|interface> address-family <ipv4-unicast|ipv6-unicast> distribute-list <export|import> <number>`

Данная команда применяет фильтры списка доступа с именем *<number>* к указанному соседу BGP для ограничения информации о маршрутизации, которую протокол BGP получает и/или анонсирует. Аргументы *export* и *import* задают направление применения списка доступа.

- `set protocols bgp <asn> neighbor <address|interface> address-family <ipv4-unicast|ipv6-unicast> prefix-list <export|import> <name>`

Данная команда применяет фильтры из списка *prfefix-list*, заданные в *<name>*, к указанному соседу BGP для ограничения маршрутной информации, которую протокол BGP получает и/или анонсирует. Аргументы *export* и *import* задают направление применения списка префиксов.

- `set protocols bgp <asn> neighbor <address|interface> address-family <ipv4-unicast|ipv6-unicast> route-map <export|import> <name>`

Данная команда применяет карту маршрутов с именем *<name>* к указанному соседу BGP для управления и модификации маршрутной информации, которой обмениваются между собой соседи. Аргументы export и import задают направление применения карты маршрутов.

- `set protocols bgp <asn> neighbor <address|interface> address-family <ipv4-unicast|ipv6-unicast> filter-list <export|import> <name>`

Данная команда применяет к определенному соседу BGP фильтры списка доступа к значениям AS-Path, указанным в *<name>*, для ограничения информации о маршрутизации, которую протокол BGP получает и/или анонсирует. Аргументы *export* и *import* задают направление, в котором применяется список доступа к AS-Path.

- `set protocols bgp <asn> neighbor <address|interface> address-family <ipv4-unicast|ipv6-unicast> capability orf <receive|send>`

Данная команда включает возможность фильтрации маршрутной информации ORF, описание которой представлено в спецификации *RFC 5291*, на локальном маршрутизаторе и включает анонсирование возможности ORF для указанного BGP соседа. Ключ *receive* настраивает маршрутизатор на анонсирование возможности приема ORF. Ключ *send* настраивает маршрутизатор на анонсирование возможности передачи ORF. Чтобы анонсировать фильтр отправителя, необходимо создать список IP префиксов для указанного BGP соседа, применяемый при входящей сортировке.

Настройка масштабирования BGP

Маршрутизаторы BGP, подключенные внутри того же автономной системы (AS) посредством BGP, принадлежат к внутренней сессии BGP, или IBGP. Для предотвращения петель таблицы маршрутизации узел IBGP не анонсирует маршруты, изученные через IBGP, другому узлу IBGP (механизм Split Horizon). Таким образом, для IBGP требуется полная меш-сеть всех узлов. Для больших сетей это быстро становится не масштабируемым.

Есть два способа, которые помогают нам смягчить требование полной меш-сети BGP в сети:

- Использование BGP Route reflectors (рефлекторов маршрутов BGP);
- Использование BGP confederation (конфедерации BGP).

Настройка BGP Route Reflectors

Внедрение узлов, выполняющих функцию отражения маршрутов, устраняет необходимость в организации полносвязной сети соседей. При настройке таких узлов необходимо указать маршрутизатору, является ли другой маршрутизатора iBGP клиентом или нет. Клиент представляет собой iBGP маршрутизатор на который узел будет отражать маршруты. Маршрутизатор не являющийся клиентом - обычный iBGP сосед. Описание механизма отражения маршрутов представлено в спецификации RFC 4456 и дополнено в спецификации RFC 7606.

```
set protocols bgp <asn> neighbor <address> address-family
<ipv4-unicast|ipv6-unicast> route-reflector-client
```

Данная команда определяет указанного соседа как клиента для отражения маршрутов.

```
set protocols bgp <asn> parameters cluster-id <id>
```

Данная команда задает идентификатор кластера, который формирует совокупность узлов, отражающих маршруты, и их клиентов, и используется узлами, отражающими маршруты, для предотвращения петель в коммуникациях. По умолчанию идентификатор кластера равен значению параметра router-id для протокола BGP, но может быть установлен в произвольное 32-битное значение.

Настройка BGP confederation

Механизм разделения в протоколе BGP подразумевает деление АС на подсистемы для уменьшения количества необходимых iBGP соединений. В пределах одной подсистемы по-прежнему требуется наличие полносвязной iBGP сети, но между подсистемами используется подобие eBGP, функционирующее как iBGP (таким образом определяется область разделения). Описание механизма представлено в спецификации RFC 5065.

```
set protocols bgp <subasn> parameters confederation identifier
<asn>
```

Данная команда задает идентификатор области разделения в протоколе BGP. Параметр <*asn*> - номер АС, которая включает в себя несколько подсистем (область разделения). Параметр <*subasn*> - номер подсистемы в рамках АС, заданной параметром <*asn*>.

```
set protocols bgp <subasn> parameters confederation
confederation peers <nsubasn>
```

Данная команда устанавливает другие области разделения <*nsubasn*> в качестве подчиненных АС, идентификатор которой указан в команде confederation identifier <*asn*>.

Мониторинг и эксплуатация BGP

```
show <ip|ipv6> bgp
```

Выводит на экран все записи в таблице маршрутизации BGP.

```
show <ip|ipv6> bgp <address|prefix>
```

Выводит на экран информацию о конкретной записи в таблице маршрутизации BGP.

```
show ip bgp cidr-only
```

Выводит на экран список маршрутов с бесклассовой междоменной маршрутизацией (CIDR).

```
show <ip|ipv6> bgp community <value>
```

Выводит на экран список маршрутов, принадлежащие указанным сообществам BGP. Допустимые значения: номер сообщества в диапазоне от 1 до 4294967200, AA:NN (AA - 2-байтовый номер АС, NN - 2-байтовый номер сообщества), no-export, local-as или no-advertise.

```
show <ip|ipv6> bgp community-list <name>
```

Выводит на экран список маршрутов, разрешенные списком сообществ BGP.

```
show ip bgp dampened-paths
```

Выводит на экран список dampened (подавленных) маршрутов BGP.

```
show ip bgp flap-statistics
```

Выводит на экран информацию о нестабильных (flapping) маршрутах BGP.

```
show ip bgp filter-list <name>
```

Выводит на экран список маршрутов BGP, разрешенные указанным списком доступа AS-Path.

- `show <ip|ipv6> bgp neighbors <address> advertised-routes`

Выводит на экран список маршрутов BGP, анонсируемых соседу <*address*>.

- `show <ip|ipv6> bgp neighbors <address> received-routes`

Выводит на экран список маршрутов BGP, полученных от указанного сосед <*address*>, до применения политики входящих соединений.

- `show <ip|ipv6> bgp neighbors <address> routes`

Выводит на экран список маршрутов BGP, которые были получены от указанного соседа <*address*> после применения фильтрации.

- `show <ip|ipv6> bgp neighbors <address> dampened-routes`

Выводит на экран dampened маршруты, полученные от BGP соседа <*address*>.

- `show <ip|ipv6> bgp regexp <text>`

Выводит на экран список маршрутов BGP, для которых значение AS-Path соответствует заданному регулярному выражению <*text*>.

- `show <ip|ipv6> bgp summary`

Выводит на экран состояние всех BGP соединений.

- `reset <ip|ipv6> bgp <address> [soft [in|out]]`

Сбрасывает BGP соединения с указанным соседом <*address*>. С аргументом *soft* эта команда инициирует плавный сброс. Если не используются опции *in* или *out*, то запускается как входящее, так и исходящее плавное изменение в настройках конфигурации маршрутов.

- `reset ip bgp all`

Сбрасывает все BGP соединения на маршрутизаторе.

- `reset ip bgp dampening`

Сбрасывает информацию о dampened соединения на маршрутизаторе.

- `reset ip bgp external`

Сбрасывает все внешние BGP соседства для маршрутизатора.

```
reset ip bgp peer-group <name> [soft [in|out]]
```

Сбрасывает BGP соединения с указанной группой соседей. С аргументом soft эта команда инициирует плавный сброс. Если не используются опции *in* или *out*, то запускается как входящее, так и исходящее плавное изменение в настройках конфигурации маршрутов.

IS-IS - Intermediate System to Intermediate System

Общая информация о протоколе IS-IS

Протокол IS-IS является протоколом с контролем состояния соединения из категории IGP протоколов для маршрутизации внутри локальной сети. Данный протокол описан в стандарте ISO/IEC 10589:2002 и спецификациях RFC 1195 и RFC 5308. Как и протокол OSPF, протокол IS-IS использует алгоритм Дейкстры для сбора данных о топологии сети и определения по этим данным кратчайшего пути до пункта назначения. Маршрутизаторы обмениваются друг с другом информацией о соседствующих узлах. Протокол IS-IS функционирует на канальном (L2) уровне. Адреса, используемые протоколом IS-IS, называются NETs, и могут быть длиной от 8 до 20 байт, но обычно - 10 байт.

Алгоритм настройки IS-IS

1) Перейти в режим Конфигурации устройства:

```
configure
```

2) Указать NET адрес маршрутизатора:

```
set protocols isis net <network-entity-title>
```

3) Активировать протокол IS-IS на сетевом интерфейсе:

```
set protocols isis interface <interface>
```

4) Установить уровень работы маршрутизатора:

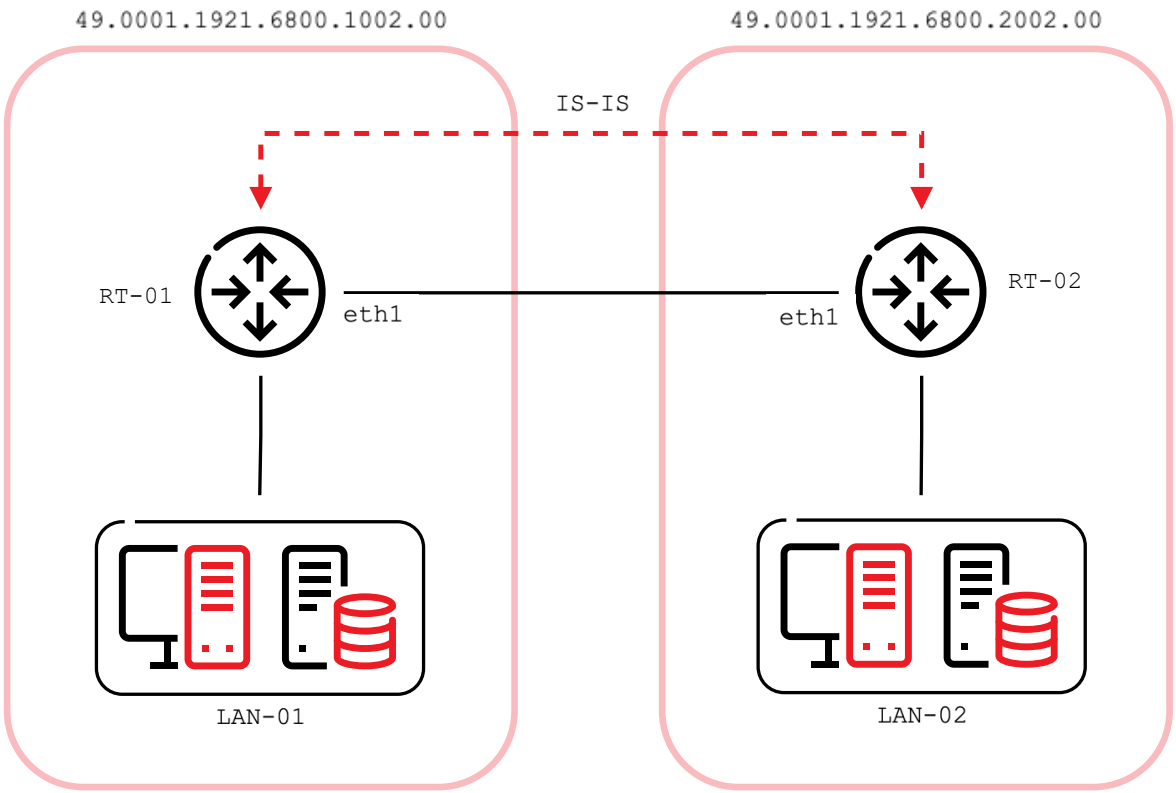
```
set protocols isis level <level-1|level-1-2|level-2>
```

5) Установить уровень работы сетевого интерфейса:

```
set protocols isis interface <interface> circuit-type <level-
1|level-1-2|level-2-only>
```

Пример настройки

Пример простой конфигурации протокола IS-IS с редистрибуцией маршрутов до непосредственно подключенных (connected) к интерфейсам подсетей.



Список команд для настройки первого узла:

```
set interfaces dummy dum0 address '203.0.113.1/24'
set interfaces ethernet eth1 address '192.0.2.1/24'
set policy prefix-list EXPORT-ISIS rule 10 action 'permit'
set policy prefix-list EXPORT-ISIS rule 10 prefix
'203.0.113.0/24'
set policy route-map EXPORT-ISIS rule 10 action 'permit'
set policy route-map EXPORT-ISIS rule 10 match ip address
prefix-list 'EXPORT-ISIS'
set protocols isis interface eth1
set protocols isis net '49.0001.1921.6800.1002.00'
set protocols isis redistribute ipv4 connected level-2 route-
map 'EXPORT-ISIS'
```

Список команд для настройки второго узла:

- set interfaces ethernet eth1 address '192.0.2.2/24'
- set protocols isis interface eth1
- set protocols isis net '49.0001.1921.6800.2002.00'

После конфигурирования необходимо убедиться в регистрации маршрутов на Узле 2.

```
fakel@F-2:~$ show ip route isis
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, D - SHARP,
       F - PBR, f - OpenFabric,
       > - selected route, * - FIB route, q - queued route, r -
rejected route

203.0.113.0/24 [115/10] via 192.0.2.1, eth1, 00:03:42
```

Основные настройки IS-IS

- set protocols isis net <network-entity-title>

Задает значение идентификатор NET в стандартном (ISO) формате.

- set protocols isis interface <interface>

Активирует протокола IS-IS на выбранном интерфейсе. Для экземпляра IS-IS необходимо использовать то же имя, что и при настройке процесса IS-IS.

- set protocols isis dynamic-hostname

Активирует механизм DHEM динамического сопоставления имени хоста. Описание механизма DHEM представлено в спецификации RFC 2763.

- set protocols isis level <level-1|level-1-2|level-2>

Устанавливает уровень, на котором будет работать маршрутизатор.

- set protocols isis lsp-mtu <size>

Задает максимальный размер структуры LSP в байтах в диапазоне 128 - 4352.

```
set protocols isis metric-style <narrow|transition|wide>
```

Задает тип метрики, который будет использоваться в работе протокола IS-IS.

- **narrow** - принимает и генерирует TLV старого типа;
- **transition** - принимает и генерирует TLV нового и старого типов;
- **wide** - принимает и генерирует TLV нового типа.

```
set protocols isis purge-originator
```

Активирует идентификацию инициатора очистки POI согласно спецификации RFC 6232 посредством добавления TLV с указанной промежуточной системой IS для структур LSP, которые не содержат информацию о POI. Если система IS является инициатором очистки, операционная система Факел добавляет TLV с идентификатором данной система в очередь для очистки.

```
set protocols isis set-attached-bit
```

Задает значение для бита ATT в структуре LSP для режима маршрутизатора станции согласно спецификации RFC 3787.

```
set protocols isis set-overload-bit
```

Задает значение для бита перегрузки, чтобы исключить прохождение через маршрутизатор транзитного трафика, согласно RFC 3787.

```
set protocols isis name default-information originate
<ipv4|ipv6> level-1
```

Генерирует маршрут по умолчанию для маршрутизатора, который работает на уровне level-1.

```
set protocols isis name default-information originate
<ipv4|ipv6> level-2
```

Генерирует маршрут по умолчанию для маршрутизатора, который работает на уровне level-2.

```
set protocols isis name default-information originate
<ipv4|ipv6> level-2
```

Настройка интерфейсов для протокола IS-IS

```
set protocols isis interface <interface> circuit-type <level-
1|level-1-2|level-2>
```

Устанавливает уровень, на котором работает сетевой интерфейс

- `set protocols isis interface <interface> hello-interval <seconds>`

Задает для сетевого интерфейса временной интервал для отправки hello сообщений (1 - 600).

- `set protocols isis interface <interface> hello-multiplier <seconds>`

Устанавливает множитель для периода удержания hello сообщений на выбранном интерфейса (2 - 100).

- `set protocols isis interface <interface> hello-padding`

Настраивает смещение в hello пакетах для учета асимметричных MTU от различных хостов согласно спецификации RFC 3719. Использование команды позволяет избежать преждевременного установления соседства, когда MTU на одном из маршрутизаторов не отвечает требованиям к соседству на другом маршрутизаторе.

- `set protocols isis interface <interface> metric <metric>`

Устанавливает метрику по умолчанию для цепочки маршрутизаторов (1 - 16777215). Максимальное значение зависит от ширины метрики.

- `set protocols isis interface <interface> network point-to-point`

Задает тип сети маршрутизаторов, как «точка-точка». По умолчанию тип сети - широковещательная.

- `set protocols isis interface <interface> passive`

Переводит выбранный интерфейс в пассивный режим.

- `set protocols isis interface <interface> password plaintext-password <text>`

Настраивает для сетевого интерфейса аутентификацию по заданному паролю.

- `set protocols isis interface <interface> priority <number>`

Задает для сетевого интерфейса приоритет выбора промежуточной системы DIS. Приоритет задается числом в диапазоне 0 - 127.

- `set protocols isis interface <interface> psnp-interval <number>`

Задает временной интервал отправки пакетов PSNP. Интервал задается в диапазоне 0 - 127.

- `set protocols isis interface <interface> no-three-way-handshake`

Отключает согласование рукопожатия handshake для соединений «точка-точка» с соседствующими узлами, описание которого представлено в спецификации RFC 5303. По умолчанию рукопожатие включен.

Настройка редистрибуции для протокола IS-IS

- `set protocols isis redistribute ipv4 <route source> level-1`

Выполняет редистрибуцию маршрутной информации для заданного источника в таблицы, используемые протоколом IS-IS, в качестве данных маршрутизатора станции.

- `set protocols isis redistribute ipv4 <route source> level-2`

Выполняет редистрибуцию маршрутной информации для заданного источника в таблицы, используемые протоколом IS-IS, в качестве данных маршрутизатора области.

- `set protocols isis redistribute ipv4 <route source> <level-1|level-2> metric <number>`

Задает метрику для маршрутов, в отношении которых выполнена редистрибуция для заданного источника. Метрика задается в диапазоне 1 - 16777215.

- `set protocols isis redistribute ipv4 <route source> <level-1|level-2> route-map <name>`

Позволяет использовать определенную карту маршрутов для отбора маршрутов заданного источника, в отношении которых выполняется редистрибуция.

Примечание

Операционная система поддерживает шесть источников маршрутной информации: bgp, connected, kernel, ospf, rip, static.

Настройка таймеров для протокола IS-IS

- `set protocols isis lsp-gen-interval <seconds>`

Задает минимальный временной интервал между генерацией одинаковых структур LSP. Интервал задается в диапазоне 1 - 120.

- `set protocols isis lsp-refresh-interval <seconds>`

Задает временной интервал обновления структуры LSP. Обычно протокол IS-IS генерирует структуру LSP только при изменении состояния соединения однако, чтобы убедиться в сходимости маршрутных данных на всех маршрутизаторах, структуры LSP генерируются на регулярной основе вне зависимости от состояния соединений. Интервал задается в диапазоне 1 - 65235. По умолчанию - 900.

```
set protocols isis max-lsp-lifetime <seconds>
```

Задает максимальное время жизни структуры LSP. Время задается в диапазоне 350 - 65535. По умолчанию - 1200. Если в течение заданного времени структура LSP не обновляется, она удаляется из базы данных. Так как есть возможность установить временной интервал обновления структуры LSP, он должен быть меньше по значению, чем время жизни структуры, в противном случае структуры LSP будут удаляться до их обновления.

```
set protocols isis spf-interval <seconds>
```

Задает минимальный временной интервал между выполнением процедур поиска кратчайшего пути. Интервал задается в диапазоне 1 - 120.

```
set protocols isis spf-delay-ietf holddown <milliseconds>
set protocols isis spf-delay-ietf init-delay <milliseconds>
set protocols isis spf-delay-ietf long-delay <milliseconds>
set protocols isis spf-delay-ietf short-delay <milliseconds>
set protocols isis spf-delay-ietf time-to-learn <milliseconds>
```

Настраивают механизм конечного автомата, основной задачей которого является контроль времени выполнения процедур поиска кратчайшего пути при регистрации событий, характерных для протоколов IGP. Описание процедуры поиска кратчайшего пути представлено в спецификации *RFC 8405*.

RIP - Routing Information Protocol

Общая информация о протоколе RIP

Протокол RIP является распространенным протоколом из категории IGP протоколов для маршрутизации внутри локальной сети. Протокол RIP был разработан в 1970-х годах компанией Xerox как часть другого протокола маршрутизации - XNS. Протокол RIP является дистанционно-векторным протоколом и основан на алгоритме поиска кратчайшего пути Беллмана-Форда. Маршрутизатор с поддержкой протокола RIP периодически отправляет соседствующим узлам обновления маршрутных данных, обеспечивая таким образом схождение физической топологии сети. Каждое обновление содержит параметр - дистанцию до определенных подсетей, который рассылается по всем соседствующим узлам (другим маршрутизаторам).

Поддерживаются следующие версии протокола RIP:

- **RIPv1** - описание представлено в спецификации *RFC 1058*;
- **RIPv2** - описание представлено в спецификации *RFC 2453*.

Основные настройки RIP

- `set protocols rip network <A.B.C.D/M>`

Активирует протокол RIP на интерфейсе, определенном указанной подсетью.

- `set protocols rip interface <interface>`

Активирует протокол RIP на интерфейсе, указанном явно по имени. На указанном интерфейсе будут включены передача и получение пакетов по протоколу RIP.

- `set protocols rip neighbor <A.B.C.D>`

Задает адрес соседствующего узла явно, если соседствующий узел не способен обрабатывать пакеты группового вещания (multicast). Не все маршрутизаторы могут обрабатывать пакеты группового вещания (multicast) - пакеты, которые адресованы целой сети или группе хостов в ней. В таких случаях необходимо устанавливать прямое соединение между маршрутизаторами.

- `set protocols rip passive-interface interface <interface>`

Переводит интерфейс в пассивный режим. В пассивном режиме все входящие пакеты интерфейс обрабатывает обычным образом, и операционная система не отправляет с него ни пакеты группового вещания (multicast) ни одноадресные пакеты (unicast) по протоколу RIP, за исключением пакетов в адрес заданного конкретного соседствующего узла.

- `set protocols rip passive-interface interface default`

Переводит все интерфейсы в пассивный интерфейс.

Дополнительные настройки RIP

- `set protocols rip default-distance <distance>`

Задает административную дистанцию по умолчанию (1 - 255).

> **Примечание**
>
> *Маршруты с административной дистанцией 255 фактический определяются операционной системой как отключенные и не устанавливаются в ядро.*

- `set protocols rip network-distance <A.B.C.D/M> distance <distance>`

Задает административную дистанцию для определенного префикса.

- `set protocols rip network-distance <A.B.C.D/M> access-list <name>`

Задает список доступа для определенного префикса. Команда может использоваться вместе с командой ***set protocols rip network-distance <A.B.C.D/M> distance <distance>***.

- `set protocols rip default-information originate`

Генерирует маршрут по умолчанию.

- `set protocols rip distribute-list access-list <in|out> <number>`

Накладывает фильтр на путь, определяемый протоколом RIP, с помощью списка доступа. Параметры *in* и *out* позволяют определить направление, к которому применяется список доступа.

- `set protocols rip distribute-list interface <interface> access-list <in|out> <number>`

Накладывает фильтр на путь, определяемый протоколом RIP, с помощью списка доступа. Можно указать интерфейс, к которому применяется список доступа.

- `set protocols rip distribute-list prefix-list <in|out> <name>`

Накладывает фильтр на путь, определяемый протоколом RIP, с помощью списка префиксов. Параметры *in* и *out* позволяют определить направление, к которому применяется список префиксов.

- `set protocols rip distribute-list interface <interface> prefix-list <in|out> <name>`

Накладывает фильтр на путь, определяемый протоколом RIP, с помощью списка префиксов. Можно указать интерфейс, к которому применяется список префиксов.

- `set protocols rip route <A.B.C.D/M>`

Задает статический маршрут только для внутреннего использования протоколом RIP.

Примечание

> *Команда применима только для операционной системы и подсистемы динамической маршрутизации FRR. Рекомендуется использовать с осторожность и только продвинутым пользователям, которые знакомы с деталями технической реализации протокола RIP. В большинстве случаев рекомендуется создавать статический маршрут в операционной системе и выполнять его редистрибуцию в контекст протокола RIP с помощью команды redistribute static.*

```
set protocols rip timers update <seconds>
```

Задает таймер обновления данных по протоколу RIP в диапазоне 5 - 2147483647. Значение таймера обновления по умолчанию - 30. По прошествии заданного количества секунд протокол RIP в инициативном порядке отправляет всем соседствующим узлам сообщение, содержащее полный список маршрутов из таблицы маршрутизации.

```
set protocols rip timers timeout <seconds>
```

Задает тайм-аут по истечении которого маршрут определяется как недоступный. Значение задается в диапазоне 5 - 2147483647, по умолчанию - 180. При этом маршрут остается в таблице маршрутизации еще некоторое время, чтобы соседствующие узлы были проинформированы об этом.

```
set protocols rip timers garbage-collection <seconds>
```

Задает таймер для окончательной очистки недоступных маршрутов. Значение задается в диапазоне 5 - 2147483647, по умолчанию - 120. По прошествии заданного количества секунд недоступный маршрут окончательно удаляется из таблицы маршрутизации.

Редистрибуция маршрутов RIP

```
set protocols rip redistribute <route source>
```

Выполняет редистрибуцию маршрутной информации от определенного источника в таблицы, используемые протоколом RIP.

```
set protocols rip redistribute <route source> metric <metric>
```

Задает метрику для маршрутов, в отношении которых выполняется редистрибуция в диапазоне 1 - 16.

```
set protocols rip redistribute <route source> route-map <name>
```

Настраивает определенную карту для выбора маршрутов, в отношении которых выполняется редистрибуция.

```
set protocols rip default-metric <metric>
```

Изменяет значение метрики по умолчанию (число пересылок между узлами) для маршрутов, в отношении которых выполняется редистрибуция. Значение задается в диапазоне 1 - 16. Значение по умолчанию - 1. Команда не оказывает никакого влияния на маршрут до непосредственно подключенных к маршрутизатора подсетей - connected, даже если для него тоже определена редистрибуция посредством команды redistribute connected. Чтобы маршруты типа connected также подвергались редистрибуции, необходимо использовать команду redistribute connected metric.

Примечание

Операционная система поддерживает шесть источников маршрутной информации: bgp, connected, kernel, ospf, rip, static.

Настройка интерфейсов

- `set interfaces <inttype> <intname> ip rip authentication plaintext-password <text>`

Выбирает интерфейс, для которого в рамках протокола RIP будет выполняться базовая аутентификация - аутентификация по паролю. Длина пароля не должна превышать 16 символов.

- `set interfaces <inttype> <intname> ip rip authentication md5 <id> password <text>`

Задает интерфейс, для которого в рамках протокола RIP будет выполняться аутентификация по ключу MD5. Длина ключа не должна превышать 16 символов.

- `set interfaces <inttype> <intname> ip rip split-horizon disable`

Отключает на интерфейсе механизм split horizon, который обычно используется дистанционно-векторными протоколами для предотвращения петель в маршрутах, во избежание которых операционная система по умолчанию не анонсирует маршруты с интерфейса, на который они были получены ранее.

- `set interfaces <inttype> <intname> ip rip split-horizon poison-reverse`

Включает обратное анонсирование источнику маршрутной информации маршрута, который был получен ранее от него, но с максимальной административной дистанцией. Такой механизм называется Poison Reverse. Если данный механизм работает совместно с механизмом split horizon, операционная система анонсирует маршруты с интерфейса, с которого они были получены, как недостижимые.

Мониторинг состояния RIP

- `show ip rip`

Выводит на экран маршруты, полученные в ходе использования протокола RIP.

```
show ip rip status
```

Выводит на экран текущий статус механизма, реализующего протокол RIP. Статус включает такую информацию, как значения таймеров, заданные фильтры, версию протокола RIP, интерфейс, на котором запущен механизм и данные по соседствующим узлам.

Групповая рассылка

Операционная система обеспечивает групповую рассылку IP пакетов посредством поддержки следующих протоколов:

- PIM;
- IGMP.

PIM и IGMP

Общая информация о протоколах PIM и IGMP

Протокол PIM может быть настроен на любом интерфейсе каждого маршрутизатора. Каждый маршрутизатор должен иметь заданную точку коммуникационной встречи - Rendevouz Point, из которой будут впоследствии выстраиваться однонаправленные общие деревья для дальнейшего их распространения через групповые рассылки.

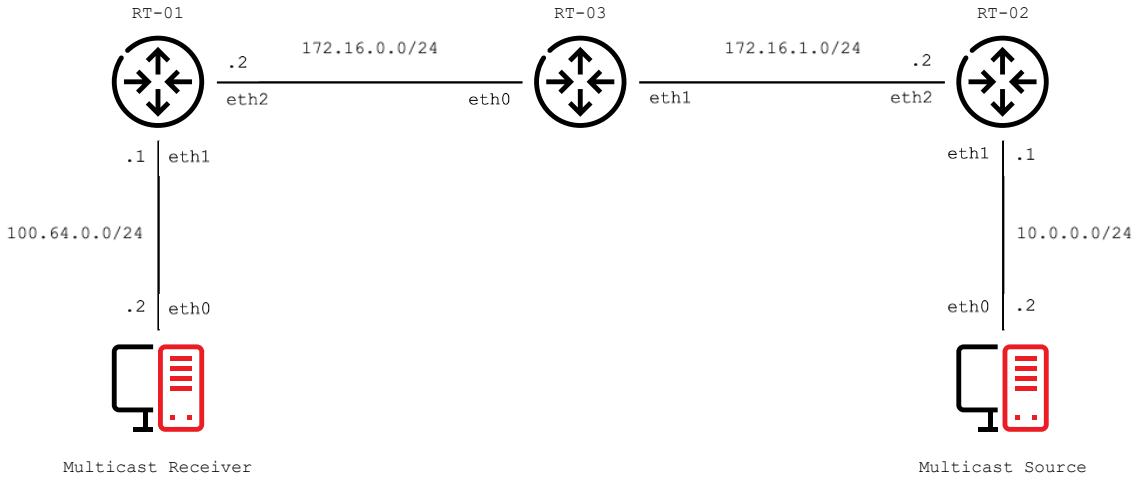
Трафик от источников в рамках групповой рассылки поступает в точку коммуникационной встречи, после чего получатели данного трафика извлекают его из общего с помощью протокола IGMP.

Получатели трафика в рамках групповой рассылки используют протокол IGMP для коммуникаций с ближайшим (локальным) маршрутизатором, поэтому кроме протокола PIM на каждом маршрутизаторе, который является получателем трафика в рамках групповой рассылки, должен быть также настроен протокол IGMP.

Операционная система поддерживает протокол IGMP версии 2 и версии 3, которая предоставляют возможность избирательных групповых рассылок на основе отправителя.

Пример настройки групповой рассылки

Пример базовой конфигурации групповых рассылок между тремя маршрутизаторами.



Список команд для настройки первого маршрутизатора:

```
set interfaces ethernet eth2 address '172.16.0.2/24'
set interfaces ethernet eth1 address '100.64.0.1/24'
set protocols ospf area 0 network '172.16.0.0/24'
set protocols ospf area 0 network '100.64.0.0/24'
set protocols igmp interface eth1
set protocols pim interface eth1
set protocols pim interface eth2
set protocols pim rp address 172.16.255.1 group '224.0.0.0/4'
```

Список команд для настройки второго маршрутизатора:

```
set interfaces ethernet eth1 address '10.0.0.1/24'
set interfaces ethernet eth2 address '172.16.1.2/24'
set protocols ospf area 0 network '10.0.0.0/24'
set protocols ospf area 0 network '172.16.1.0/24'
set protocols pim interface eth1
set protocols pim interface eth2
set protocols pim rp address 172.16.255.1 group '224.0.0.0/4'
```

Список команд для настройки третьего маршрутизатора:

```
set interfaces dummy dum0 address '172.16.255.1/24'
set interfaces ethernet eth0 address '172.16.0.1/24'
```

```
set interfaces ethernet eth1 address '172.16.1.1/24'
set protocols ospf area 0 network '172.16.0.0/24'
set protocols ospf area 0 network '172.16.255.0/24'
set protocols ospf area 0 network '172.16.1.0/24'
set protocols pim interface dum0
set protocols pim interface eth0
set protocols pim interface eth1
set protocols pim rp address 172.16.255.1 group '224.0.0.0/4'
```

Основные настройки протоколов PIM и IGMP

```
set protocols pim interface <interface-name>
```

Активирует использование протокола PIM на выбранном интерфейсе, чтобы обеспечить взаимодействие с соседствующими узлами в рамках протокола PIM.

```
set protocols pim rp address <address> group <multicast-address/mask-bits>
```

Настраивает точку коммуникационной встречи для протокола PIM с целью агрегации в ней сообщений групповой рассылки. Необходимо указать адрес точки. Можно указать префикс группы в качестве классификатора. Указанные значения должны быть общими для каждого маршрутизатора в сети с использованием протокола PIM.

```
set protocols igmp interface eth1
```

Активирует использованием протокола IGMP на выбранном интерфейсе, чтобы протокол PIM мог использовать получаемые с данного интерфейса отчеты и запросы по протоколу IGMP. По умолчанию используется протокол IGMP версии 3.

Детальные настройки протоколов PIM и IGMP

```
set protocols igmp interface eth1
```

Задает для интерфейса приоритет в рамках протокола PIM, который влияет на выбор узла сети в качестве основного маршрутизатора сегмента сети. Чем больше значение, тем выше приоритет при выборе основного маршрутизатора. Значение приоритета задается в диапазоне 1 - 4294967295. Значение по умолчанию по умолчанию - 1),

```
set protocols pim int <interface> hello <seconds>
```

Задает для интерфейса временной интервал отправки сообщений Hello в рамках протокола PIM. Значение интервала задается в диапазоне 1 - 180.

```
set protocols pim rp keep-alive-timer <seconds>
```

Задает тайм-аут для передачи потока данных о состоянии (S,G) из базы TIB. Значение тайм-аута задается в диапазоне 31 - 60000. 31 секунда выбрана в качестве минимального значения, так как некоторые аппаратные платформы не способны эффективно обрабатывать поток данных в течение более, чем 30 секунд.

- `set protocols igmp interface <interface> join <multicast-address> source <IP-address>`

Добавляет выбранный интерфейс в группу рассылки путем задания адреса групповой рассылки и IP-адреса отправителя.

- `set protocols igmp interface <interface> query-interval <seconds>`

Задает для интерфейса временной интервал для обработки запросов IGMP от хостов, которые будут использоваться протоколом PIM. Значение временного интервала задается в диапазоне 1 - 1800.

- `set protocols igmp interface <interface> query-max-response-time <deciseconds>`

Задает для интерфейса тайм-аут для обработки входящих запросов IGMP. Если отчет не поступил на данный интерфейс в течение указанного времени, состояния (S,G) или (*,G) из базы TIB считаются истекшими. Значение тайм-аута задается в диапазоне 10 - 250.

- `set protocols igmp interface <interface> version <version-number>`

Определяет для выбранного интерфейса, какая версия протокола IGMP - 2 или 3 - будет использоваться. По умолчанию используется протокол IGMP версии 3.

IGMP прокси

Общая информация об IGMP прокси

IGMP прокси отправляет сообщения IGMP от имени подключенных клиентов. Конфигурация IGMP прокси должна содержать только один интерфейс типа upstream и один или несколько интерфейсов типа downstream.

Пример настройки IGMP прокси

Внутренний (LAN) интерфейс eth1 доступен только после трансляции адресов (NAT). Чтобы подписать подсеть 10.0.0.0/23, которая подключена непосредственно к внешнему (WAN) интерфейсу eth0 на получение групповых рассылок, необходимо сконфигурировать IGMP прокси.

- `set protocols igmp-proxy interface eth0 role upstream`

```
set protocols igmp-proxy interface eth0 alt-subnet 10.0.0.0/23
set protocols igmp-proxy interface eth1 role downstream
```

Основные настройки IGMP прокси

```
set protocols igmp-proxy interface <interface> role <upstream | downstream>
```

Выбирает интерфейсы, на которых будет настроен IGMP.

Интерфейсы могут быть настроены в качестве:

- **upstream** - исходящий интерфейс, который используется для связи с источниками групповых рассылок.
- **downstream** - интерфейсы, используемые для распространения данных по подсетям получателей групповых рассылок, в которых клиенты могут объединяться в группы для получения данных.

```
set protocols igmp-proxy interface <interface> alt-subnet <network>
```

Указывает альтернативный источник групповых рассылок и сообщений по протоколу IGMP. Адрес подсети должен быть указан в формате A.B.C.D/M. По умолчанию маршрутизатор получает данные от источников из той же подсети, что и интерфейс. Если источник групповых рассылок находится в удаленной подсети, ее адрес должен быть указан, чтобы трафик от такого источника был обработан. Это актуально для интерфейсов типа upstream, так как обычно источник групповых рассылок находится в удаленной подсети.

```
set protocols igmp-proxy disable-quickleave
```

Отключает режим, в котором система не отправляет сообщение IGMP Leave с интерфейса типа upstream, если это сообщение уже было получено на любом из интерфейсов типа downstream. В этом режиме система также не запрашивает отчеты о принадлежности к группе с интерфейсов типа downstream, и, если отчет все же получен, группы не объединяются повторно по отношению к интерфейсу типа upstream.

Необходимо обеспечить работу системы в качестве фактического клиента групповой рассылки по отношению к интерфейсу upstream, поэтому режим должен быть включен. Однако в этом случае велик риск значительного расхода доступной на интерфейсе полосы пропускания.

```
set protocols igmp-proxy disable
```

Отключает IGMP прокси.

```
restart igmp-proxy
```

Перезагружает IGMP прокси.

BFD - Bidirectional Forwarding Detection protocol

Общая информация о протоколе BFD

Задача протокола BFD - быстрое определение недоступности соседствующего узла и принятие соответствующих мер. Протокол BFD отправляет значительное количество небольших UDP пакетов, чтобы убедиться, что соседствующих узел по-прежнему доступен. Это позволяет избежать использования лишний раз таймеров, заданных для протоколов BGP и OSPF.

Базовое описание протокола представлено в спецификации *RFC 5880*, а описание его расширения - в спецификациях *RFC 5881* и *RFC 5883*.

Основные настройки BFD

```
set protocols bfd peer <address>
```

Задает IPv4 или IPv6 адрес соседствующего узла.

```
set protocols bfd peer <address> echo-mode
```

Активирует режим передачи эхо-сообщений для выбранного узла.

```
set protocols bfd peer <address> multihop
```

Определяет соседствующий узел как не подключенный непосредственно.

```
set protocols bfd peer <address> source [address <address> |
interface <interface>]
```

Закрепляет выбранный узел за определенным интерфейсом или адресом устройства. Использование команды является обязательным при построении сети с IPv6 адресацией.

```
set protocols bfd peer <address> interval echo-interval <10-
60000>
```

Задает временной интервал для получения эхо-сообщений.

```
set protocols bfd peer <address> interval multiplier <2-255>
```

Задает множитель, который будет использован при определении временных интервалов.

- `set protocols bfd peer <address> interval [receive | transmit] <10-60000>`

Задает временной интервал получения/передачи данных по протоколу BFD. Значение временного интервала задается в миллисекундах.

- `set protocols bfd peer <address> shutdown`

Определяет соседствующий узел как отключенный.

Настройка интеграции BFD с BGP

- `set protocols bgp <asn> neighbor <address> bfd`

Активирует использование протокола BFD для отдельного соседствующего узла в контексте протокола BGP. Когда BFD-узел выходит из строя, он протокол BFD немедленно просит BGP выключить соединение с соседом, а когда он снова поднимается, уведомляет BGP о попытке соединения с ним.

- `set protocols bgp <asn> peer-group <group> bfd`

Активирует использование протокола BFD для группы соседствующих узлов в контексте протокола BGP. Когда BFD-узел, входящий в группу соседствующих узлов, выходит из строя, он протокол BFD немедленно просит BGP выключить соединение с соседом, а когда он снова поднимается, уведомляет BGP о попытке соединения с ним.

Настройка интеграции BFD с OSPF

- `set interfaces ethernet <interface> ip ospf bfd`

Активирует BFD в контексте протокола OSPF на интерфейсе. Каждый раз, когда обнаруживается новый сосед, создается BFD-пир для мониторинга состояния соединения и быстрой конвергенции.

- `set interfaces ethernet <interface> ipv6 ospfv3 bfd`

Активирует использование протокола BFD в контексте протокола OSPFv3 на интерфейсе. Каждый раз, когда обнаруживается новый сосед, создается BFD-пир для мониторинга состояния соединения и быстрой конвергенции.

Настройка интеграции BFD с ISIS

- `set protocols isis <name> interface <interface> bfd`

Активирует использование протокола BFD в контексте протокола ISIS интерфейсе. Каждый раз, когда обнаруживается новый сосед, создается BFD-пир для мониторинга состояния соединения и быстрой конвергенции. Если настроена поддержка как IPv4, так и IPv6, то будет создана только сессия BFD на базе IPv6.

Мониторинг состояния BFD

- `show protocols bfd peer`

Выводит на экран список соседствующих узлов для протокола BFD.

MPLS - Multiprotocol Label Switching

Общая информация об MPLS

Механизм MPLS представляет собой механизм перенаправления IP пакетов, отличающийся от стандартного. Вместо IP-адресов для принятия решения о перенаправлении пакета на исходящий интерфейс маршрутизатор использует специальный заголовок длиной 32 бита (4 байта), который называется MPLS меткой. Данная метка размещается между заголовком Ethernet фрейма (L2) и заголовком IP пакета (L3). Возможно как статическое, так и динамическое размещение MPLS меток, однако в настоящем разделе рассматривается только динамическое размещение с использованием одного из подходящих для данных целей протоколов/механизмов: протокола LDP, протокола RSVP, механизма SR с использованием протоколов OSPF или IS-IS. Указанные протокола используются фактически для формирования одноадресного пути до пункта назначения в сети, который идеологически схож с туннелем и именуется как коммутируемый на основе меток путь - LSP. Для упрощения понимания работы LSP можно провести его сравнение с GRE туннелем. Они отличаются с точки зрения технической реализации, но схожи между собой в части управления движением пакета через туннель. Рекомендуется рассматривать механизм MPLS как механизм туннелирования, позволяющий доставлять по сети различные типы пакетов, значительно упрощающий проектирование потоков и схем движения трафика в сети за счет использования протокола RSVP или механизма SR, и в основном упрощающий организацию движения потоков данных как внутри сети, так и при межсетевом взаимодействии.

> **Примечание**
>
> *Поддержка механизма MPLS в операционной системе имеет ряд ограничений. Например, механизм MPLS нельзя использовать совместно с механизмами защиты коммуникаций L3 VPN, L2 VPN, а также с механизмом динамического построения VPN сетей. Протокол RSVP нельзя использовать как базовый протокол маршрутизации в стеке по отношению к транспортному протоколу (ограничение компонента FRR, интегрированного в систему). В настоящий момент устройство под управлением операционной системы может быть сконфигурировано как MPLS маршрутизатор на последней миле.*

Протокол LDP

Архитектура механизма MPLS не предусматривает использование одного конкретного протокола для определения LSP. Операционная система использует протокол LDP в реализации, заложенной в компонент FRR и соответствующей спецификации RFC 5036

Протокол LDP представляет собой сигнальный протокол, использующий в качестве транспорта TCP и динамически распределяющий MPLS метки, создающий таким образом LSP. Протокол LDP не является протоколом маршрутизации сам по себе, но он использует данные, полученные в результате работы других протоколов маршрутизации, для принятия решения о перенаправлении пакетов. Для корректного использования протокола LDP на маршрутизаторе необходимо, чтобы другие маршрутизаторы также использовали данный протокол.

Для того, чтобы протокол LDP на маршрутизаторе обменивался анонсами меток с другими маршрутизаторами, устанавливается сессия TCP как с автоматически обнаруживаемыми, так и статически заданными маршрутизаторами. Протокол LDP предусматривает попытку установления TCP сессии с другими маршрутизаторами, используя для этого транспортные адреса, поэтому необходимо убедиться в том, что эти адреса присутствуют в таблице маршрутизации и являются достижимыми для обеспечения движения трафика.

Рекомендуется использовать одинаковые адреса для идентификации маршрутизатора в контексте протокола LDP и для автоматически обнаруживаемых маршрутизаторов, которые должны быть явно заданы в конфигурации протокола LDP в операционной системе.

Также необходимо помнить, что принципиальная реализация протокола LDP схожа с реализацией протокола BGP в том, что она использует TCP в качестве транспорта. Однако в отличие от протокола BGP реализация LDP не предусматривает возможность обновления оперативных данных. Это означает, что в случае изменения топологии соседствующих узлов необходимо вручную изменить конфигурацию протокола LDP, чтобы он функционировал корректно.

Пример настройки MPLS

Пример настройки механизма MPLS:

```
set protocols ospf area 0 network '192.168.255.252/32'
set protocols ospf area 0 network '192.168.0.5/32'
set protocols ospf parameters router-id '192.168.255.252'
set protocols mpls interface 'eth1'
set protocols mpls ldp discovery transport-ipv4-address
'192.168.255.252'
set protocols mpls ldp interface 'eth1'
set protocols mpls ldp interface 'lo'
set protocols mpls ldp router-id '192.168.255.252'
set interfaces ethernet eth1 address '192.168.0.5/31'
set interfaces loopback lo address '192.168.255.252/32'
```

Основные настройки MPLS

- `set protocols mpls interface <interface>`

Активирует использование механизма MPLS для выбранного интерфейса.

- `set protocols mpls ldp interface <interface>`

Активирует использование протокола LDP на выбранном интерфейсе.

- `set protocols mpls ldp router-id <address>`

Задет IP-адрес, используемый в качестве идентификатора маршрутизатора на устройстве.

- `set protocols mpls ldp discovery transport-ipv4-address <address>`

Задает IPv4 адрес в качестве транспортного, используемого протоколом LDP.

- `set protocols mpls ldp discovery transport-ipv6-address <address>`

Задает IPv6 адрес в качестве транспортного, используемого протоколом LDP.

- `set protocols mpls ldp neighbor <address> password <password>`

Настраивает параметры аутентификации для соседствующих узлов в контексте протокола LDP. Необходимо указать адрес соседствующего узла и пароль, который должен быть общим для соседствующих узлов.

- `set protocols mpls ldp neighbor <address> session-holdtime <seconds>`

Задает время удержания сессии для соседствующих узлов в контексте протокола LDP. Необходимо указать адрес соседствующего узла и время удержания сессии в секундах. Чтобы изменения вступили в силу, перезагрузите соседствующий узел.

- `set protocols mpls ldp neighbor <address> ttl-security <disable | hop count>`

Активирует механизм контроля TTL для соседствующих узлов в контексте протокола LDP (включить механизм и задать количество пересылок между узлами, отключить механизм). По умолчанию количество пересылок - 255, что эквивалентно максимальному значению TTL.

- `set protocols mpls ldp discovery hello-ipv4-interval <seconds>`

Задает временной интервал для отправки hello сообщений для протокола IPv4.

```
set protocols mpls ldp discovery hello-ipv4-holdtime <seconds>
```

Задает временной интервал отправки hello сообщений для протокола IPv4.

```
set protocols mpls ldp discovery hello-ipv6-interval <seconds>
```

Задает временной интервал удержания hello сообщений для протокола IPv6.

```
set protocols mpls ldp discovery hello-ipv6-holdtime <seconds>
```

Задает временной интервал удержания hello сообщений для протокола IPv6.

```
set protocols mpls ldp discovery session-ipv4-holdtime
<seconds>
```

Задает временной интервал удержания TCP сессий для протокола IPv4.

```
set protocols mpls ldp discovery session-ipv6-holdtime
<seconds>
```

Задает временной интервал удержания TCP сессий для протокола IPv6.

```
set protocols mpls ldp import ipv4 import-filter filter-access-
list <access list number>
```

Настраивает импорт классов эквивалентности - FEC от соседствующих узлов в контексте протокола LDP для протокола IPv4.

```
set protocols mpls ldp import ipv6 import-filter filter-access-
list6 <access list number>
```

Настраивает импорт классов эквивалентности - FEC от соседствующих узлов в контексте протокола LDP для протокола IPv6.

```
set protocols mpls ldp export ipv4 export-filter filter-access-
list <access list number>
```

Настраивает экспорт классов FEC для соседствующих узлов в контексте протокола LDP для протокола IPv4.

```
set protocols mpls ldp export ipv6 export-filter filter-access-
list6 <access list number>
```

Настраивает экспорт классов FEC для соседствующих узлов в контексте протокола LDP для протокола IPv6.

- `set protocols mpls ldp export ipv4 explicit-null`

Управляет анонсом классов FEC с меткой 0 для протокола IPv4. Метка 0 используется для явных нулевых операций в контексте протокола LDP.

- `set protocols mpls ldp export ipv6 explicit-null`

Управляет анонсом классов FEC с меткой 0 для протокола IPv6. Метка 0 используется для явных нулевых операций в контексте протокола LDP.

- `set protocols mpls ldp allocation ipv4 access-list <access list number>`

Задает фильтры, которые позволяют контролировать процесс размещения локальных классов FEC в контексте протокола LDP для протокола IPv4.

- `set protocols mpls ldp allocation ipv6 access-list6 <access list number>`

Задает фильтры, которые позволяют контролировать процесс размещения локальных классов FEC в контексте протокола LDP для протокола IPv6.

- `set protocols mpls ldp parameters cisco-interop-tlv`

Активирует режим использования формата данных, не соответствующего требованиям компании Cisco, который подразумевает отправку и интерпретацию структур TLV для коммуникаций по протоколу LDP поверх протокола IPv6. Описание данной особенности представлено в спецификации RFC 7552.

- `set protocols mpls ldp parameters ordered-control`

Активирует режим упорядоченного распространения меток. По умолчанию компонент FRR в составе операционной системы использует режим независимого распространения меток. Описание данной особенности представлено в спецификации RFC 5036.

- `set protocols mpls ldp parameters transport-prefer-ipv4`

Активирует режим использования протокола IPv4 в качестве предпочтительного для TCP коммуникаций между соседствующими маршрутизаторами в контексте протокола LDP. Рекомендуется использовать эту команду в случае, если на интерфейсе задан как IPv4 адрес, так и IPv6 адрес.

- `set protocols mpls ldp targeted-neighbor ipv4 enable`

Активирует режим принятия таргетированных LDP сессий данным маршрутизатором для протокола IPv4. В этом режиме маршрутизатор будет отвечать на любые попытки

установления сессий, адресованных именно ему, за исключением локальных TCP соединений.

- `set protocols mpls ldp targeted-neighbor ipv6 enable`

Активирует режим принятия таргетированных LDP сессий данным маршрутизатором для протокола IPv6. В этом режиме маршрутизатор будет отвечать на любые попытки установления сессий, адресованных именно ему, за исключением локальных TCP соединений.

- `set protocols mpls ldp targeted-neighbor ipv4 address <address>`

Активирует режим инициирования таргетированных LDP сессий данным маршрутизатором для протокола IPv4. В этом режиме маршрутизатор будет инициировать установление сессий с другим конкретным маршрутизатором.

- `set protocols mpls ldp targeted-neighbor ipv6 address <address>`

Активирует режим инициирования таргетированных LDP сессий данным маршрутизатором для протокола IPv6. В этом режиме маршрутизатор будет инициировать установление сессий с другим конкретным маршрутизатором.

- `set protocols mpls ldp targeted-neighbor ipv4 hello-holdtime <seconds>`

Задает временной интервал удержания hello сообщений для протокола IPv4 определенным соседствующим маршрутизаторам, которые являются адресатами таргетированных LDP сессий.

- `set protocols mpls ldp targeted-neighbor ipv4 hello-interval <seconds>`

Задает временной интервал отправки hello сообщений для протокола IPv4 определенным соседствующим маршрутизаторам, которые являются адресатами таргетированных LDP сессий.

- `set protocols mpls ldp targeted-neighbor ipv6 hello-holdtime <seconds>`

Задает временной интервал удержания hello сообщений для протокола IPv6 определенным соседствующим маршрутизаторам, которые являются адресатами таргетированных LDP сессий.

- `set protocols mpls ldp targeted-neighbor ipv6 hello-interval <seconds>`

Задает временной интервал отправки hello сообщений для протокола IPv6 определенным соседствующим маршрутизаторам, которые являются адресатами таргетированных LDP сессий.

- `reset mpls ldp neighbor <IPv4 or IPv6 address>`

Сбрасывает LDP сессии, установленные с определенным соседствующим маршрутизатором в контексте протокола LDP по указанному адресу.

Мониторинг состояния MPLS

- `show mpls ldp binding`

Выводит на экран информацию о метках из базы LIB.

- `show mpls ldp discovery`

Выводит на экран информацию о соседствующих маршрутизаторах в контексте протокола LDP, обнаруженных путем рассылки hello сообщений.

- `show mpls ldp interface`

Выводит на экран информацию о текущем статусе протокола LDP на интерфейсах.

- `show mpls ldp neighbor`

Выводит на экран общую информацию о соседствующих маршрутизаторах в контексте протокола LDP.

- `show mpls ldp neighbor detail`

Выводит на экран детальную информацию о соседствующих маршрутизаторах в контексте протокола LDP.

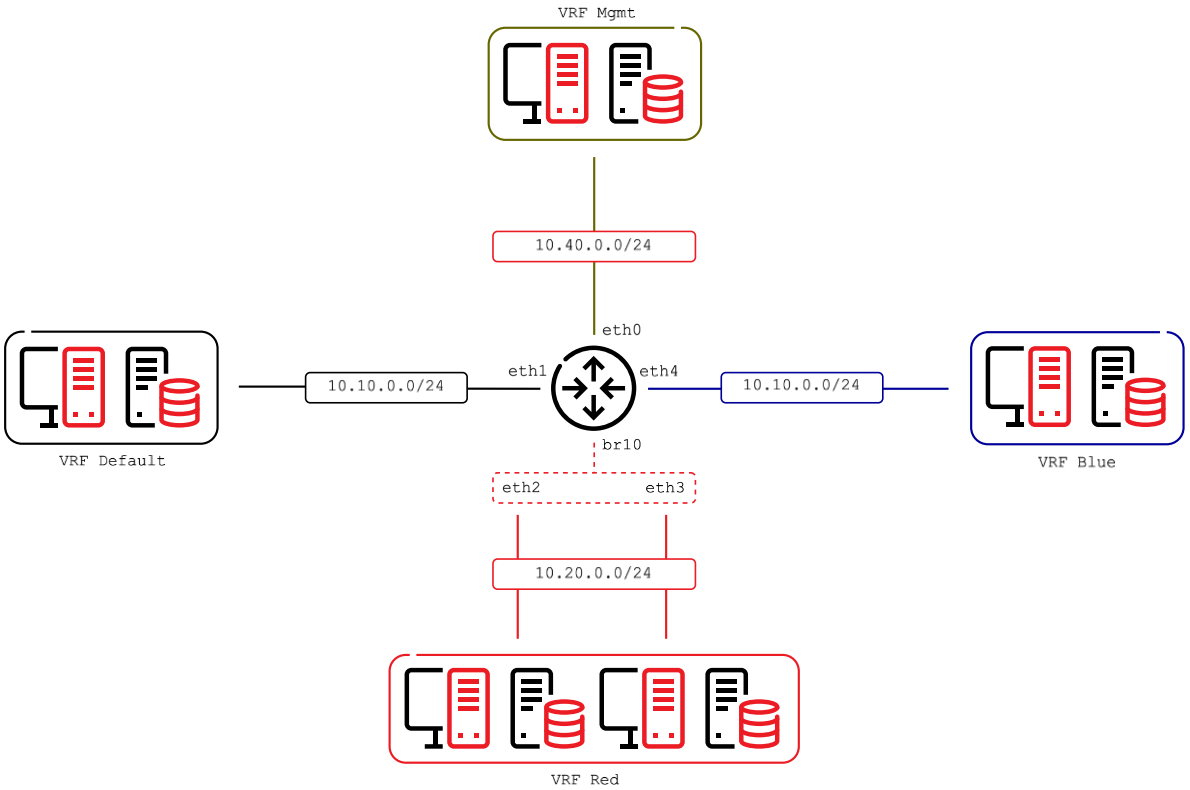
VRF - Virtual routing and forwarding

Общая информация о VRF

VRF – технология, позволяющая реализовывать на базе одного физического маршрутизатора несколько виртуальных – каждого со своей независимой таблицей маршрутизации. Преимуществом виртуальной маршрутизации является полная изоляция маршрутов как между двумя виртуальными маршрутизаторами, так и между виртуальным и реальным. Устройства VRF совместно с правилами обработки трафика на уровне IP (IP Rules) позволяют создать виртуальные маршрутизаторы. При создании устройства VRF создается ассоциированная с ним таблица маршрутизации. После создания устройства VRF его необходимо привязать к определенному сетевому интерфейсу.

Пример настройки VRF

Пример топологии с использованием экземпляров VRF:



- Рабочая станция *PC1* располагается в зоне *VRF default* и выполняет роль файлового сервера.
- Рабочая станция *PC2* располагается в зоне *VRF* blue которая соответствует отделу разработки.
- Рабочие станции *PC3* и *PC4* подключены к мосту (bridge) на маршрутизаторе *R1*, который располагается в зоне *VRF red*, соответствующей отделу кадров.
- Маршрутизатор *R1* управляется через выделенную сеть, которая располагается в зоне *VRF mgmt*.

Применение настроек на маршрутизаторе:

```
set interfaces bridge br10 address '10.30.0.254/24'
set interfaces bridge br10 member interface eth3
set interfaces bridge br10 member interface eth4
set interfaces bridge br10 vrf 'red'

set interfaces ethernet eth0 address 'dhcp'
```

```
set interfaces ethernet eth0 vrf 'mgmt'
set interfaces ethernet eth1 address '10.0.0.254/24'
set interfaces ethernet eth2 address '10.20.0.254/24'
set interfaces ethernet eth2 vrf 'blue'

set protocols static interface-route 10.20.0.0/24 next-hop-interface eth2 next-hop-vrf 'blue'
set protocols static interface-route 10.30.0.0/24 next-hop-interface br10 next-hop-vrf 'red'
set protocols vrf blue static interface-route 10.0.0.0/24 next-hop-interface eth1 next-hop-vrf 'default'
set protocols vrf red static interface-route 10.0.0.0/24 next-hop-interface eth1 next-hop-vrf 'default'

set service ssh disable-host-validation
set service ssh vrf 'mgmt'

set system domain-name 'fakel.net'
set system host-name 'R1'
set system name-server 'eth0'

set vrf name blue table '3000'
set vrf name mgmt table '1000'
set vrf name red table '2000'
```

После применения настрое на физическом маршрутизаторе для каждого экземпляра VRF появится своя таблица маршрутизации:

Таблица маршрутизации по умолчанию:

```
fakel@R1:~$ show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, D - SHARP,
       F - PBR, f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
```

```
C>* 10.0.0.0/24 is directly connected, eth1, 00:07:44
S>* 10.20.0.0/24 [1/0] is directly connected, eth2 (vrf blue),
weight 1, 00:07:38
S>* 10.30.0.0/24 [1/0] is directly connected, br10 (vrf red),
weight 1, 00:07:38
```

Таблица маршрутизации Red:

```
fakel@R1:~$ show ip route vrf red
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, D - SHARP,
       F - PBR, f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r -
rejected, b - backup


VRF red:
K>* 0.0.0.0/0 [255/8192] unreachable (ICMP unreachable), 00:07:57
S>* 10.0.0.0/24 [1/0] is directly connected, eth1 (vrf default),
weight 1, 00:07:40
C>* 10.30.0.0/24 is directly connected, br10, 00:07:54
```

Таблица маршрутизации Blue:

```
fakel@R1:~$ show ip route vrf blue
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, D - SHARP,
       F - PBR, f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r -
rejected, b - backup


VRF blue:
K>* 0.0.0.0/0 [255/8192] unreachable (ICMP unreachable), 00:08:00
S>* 10.0.0.0/24 [1/0] is directly connected, eth1 (vrf default),
weight 1, 00:07:44
C>* 10.20.0.0/24 is directly connected, eth2, 00:07:53
```

Основные настройки VRF

- `set vrf name <name>`

Создает новый экземпляр VRF с именем, указанным в параметре <*name*>. Заданное имя используется в дальнейшем при организации привязки сетевых интерфейсов к созданному экземпляру VRF.

```
set vrf name <name> table <id>
```

Связывает таблицу маршрутизации определенного идентификатора <*id*> с экземпляром VRF <*name*>.

> **Примечание**
>
> *Заданный идентификатор таблицы маршрутизации не может быть изменен после того, как он был привязан к экземпляру VRF. Это возможно только при удалении экземпляра VRF и созданию нового экземпляра VRF.*

```
set vrf bind-to-all
```

Активирует работу сервисов TCP и UDP, запущенных в контексте экземпляра VRF по умолчанию, также в контекстах всех остальных экземпляров VRF, несмотря на отсутствие фактической привязки к данным экземплярам VRF.

> **Примечание**
>
> *По умолчанию при назначении портов все свободные сокеты связываются с экземпляром VRF по умолчанию. Такая конфигурация будет игнорироваться сетевыми пакетами, поступающими на сетевой интерфейс, связанный с другим экземпляром VRF.*

```
set interfaces <dummy | ethernet | bonding | bridge | pppoe>
<interface> vrf <name>
```

Связывает сетевой интерфейс <*interface*> с экземпляром VRF <*name*>.

Настройка маршрутизации для VRF

Статические маршруты

```
set protocols vrf <name> static route <subnet> next-hop
<address>
```

Задает адрес <*address*> следующего маршрутизатора для статического маршрута IPv4 <*subnet*> в экземпляре VRF <*name*>. Для экземпляра VRF можно создавать несколько статических маршрутов IPv4.

```
set protocols vrf <name> static route <subnet> next-hop
<address> disable
```

Отключает использование записи со статическим маршрутом IPv4 <*subnet*> в экземпляре VRF <*name*>.

```
set protocols vrf <name> static route <subnet> next-hop
<address> distance <distance>
```

Задает административную дистанцию <*distance*> до следующего маршрутизатора <*address*> для маршрута IPv4 <*subnet*> в экземпляре VRF <*name*>. Маршруты с наименьшей административной дистанцией - в приоритете перед маршрутами с наибольшей административной дистанцией. Административную дистанцию можно определить в диапазоне от 1 до 255. Значение административной дистанции по умолчанию - 1.

```
set protocols vrf <name> static route6 <subnet> next-hop
<address>
```

Задает адрес <*address*> следующего маршрутизатора для статического маршрута IPv6 <*subnet*> в экземпляре VRF <*name*>. Для устройства VRF можно создавать несколько статических маршрутов IPv6.

```
set protocols vrf <name> static route6 <subnet> next-hop
<address> disable
```

Отключает использование записи со статическим маршрутом IPv6 <*subnet*> в экземпляре VRF <*name*>.

```
set protocols vrf <name> static route6 <subnet> next-hop
<address> distance <distance>
```

Задает административную дистанцию <*distance*> до следующего маршрутизатора <*address*> для маршрута IPv6 <*subnet*> в экземпляре VRF <*name*>. Маршруты с наименьшей административной дистанцией - в приоритете перед маршрутами с наибольшей административной дистанцией. Административную дистанцию можно определить в диапазоне от 1 до 255. Значение административной дистанции по умолчанию - 1.

Утечка маршрутов

```
set protocols vrf <name> static route <subnet> next-hop
<address> next-hop-vrf <default | vrf-name>
```

Добавляет статический маршрут IPv4 в таблицу маршрутизации экземпляра VRF <*name*> для доступа к подсети <*subnet*> через следующий маршрутизатор <*address*> с адресом, принадлежащим другому экземпляру VRF <*default* | *vrf-name*>. Используется в случае, если есть общие сервисы или маршруты, которые должны быть общими для нескольких экземпляров VRF.

- `set protocols vrf <name> static route6 <subnet> next-hop <address> next-hop-vrf <default | vrf-name>`

Добавляет статический маршрут IPv6 в таблицу маршрутизации экземпляра VRF *<name>* для доступа к подсети *<subnet>* через следующий маршрутизатор *<address>* с адресом, принадлежащим другому экземпляру VRF *<default | vrf-name>*. Используется в случае, если есть общие сервисы или маршруты, которые должны быть общими для нескольких экземпляров VRF.

Маршруты для интерфейсов

- `set protocols vrf <name> static interface-route <subnet> next-hop-interface <interface>`

Определяет интерфейс *<interface>* следующего маршрутизатора для статического маршрута IPv4 *<subnet>* на основе интерфейса для экземпляра VRF *<name>*. Интерфейс *<interface>* будет являться интерфейсом следующего узла, через который будет направляться трафик для заданной подсети *<subnet>*.

- `set protocols vrf <name> static interface-route <subnet> next-hop-interface <interface> disable`

Отключает использование статического маршрута IPv4 *<subnet>* на основе интерфейса *<interface>* для экземпляра VRF *<name>*.

- `set protocols vrf <name> static interface-route <subnet> next-hop-interface <interface> distance <distance>`

Задает административную дистанцию *<distance>* до следующего маршрутизатора на основе интерфейса *<interface>* для маршрута IPv4 *<subnet>* в экземпляре VRF *<name>*. Маршруты с наименьшей административной дистанцией - в приоритете перед маршрутами с наибольшей административной дистанцией. Административную дистанцию можно определить в диапазоне от 1 до 255. Значение административной дистанции по умолчанию - 1.

- `set protocols vrf <name> static interface-route6 <subnet> next-hop-interface <interface>`

Определяет интерфейс *<interface>* следующего маршрутизатора для статического маршрута IPv6 *<subnet>* на основе интерфейса для экземпляра VRF *<name>*. Интерфейс *<interface>* будет являться интерфейсом следующего узла, через который будет направляться трафик для заданной подсети *<subnet>*.

- `set protocols vrf <name> static interface-route6 <subnet> next-hop-interface <interface> disable`

Отключает использование статического маршрута IPv6 *<subnet>* на основе интерфейса *<interface>* для экземпляра VRF *<name>*.

- `set protocols vrf <name> static interface-route6 <subnet> next-hop-interface <interface> distance <distance>`

Задает административную дистанцию <*distance*> до следующего маршрутизатора на основе интерфейса <*interface*> для маршрута IPv6 <*subnet*> в экземпляре VRF <*name*>. Маршруты с наименьшей административной дистанцией - в приоритете перед маршрутами с наибольшей административной дистанцией. Административную дистанцию можно определить в диапазоне от 1 до 255. Значение административной дистанции по умолчанию - 1.

Нулевой маршрут

Нулевой маршрут — это маршрут, для которого система отбрасывает подходящие пакеты без ответа. Использование нулевого маршрута позволяет предотвратить утечку данных о топологии сети через публичные интерфейсы, но не мешает использовать ее в качестве более специфического маршрута внутри сети.

- `set protocols vrf <name> static route <subnet> blackhole`

Создает нулевой маршрут IPv4 <*subnet*> в экземпляре VRF <*name*>.

- `set protocols vrf <name> static route <subnet> blackhole distance <distance>`

Задает административную дистанцию <*distance*> для нулевого маршрута IPv4 <*subnet*> в экземпляре VRF <*name*>. Маршруты с наименьшей административной дистанцией - в приоритете перед маршрутами с наибольшей административной дистанцией.

- `set protocols vrf <name> static route6 <subnet> blackhole`

Создает нулевой маршрут IPv6 <subnet> в экземпляре VRF <*name*>.

- `set protocols vrf <name> static route6 <subnet> blackhole distance <distance>`

Задает административную дистанцию <*distance*> для нулевого маршрута IPv6 <*subnet*> в экземпляре VRF <*name*>. Маршруты с наименьшей административной дистанцией - в приоритете перед маршрутами с наибольшей административной дистанцией.

Мониторинг состояния VRF

- `show vrf`

Выводит на экран список созданных экземпляров VRF.

- `show vrf <name>`

Выводит на экран информацию об определенном <*name*> экземпляре VRF с именем.

- `show ip route vrf <name>`

Выводит на экран содержимое таблицы маршрутизации IPv4 для определенного <*name*> экземпляра VRF.

- `show ipv6 route vrf <name>`

Выводит на экран содержимое таблицы маршрутизации IPv6 для определенного <*name*> экземпляра VRF.

- `ping <host> vrf <name>`

Проверяет доступности или недоступности сетевого узла.

- `traceroute vrf <name> [ipv4 | ipv6] <host>`

Отображает маршрут следования данных до определенного хоста. При использовании опции IPv4 или IPv6 отображаются пакеты к семейству IP-адресов данного хоста. Эта опция полезна, когда указанный хост представляет собой имя хоста, а не IP-адрес.

Защищенные коммуникации

В ПО **Факел** защищенные коммуникации реализованы следующими механизмами:

- Протокол IPsec
- Сервер L2TP
- Сервер OpenConnect
- Сервер PPTP
- Ключевая схема RSA
- DMVPN
- Site-to-Site VPN

Протокол IPSec

Общая информация о протоколе IPSec

Протокол IPSec — это набор протоколов для защиты данных, передаваемых по сетям IP (Internet Protocol), в том числе в интернете. IPSec обеспечивает конфиденциальность, целостность и аутентификацию информации на сетевом уровне.

IPSec работает на сетевом уровне (Layer 3 OSI модели), обеспечивая безопасность для всех приложений, которые используют сеть IP для своей коммуникации, без необходимости изменения отдельных приложений. Это делает IPSec особенно ценным для создания защищенных соединений на уровне всей сети.

Ключевые механизмы IPSec:

Аутентификация заголовка (AH)

AH обеспечивает подтверждение подлинности отправителя и гарантирует, что пакет данных не был изменен во время передачи. Однако AH не шифрует данные, поэтому содержимое пакета остается видимым для кого угодно, кто перехватит трафик.

Защита полезной нагрузки (ESP)

ESP предлагает более комплексный уровень безопасности, обеспечивая шифрование данных для конфиденциальности, а также аутентификацию данных и отправителя, чтобы защитить информацию от изменений и подтвердить источник. Это делает ESP более предпочтительным вариантом для большинства приложений IPSec.

Протокол обмена ключами Internet Key Exchange (IKE)

IKE играет центральную роль в установлении и поддержании безопасных соединений IPSec. Он используется для автоматического согласования параметров безопасности и обмена ключами между участниками соединения. IKE работает в два этапа:

- **IKE Phase 1**: Цель этого этапа — установить защищенный канал (IKE SA) для безопасного обмена ключами. Здесь участники согласовывают политику безопасности и аутентифицируют друг друга, обычно используя предварительно разделяемые ключи, сертификаты или методы аутентификации на основе асимметричного шифрования.
- **IKE Phase 2**: На этом этапе устанавливаются параметры безопасности (SA) для защиты самих данных, передаваемых по IPSec. Создаются новые ключи сессии, и участники могут начать защищенный обмен данными.

Режимы работы IPSec

- **Транспортный режим**: Шифруется только полезная нагрузка каждого пакета, оставляя IP-заголовок нетронутым. Используется для защиты трафика между конечными узлами, например, между клиентом и сервером.
- **Туннельный режим**: Шифрует весь пакет, включая исходный IP-заголовок, и инкапсулирует его в новый IP-пакет с новым заголовком. Это идеально подходит для VPN, где требуется защита данных между двумя сетевыми устройствами, например, между двумя шлюзами.

Security Association

SA (Security Association) — это фундаментальное понятие в IPSec, представляющее собой набор параметров, которые определяют, как данные будут обрабатываться при использовании IPSec. SA включает в себя множество атрибутов, таких как метод шифрования, алгоритм хеширования, режим (транспортный или туннельный) и ключи.

Применение IPSec

IPSec широко используется для создания VPN, защиты данных на транспортном уровне и в качестве основы для безопасных соединений между различными сегментами сети. Он обеспечивает комплексную защиту, охватывающую аутентификацию, целостность и конфиденциальность данных.

Несмотря на свою сложность и потенциальные сложности с настройкой, IPSec остается ключевым компонентом современной сетевой безопасности благодаря его гибкости, масштабируемости и сильной защите.

Пример настройки протокола IPSec

IPSec с использованием GRE

Первый вариант развертывания VPN - настройка политики IPSec для учета пакетов протокола GRE, которыми обмениваются внешние адреса маршрутизаторов. Это лучший вариант, если оба маршрутизатора имеют статические внешние адреса.

Предположим, что маршрутизатор *LEFT* имеет внешний адрес *192.0.2.10* на интерфейсе *eth0*, а маршрутизатор *RIGHT - 203.0.113.45*.

Список команд для настройки первого маршрутизатора (RIGHT):

Настройки GRE туннеля

```
set interfaces tunnel tun0 encapsulation gre
set interfaces tunnel tun0 source-address 192.0.2.10
set interfaces tunnel tun0 remote 203.0.113.45
set interfaces tunnel tun0 address 10.10.10.1/30
```

Настройка интерфейса IPSec

```
set vpn ipsec ipsec-interfaces interface eth0
```

Настройка группы алгоритмов IKE

```
set vpn ipsec ike-group MyIKEGroup proposal 1 dh-group '2'
set vpn ipsec ike-group MyIKEGroup proposal 1 encryption
'aes128'
set vpn ipsec ike-group MyIKEGroup proposal 1 hash 'sha1'
```

Группа настроек алгоритмов ESP

```
set vpn ipsec esp-group MyESPGroup proposal 1 encryption
'aes128'
set vpn ipsec esp-group MyESPGroup proposal 1 hash 'sha1'
```

Настройка туннеля Ipsec

```
set vpn ipsec site-to-site peer 203.0.113.45 authentication
mode pre-shared-secret
set vpn ipsec site-to-site peer 203.0.113.45 authentication
pre-shared-secret MYSECRETKEY
set vpn ipsec site-to-site peer 203.0.113.45 ike-group
MyIKEGroup
set vpn ipsec site-to-site peer 203.0.113.45 default-esp-group
MyESPGroup
set vpn ipsec site-to-site peer 203.0.113.45 local-address
192.0.2.10
```

Настройка политики IPSec с учетом GRE

```
set vpn ipsec site-to-site peer 203.0.113.45 tunnel 1 protocol gre
```

Список команд для настройки второго маршрутизатора (LEFT):

Настройки GRE туннеля

```
set interfaces tunnel tun0 encapsulation gre
set interfaces tunnel tun0 source-address 203.0.113.45
set interfaces tunnel tun0 remote 192.0.2.10
set interfaces tunnel tun0 address 10.10.10.2/30
```

Настройка интерфейса IPSec

```
set vpn ipsec ipsec-interfaces interface eth0
```

Настройка группы алгоритмов IKE

```
set vpn ipsec ike-group MyIKEGroup proposal 1 dh-group '2'
set vpn ipsec ike-group MyIKEGroup proposal 1 encryption 'aes128'
set vpn ipsec ike-group MyIKEGroup proposal 1 hash 'sha1'
```

Группа настроек алгоритмов ESP

```
set vpn ipsec esp-group MyESPGroup proposal 1 encryption 'aes128'
set vpn ipsec esp-group MyESPGroup proposal 1 hash 'sha1'
```

Настройка туннеля Ipsec

```
set vpn ipsec site-to-site peer 192.0.2.10 authentication mode pre-shared-secret
set vpn ipsec site-to-site peer 192.0.2.10 authentication pre-shared-secret MYSECRETKEY
set vpn ipsec site-to-site peer 192.0.2.10 ike-group MyIKEGroup
set vpn ipsec site-to-site peer 192.0.2.10 default-esp-group MyESPGroup
set vpn ipsec site-to-site peer 192.0.2.10 local-address 203.0.113.45
```

Настройка политики IPSec с учетом GRE

```
set vpn ipsec site-to-site peer 192.0.2.10 tunnel 1 protocol
gre
```

Построение туннелей с интерфейсов Loopback

Описанная выше схема не работает, если один из маршрутизаторов имеет динамический внешний адрес. Классическим решением этой проблемы является задание адреса на интерфейсе Loopback и использование его в качестве адреса источника для GRE туннеля, а затем настройка политики IPsec для соответствия этим адрес на Loopback интерфейсах.

Предположим, что маршрутизатор *LEFT* имеет статический адрес *192.0.2.10* на интерфейсе *eth0*, а маршрутизатор *RIGHT* имеет динамический адрес на интерфейсе *eth0*.

Список команд для настройки маршрутизатора (LEFT):

```
set interfaces loopback lo address 192.168.99.1/32
set interfaces tunnel tun0 encapsulation gre
set interfaces tunnel tun0 address 10.10.10.1/30
set interfaces tunnel tun0 source-address 192.168.99.1
set interfaces tunnel tun0 remote 192.168.99.2
```

Список команд для настройки маршрутизатора (RIGHT):

```
set interfaces loopback lo address 192.168.99.2/32
set interfaces tunnel tun0 encapsulation gre
set interfaces tunnel tun0 address 10.10.10.2/30
set interfaces tunnel tun0 source-address 192.168.99.2
set interfaces tunnel tun0 remote 192.168.99.1
```

Далее необходимо выполнить настройку использования стека протоколов IPSec. Однако теперь необходимо обеспечить работу политики IPsec с динамическим адресом на стороне маршрутизатора *RIGHT*. Сложность состоит в том, что аутентификация с использованием общего секретного ключа (Pre-shared Secret) не работает с динамическим адресом, поэтому необходимо использовать ключ RSA.

Во-первых, на маршрутизаторе *LEFT* необходимо использовать команду generate vpn rsa-key bits 2048. Длину ключа можно выбрать отличную от представленной в примере.

```
fakel@left# run generate vpn rsa-key bits 2048
Generating rsa-key to /config/ipsec.d/rsa-keys/localhost.key
```

```
Your new local RSA key has been generated
The public portion of the key is:


0sAQO2335[long string here]
```

Затем - на маршрутизаторе *RIGHT* необходимо добавить сформированный ключ RSA в конфигурацию.

```
fakel@right# set vpn rsa-keys rsa-key-name LEFT rsa-key
KEYGOESHERE
```

Теперь можно выполнить настройку политики IPSec. Для этого теперь необходимо использовать идентификатор маршрутизатора *RIGHT* вместо адреса (динамического).

Список команд, которые необходимо выполнить на маршрутизаторе ***LEFT*** (статический адрес).

- set vpn rsa-keys rsa-key-name RIGHT rsa-key <PUBLIC KEY FROM THE RIGHT>
- set vpn ipsec ipsec-interfaces interface eth0
- set vpn ipsec esp-group MyESPGroup proposal 1 encryption aes128
- set vpn ipsec esp-group MyESPGroup proposal 1 hash sha1
- set vpn ipsec ike-group MyIKEGroup proposal 1 dh-group 2
- set vpn ipsec ike-group MyIKEGroup proposal 1 encryption aes128
- set vpn ipsec ike-group MyIKEGroup proposal 1 hash sha1
- set vpn ipsec site-to-site peer @RIGHT authentication mode rsa
- set vpn ipsec site-to-site peer @RIGHT authentication rsa-key-name RIGHT
- set vpn ipsec site-to-site peer @RIGHT default-esp-group MyESPGroup
- set vpn ipsec site-to-site peer @RIGHT ike-group MyIKEGroup
- set vpn ipsec site-to-site peer @RIGHT local-address 192.0.2.10
- set vpn ipsec site-to-site peer @RIGHT connection-type respond
- set vpn ipsec site-to-site peer @RIGHT tunnel 1 local prefix 192.168.99.1/32
- set vpn ipsec site-to-site peer @RIGHT tunnel 1 remote prefix 192.168.99.2/32

Список команд, которые необходимо выполнить на маршрутизаторе *RIGHT* (динамический адрес).

- set vpn rsa-keys rsa-key-name LEFT rsa-key <PUBLIC KEY FROM THE LEFT>
- set vpn ipsec ipsec-interfaces interface eth0
- set vpn ipsec esp-group MyESPGroup proposal 1 encryption aes128
- set vpn ipsec esp-group MyESPGroup proposal 1 hash sha1
- set vpn ipsec ike-group MyIKEGroup proposal 1 dh-group 2
- set vpn ipsec ike-group MyIKEGroup proposal 1 encryption aes128
- set vpn ipsec ike-group MyIKEGroup proposal 1 hash sha1
- set vpn ipsec site-to-site peer 192.0.2.10 authentication id @RIGHT
- set vpn ipsec site-to-site peer 192.0.2.10 authentication mode rsa
- set vpn ipsec site-to-site peer 192.0.2.10 authentication rsa-key-name LEFT
- set vpn ipsec site-to-site peer 192.0.2.10 authentication remote-id LEFT
- set vpn ipsec site-to-site peer 192.0.2.10 connection-type initiate
- set vpn ipsec site-to-site peer 192.0.2.10 default-esp-group MyESPGroup
- set vpn ipsec site-to-site peer 192.0.2.10 ike-group MyIKEGroup
- set vpn ipsec site-to-site peer 192.0.2.10 local-address any
- set vpn ipsec site-to-site peer 192.0.2.10 tunnel 1 local prefix 192.168.99.2/32
- set vpn ipsec site-to-site peer 192.0.2.10 tunnel 1 remote prefix 192.168.99.1/32

Настройки GRE туннеля

- set interfaces tunnel tunN encapsulation gre

- set interfaces tunnel tunN source-address <address>

- `set interfaces tunnel tunN remote <address>`

- `set interfaces tunnel tunN address <address>`

Настройка интерфейса IPSec

- `set vpn ipsec ipsec-interfaces interface <ethN>`

Настройка группы алгоритмов IKE

- `set vpn ipsec ike-group <text> proposal <165535> dh-group <number>`

- `set vpn ipsec ike-group <text> proposal <165535> encryption <type>`

- `set vpn ipsec ike-group <text> proposal <165535> hash <type>`

Группа настроек алгоритмов ESP

- `set vpn ipsec esp-group <text> proposal <165535> encryption <number>`

- `set vpn ipsec esp-group <text> proposal <165535> hash <number>`

Настройка туннеля IPSec

- `set vpn ipsec esp-group <text> proposal <165535> hash <number>`

- `set vpn ipsec site-to-site peer <address> authentication mode pre-shared-secret`

- `set vpn ipsec site-to-site peer <address> authentication pre-shared-secret <secret-key>`

- `set vpn ipsec site-to-site peer <address> ike-group <text>`

- `set vpn ipsec site-to-site peer <address> default-esp-group <text>`

- `set vpn ipsec site-to-site peer <address> local-address <x.x.x.x>`

Алгоритм RSA

Ключевая схема RSA может использоваться для таких сервисов, как обмен ключами и для организации шифрования. Для того чтобы стек протоколов IPSec работал с динамическими адресами с одной и другой стороны, необходимо использовать ключи RSA для аутентификации. Настройка данной ключевой схемы проста и осуществляется быстро.

Пример настройки RSA

Пример настройки RSA между двумя маршрутизаторами.

Сначала на обоих маршрутизаторах необходимо выполнить команду ***generate vpn rsa-key bits 2048*** в режиме Администрирования. Опционально можно выбрать длину ключа, отличную от значения по умолчанию - 2048.

```
fakel@left# run generate vpn rsa-key bits 2048
Generating rsa-key to /config/ipsec.d/rsa-keys/localhost.key

Your new local RSA key has been generated
The public portion of the key is:

0sAQO2335[long string here]
```

Обязательно скопируйте или запиши данный ключ RSA, так как его нужно будет на противоположный маршрутизатор.

- `set vpn rsa-keys rsa-key-name LEFT rsa-key KEYGOESHERE`

После подготовки RSA ключей, можно переходить к настройке IPSec.

Команды для настройки первого маршрутизатора:

```
set vpn rsa-keys rsa-key-name RIGHT rsa-key <PUBLIC KEY FROM THE RIGHT>
set vpn ipsec ipsec-interfaces interface eth0
set vpn ipsec nat-traversal 'enable'
set vpn ipsec esp-group MyESPGroup proposal 1 encryption aes128
set vpn ipsec esp-group MyESPGroup proposal 1 hash sha1
set vpn ipsec ike-group MyIKEGroup proposal 1 dh-group 2
set vpn ipsec ike-group MyIKEGroup proposal 1 encryption aes128
set vpn ipsec ike-group MyIKEGroup proposal 1 hash sha1
set vpn ipsec site-to-site peer 192.0.2.60 authentication mode rsa
set vpn ipsec site-to-site peer 192.0.2.60 authentication id @LEFT
set vpn ipsec site-to-site peer 192.0.2.60 authentication rsa-key-name RIGHT
set vpn ipsec site-to-site peer 192.0.2.60 authentication remote-id RIGHT
set vpn ipsec site-to-site peer 192.0.2.60 default-esp-group MyESPGroup
set vpn ipsec site-to-site peer 192.0.2.60 ike-group MyIKEGroup
set vpn ipsec site-to-site peer 192.0.2.60 local-address any
set vpn ipsec site-to-site peer 192.0.2.60 connection-type initiate
set vpn ipsec site-to-site peer 192.0.2.60 tunnel 1 local prefix 192.168.99.1/32
set vpn ipsec site-to-site peer 192.0.2.60 tunnel 1 remote prefix 192.168.99.2/32
```

Команды для настройки второго маршрутизатора:

```
set vpn rsa-keys rsa-key-name LEFT rsa-key <PUBLIC KEY FROM THE LEFT>
set vpn ipsec ipsec-interfaces interface eth0
set vpn ipsec nat-traversal 'enable'
```

- set vpn ipsec esp-group MyESPGroup proposal 1 encryption aes128
- set vpn ipsec esp-group MyESPGroup proposal 1 hash sha1
- set vpn ipsec ike-group MyIKEGroup proposal 1 dh-group 2
- set vpn ipsec ike-group MyIKEGroup proposal 1 encryption aes128
- set vpn ipsec ike-group MyIKEGroup proposal 1 hash sha1
- set vpn ipsec site-to-site peer @LEFT authentication id @RIGHT
- set vpn ipsec site-to-site peer @LEFT authentication mode rsa
- set vpn ipsec site-to-site peer @LEFT authentication rsa-key-name LEFT
- set vpn ipsec site-to-site peer @LEFT connection-type respond
- set vpn ipsec site-to-site peer @LEFT default-esp-group MyESPGroup
- set vpn ipsec site-to-site peer @LEFT ike-group MyIKEGroup
- set vpn ipsec site-to-site peer @LEFT local-address any
- set vpn ipsec site-to-site peer @LEFT tunnel 1 local prefix 192.168.99.2/32
- set vpn ipsec site-to-site peer @LEFT tunnel 1 remote prefix 192.168.99.1/32

Основные настройки алгоритма RSA

- set vpn rsa-keys rsa-key-name <name> rsa-key <text>

- set vpn ipsec ipsec-interfaces interface <ethN>

- set vpn ipsec nat-traversal <enable | disable>

- set vpn ipsec esp-group <text> proposal <1-65535> encryption <type>

- set vpn ipsec esp-group <text> proposal <1-65535> hash <type>

- `set vpn ipsec ike-group <text> proposal <1-65535> dh-group <number>`
- `set vpn ipsec ike-group <text> proposal <1-65535> encryption <type>`
- `set vpn ipsec ike-group <text> proposal <1-65535> hash <type>`
- `set vpn ipsec site-to-site peer <x.x.x.x> authentication mode <pre-shared-secret | rsa | x509>`
- `set vpn ipsec site-to-site peer <x.x.x.x> authentication id <test>`
- `set vpn ipsec site-to-site peer <x.x.x.x> authentication rsa-key-name <text>`
- `set vpn ipsec site-to-site peer <x.x.x.x> authentication remote-id <text>`
- `set vpn ipsec site-to-site peer <x.x.x.x> default-esp-group <text>`
- `set vpn ipsec site-to-site peer <x.x.x.x> ike-group <text>`
- `set vpn ipsec site-to-site peer <x.x.x.x> local-address <x.x.x.x>`

- `set vpn ipsec site-to-site peer <x.x.x.x> connection-type <initiate | respond>`

- `set vpn ipsec site-to-site peer <x.x.x.x> tunnel <0-4294967295> local prefix <x.x.x.x/x>`

- `set vpn ipsec site-to-site peer <x.x.x.x> tunnel <0-4294967295> remote prefix <x.x.x.x/x>`

Протокол L2TP

Общая информация о протоколе L2TP

Протокол L2TP (Layer 2 Tunneling Protocol) является комбинацией и развитием двух предыдущих протоколов: PPTP (Point-to-Point Tunneling Protocol) от Microsoft и L2F (Layer 2 Forwarding) от Cisco Systems. Этот протокол, определенный в RFC 2661, используется для поддержки виртуальных частных сетей (VPN) и позволяет передавать данные между двумя узлами через Интернет, обеспечивая при этом конфиденциальность и целостность передаваемой информации.

Основные характеристики и функции L2TP:

- **Создание туннеля**: L2TP работает на втором уровне OSI (канальный уровень), позволяя инкапсулировать пакеты одного протокола внутри протокола другого типа. Это означает, что L2TP может использоваться для туннелирования различных типов трафика, включая IP, IPX или NetBIOS через Интернет.
- **Совместимость с IPsec**: Часто L2TP используется в сочетании с IPsec (Internet Protocol Security). IPsec обеспечивает шифрование на уровне сетевого интерфейса, а L2TP – создание туннеля. Вместе они формируют L2TP/IPsec, предлагая механизмы аутентификации, конфиденциальности (шифрования) и целостности данных.
- **Многосессионность**: L2TP позволяет на одном физическом соединении одновременно обрабатывать несколько сессий. Это делает протокол гибким и эффективным решением для современных коммуникаций.
- **Аутентификация**: Протокол поддерживает различные механизмы аутентификации, включая пароли, цифровые сертификаты и аутентификацию на основе ключей.

Технические детали:

- **Инкапсуляция**: L2TP инкапсулирует данные путем размещения исходного кадра внутри L2TP-заголовка, который затем инкапсулируется в UDP-пакет (по умолчанию используется порт 1701) для транспортировки через IP-сеть.
- **Структура**: L2TP использует концепцию "туннелей" и "сессий". Туннель представляет собой управляющее соединение между L2TP-клиентом и L2TP-сервером. В рамках одного туннеля может быть установлено множество сессий, каждая из которых представляет собой отдельный поток данных.
- **Управляющие сообщения**: Для управления сессиями и туннелями L2TP использует управляющие сообщения, которые передаются по тому же каналу, что и данные, но без их инкапсуляции. Это позволяет эффективно управлять подключениями и адаптироваться к изменениям в сети.

L2TP сам по себе не предусматривает шифрование данных, что является его основным недостатком в контексте обеспечения безопасности данных. Однако, как было упомянуто, совместное использование с IPsec решает эту проблему, делая L2TP/IPsec популярным выбором для создания безопасных VPN-соединений.

Пример настройки протокола L2TP

IPSec с использованием L2TP

Пример настройки простого использования протокола L2TP поверх стека протоколов IPsec для защищенного удаленного доступа (VPN).

Список команд для настройки L2TP:

```
set vpn ipsec ipsec-interfaces interface eth0
set vpn ipsec nat-traversal enable
set vpn ipsec nat-networks allowed-network 0.0.0.0/0
set vpn l2tp remote-access outside-address 192.0.2.2
set vpn l2tp remote-access client-ip-pool start 192.168.255.2
set vpn l2tp remote-access client-ip-pool stop 192.168.255.254
set vpn l2tp remote-access ipsec-settings authentication mode
pre-shared-secret
set vpn l2tp remote-access ipsec-settings authentication pre-
shared-secret <secret>
set vpn l2tp remote-access authentication mode local
set vpn l2tp remote-access authentication local-users username
test password 'test'
```

В данном примере предполагается, что внешний адрес будет *192.0.2.2*.

Если на внешнем интерфейсе установлена локальная политика межсетевого экрана, то необходимо разрешить сетевой трафик по указанным ниже портам:

- Номер порта UDP 500 (для протокола IKE);
- Номер протокола IP 50 (для протокола ESP);
- Номер порта UDP 1701 (для стека протоколов IPsec);
- Номер порта UDP 4500 (для механизма NAT-Traversal).

Разрешение сетевого трафика по порту UDP 4500 необходимо для работы механизма NAT-Traversal (NAT-T), когда между VPN сервером и VPN клиентом находится шлюз, выполняющий трансляцию адресов (NAT). В этом случае сетевые пакеты ESP дополнительно упаковываются в UDP для обеспечения работы стека протоколов IPSec.

Пример настройки локальной политики межсетевого экрана:

```
set firewall name OUTSIDE-LOCAL rule 40 action 'accept'
set firewall name OUTSIDE-LOCAL rule 40 protocol 'esp'
set firewall name OUTSIDE-LOCAL rule 41 action 'accept'
set firewall name OUTSIDE-LOCAL rule 41 destination port '500'
set firewall name OUTSIDE-LOCAL rule 41 protocol 'udp'
set firewall name OUTSIDE-LOCAL rule 42 action 'accept'
set firewall name OUTSIDE-LOCAL rule 42 destination port '4500'
set firewall name OUTSIDE-LOCAL rule 42 protocol 'udp'
set firewall name OUTSIDE-LOCAL rule 43 action 'accept'
set firewall name OUTSIDE-LOCAL rule 43 destination port '1701'
set firewall name OUTSIDE-LOCAL rule 43 ipsec 'match-ipsec'
set firewall name OUTSIDE-LOCAL rule 43 protocol 'udp'
```

Для разрешения VPN клиенту подключаться через указанный внешний адрес также необходимо наличие правила трансляции (NAT).

Пример настройки правила трансляции:

```
set nat source rule 110 outbound-interface 'eth0'
set nat source rule 110 source address '192.168.255.0/24'
set nat source rule 110 translation address masquerade
```

VPN клиент обычно запрашивает параметры конфигурации, в числе которых может быть и список DNS серверов для клиента.

Пример настройки DNS сервера для VPN клиентов:

```
set vpn l2tp remote-access name-server '198.51.100.8'
set vpn l2tp remote-access name-server '198.51.100.4'
```

Список установленных соединений можно посмотреть с помощью команды **show vpn remote-access** или команды **show l2tp-server sessions** в режиме Администрирования.

Пример работы команды show vpn remote-access:

```
fakel@fakel:~$ show vpn remote-access
ifname | username | calling-sid  |      ip       | rate-limit | type | comp | state  |  uptime
-------+----------+--------------+---------------+------------+------+------+--------+----------

ppp0   | vyos     | 192.168.0.36 | 192.168.255.1 |            | l2tp |      | active | 00:06:13
```

Пример настройки сервера доступа LNS (L2TP Network Server)

Сервер доступа LNS зачастую используется для подключения к серверу LAC (L2TP Access Concentrator).

Пример настройки сервера доступа LNS:

```
set vpn l2tp remote-access outside-address 192.0.2.2
set vpn l2tp remote-access client-ip-pool start 192.168.255.2
set vpn l2tp remote-access client-ip-pool stop 192.168.255.254
set vpn l2tp remote-access lns shared-secret 'secret'
set vpn l2tp remote-access ccp-disable
set vpn l2tp remote-access authentication mode local
set vpn l2tp remote-access authentication local-users username
test password 'test'
```

В приведенном примере в качестве внешнего IP-адреса используется 192.0.2.2. Для сервера LAC обычно требуется пароль аутентификации, который в приведенной в примере конфигурации задан как lns shared-secret 'secret'. Данная настройка требует отключения протокола управления сжатием (Compression Control Protocol, CCP), что достигается командой set vpn l2tp remote-access ccp-disable.

Пример настройки ограничения полосы пропускания

Лимиты пропускной способности могут быть установлены как для локальных пользователей, так и для пользователей RADIUS с помощью специальных атрибутов.

Пример настройки ограничений полосы пропускания для локальных пользователей:

- `set vpn l2tp remote-access outside-address 192.0.2.2`
- `set vpn l2tp remote-access client-ip-pool start 192.168.255.2`
- `set vpn l2tp remote-access client-ip-pool stop 192.168.255.254`
- `set vpn l2tp remote-access authentication mode local`
- `set vpn l2tp remote-access authentication local-users username test password test`
- `set vpn l2tp remote-access authentication local-users username test rate-limit download 20480`
- `set vpn l2tp remote-access authentication local-users username test rate-limit upload 10240`

Лимиты пропускной способности устанавливаются в единицах кбит/с (килобит в секунду).

Настроенные лимиты можно посмотреть с помощью команды ***show vpn remote-access*** в режиме Администрирования.

Пример работы команды show vpn remote-access:

```
fakel@fakel:~$ show vpn remote-access
ifname | username | calling-sid  |      ip       | rate-limit  | type | comp | state  |  uptime
-------+----------+--------------+---------------+-------------+------+------+--------+----------

ppp0   | test     | 192.168.0.36 | 192.168.255.2 | 20480/10240 | l2tp |      | active | 00:06:30
```

Основные настройки протокола L2TP

- `set vpn ipsec ipsec-interfaces interface <ethN>`

- `set vpn ipsec nat-traversal enable`

- `set vpn ipsec nat-networks allowed-network <x.x.x.x/x>`

- `set vpn l2tp remote-access outside-address <x.x.x.x>`

- `set vpn l2tp remote-access client-ip-pool start <x.x.x.x>`

- `set vpn l2tp remote-access client-ip-pool stop <x.x.x.x>`

- `set vpn l2tp remote-access ipsec-settings authentication mode pre-shared-secret`

- `set vpn l2tp remote-access ipsec-settings authentication pre-shared-secret <secret>`

- `set vpn l2tp remote-access authentication mode local`

- `set vpn l2tp remote-access authentication local-users username <text-name> password <text-password>`

- `set vpn l2tp remote-access name-server '198.51.100.8'`

- `set vpn l2tp remote-access lns shared-secret 'secret'`

- `set vpn l2tp remote-access ccp-disable`

- `set vpn l2tp remote-access authentication local-users username test rate-limit download 20480`

- `set vpn l2tp remote-access authentication local-users username test rate-limit upload 10240`

```
show vpn remote-access
```

Аутентификация RADIUS

Чтобы использовать аутентификацию на основе протокола RADIUS, необходимо изменить режим аутентификации в конфигурации. Предыдущие настройки, например, локальные пользователи, по-прежнему остаются в конфигурации, однако они не используются, если режим был изменен с локального на RADIUS. После изменения режима на локальный все локальные учетные записи будут снова использоваться.

```
set vpn l2tp remote-access authentication mode <local|radius>
```

Поскольку одиночный RADIUS сервер будет являться единой точкой отказа в случае выхода из строя, можно настроить несколько RADIUS серверов, которые будут последовательно опрашиваться.

```
set vpn l2tp remote-access authentication radius server
10.0.0.1 key 'foo'
set vpn l2tp remote-access authentication radius server
10.0.0.2 key 'foo'
```

Примечание

Некоторые RADIUS сервера используют список контроля доступа, который разрешает запросы от определенных RADIUS клиентов. Убедитесь, что ваш маршрутизатор с операционной системой Факел добавлен в список разрешенных RADIUS клиентов.

Адрес отправителя

Если в качестве внутреннего протокола маршрутизации (IGP) используется протокол OSPF, то всегда используется ближайший интерфейс, подключенный к RADIUS серверу. В операционной системе Факел можно привязать все исходящие RADIUS запросы к одному IP-адресу источника, например, к интерфейсу Dummy.

```
set vpn l2tp remote-access authentication radius source-address
10.0.0.3
```

Приведенная выше команда будет использовать адрес *10.0.0.3* в качестве исходного IP-адреса для всех RADIUS запросов на этом устройстве.

Примечание

Параметр source-address должен быть ассоциирован с одним из интерфейсов операционной системы Факел. Рекомендуется использовать интерфейс Dummy.

Ограничение полосы пропускания

Для ограничения полосы пропускания в части пользователей RADIUS необходимо использовать параметр rate-limit.

```
set vpn l2tp remote-access authentication radius rate-limit enable
```

Для ограничения полосы пропускания в протоколе RADIUS предусмотрен параметр Filter-Id, значение которого можно переопределять.

```
set vpn l2tp remote-access authentication radius rate-limit attribute Download-Speed
```

> **Примечание**
>
> *При задании пользовательских параметров в протоколе RADIUS необходимо объявить их в специальных словарях как стороне RADIUS сервера, так и на стороне RADIUS клиента, которым в нашем примере является маршрутизатор с операционной системой Факел. Словари RADIUS в операционной системе Факел располагаются по следующему пути: /usr/share/accel-ppp/radius/.*

Расширенные функции RADIUS

Значения параметров протокола RADIUS, полученных от сервера, имеют более высокий приоритет по сравнению со значениями, заданными для тех же параметров в конфигурации через **Командная строка**.

Распределение IP-адресов между клиентами

Если RADIUS сервер возвращает параметр Framed-IP-Address, то IP-адрес в качестве его значения будет выделен клиенту, а параметр ip-pool в конфигурации CLI игнорируется.

Переименование интерфейсов клиентов

Если RADIUS сервер возвращает параметр NAS-Port-Id, то в результате туннельные интерфейсы PPP будут переименованы.

> **Примечание**
>
> *Значение параметра NAS-Port-Id должно быть длиной не более 16 символов. В противном случае интерфейс не будет переименован.*

Основные настройки аутентификации RADIUS

```
set vpn l2tp remote-access authentication mode <local|radius>
```

- `set vpn l2tp remote-access authentication radius server <x.x.x.x> key <text>`

- `set vpn l2tp remote-access authentication radius source-address <x.x.x.x>`

- `set vpn l2tp remote-access authentication radius rate-limit enable`

- `set vpn l2tp remote-access authentication radius rate-limit attribute Download-Speed`

Сервер OpenConnect

Общая информация о сервере OpenConnect

Операционная система Факел поддерживает работу маршрутизатора в режиме OpenConnect VPN сервера. Данный сервер позволяет установить с маршрутизатором SSL/TLS соединение и обеспечить полный доступ к защищаемой (корпоративной) сети. Такой подход к организации VPN объединяет конечных пользователей и ресурсы в защищаемой (корпоративной) сети в виртуальную сеть с контролем доступа, основанным только на информации сетевого уровня, такой как IP-адрес отправителя и номер порта. Таким образом достигается необходимая степень защиты всех типов трафика для устройств между открытыми и защищаемыми (корпоративными) сетями, а также шифрование с помощью протоколов SSL/TLS.

Удаленный пользователь с помощью VPN клиента с поддержкой OpenConnect подключается к маршрутизатору и получает IP-адрес из заданного пула адресов, получая доступ к ресурсам защищаемой (корпоративной) сети.

> **Примечание**
>
> *Все сертификаты должны размещаться в операционной системе Факел по следующему пути /config/auth. Если сертификаты размещаются не в каталоге /config, они не будут перенесены при обновлении ПО.*

Пример настройки сервера OpenConnect

В первую очередь необходимо сформировать сертификат, который будет удостоверять подлинность пользователей, пытающихся получить доступ к ресурсам защищаемой

(корпоративной) сети через SSL/TLS туннели. Следующая команда позволяет создать сертификаты, выпущенные внутренним центром сертификации, которые будут размещены по следующему пути */config/auth*.

```
openssl req -newkey rsa:4096 -new -nodes -x509 -days 3650 -keyout /config/auth/server.key -out /config/auth/server.crt
openssl req -new -x509 -key /config/auth/server.key -out /config/auth/ca.crt
```

Можно также создать сертификат с помощью инструмента certbot - простого в использовании клиента, который получает сертификат из - Let's Encrypt - открытого центра - сертификации, созданного консорциумом организаций, и размещает его на веб-сервере.

```
sudo certbot certonly --standalone --preferred-challenges http -d <domain name>
```

После того, как сертификат подготовлен, можно переходить к настройке сервера OpenConnect:

- Используется локальный пользователь с логином *user4* и паролем *SecretPassword*.
- VPN клиенту выделяется IP-адрес из пула *100.64.0.0/24*.
- IP-адрес шлюза должен быть адресом одного из интерфейсов маршрутизатора.

```
set vpn openconnect authentication local-users username user4 password 'SecretPassword'
set vpn openconnect authentication mode 'local'
set vpn openconnect network-settings client-ip-settings subnet '100.64.0.0/24'
set vpn openconnect network-settings name-server '10.1.1.1'
set vpn openconnect network-settings name-server '10.1.1.2'
set vpn openconnect ssl ca-cert-file '/config/auth/fullchain.pem'
set vpn openconnect ssl cert-file '/config/auth/cert.pem'
set vpn openconnect ssl key-file '/config/auth/privkey.pem'
```

После применения изменений, внесенных в конфигурацию, необходимо убедиться в том, что подключение VPN клиента выполнено успешно.

```
fakel@fakel:~$ show openconnect-server sessions
```

```
interface    username    ip            remote IP      RX        TX        state      uptime
-----------  ----------  ------------  -------------  --------  --------  ---------  --------
sslvpn0      user4       100.64.0.105  xx.xxx.49.253  127.3 KB  160.0 KB  connected  12m:28s
```

Примечание

В качестве совместимого VPN клиента может использоваться клиент Cisco (R) AnyConnect (R).

Основные настройки сервера OpenConnect

- `set vpn openconnect authentication local-users username <user> password <pass>`
- `set vpn openconnect authentication mode <local|radius>`
- `set vpn openconnect network-settings client-ip-settings subnet <subnet>`
- `set vpn openconnect network-settings name-server <address>`
- `set vpn openconnect network-settings name-server <address>`
- `set vpn openconnect ssl ca-cert-file <file>`
- `set vpn openconnect ssl cert-file <file>`
- `set vpn openconnect ssl key-file <file>`

Сервер PPTP

Общая информация о сервере OpenConnect

Протокол PPTP реализован в ПО **Факел** только для обеспечения обратной совместимости. PPTP имеет множество известных проблем с безопасностью, и вместо него необходимо использовать одну из многих других современных реализаций VPN.

По умолчанию если не задана иная конфигурация, то для аутентификации используется протокол MS-CHAP v2, а для шифрования - протокол MPPE с разрядностью 128 бит и без контроля соединений. Если в конфигурации не задан адрес шлюза, то используется наименьший IP-адрес из пула адресов клиента, заданного подсетью с маской /24. Например, в приведенном ниже примере это будет IP-адрес 192.168.0.1.

Пример настройки сервера PPTP

Пример настройки на стороне сервера

```
set vpn pptp remote-access authentication local-users username test password 'test'
set vpn pptp remote-access authentication mode 'local'
set vpn pptp remote-access client-ip-pool start '192.168.0.10'
set vpn pptp remote-access client-ip-pool stop '192.168.0.15'
set vpn pptp remote-access gateway-address '10.100.100.1'
set vpn pptp remote-access outside-address '10.1.1.120'
```

Пример настройки на стороне клиента

В данному примере рассматривается настройка PPTP клиента в операционной среде Debian 9.

Выполните установку клиентского ПО с помощью менеджера пакетов apt и выполните команду pptpsetup, чтобы сформировать начальную конфигурацию.

```
apt-get install pptp-linux
pptpsetup --create TESTTUNNEL --server 10.1.1.120 --username test --password test –encrypt
pon TESTTUNNEL
```

Команда TESTTUNNEL позволяет установить PPTP туннель с удаленным хостом (PPTP сервером).

Текущее состояние сессий (туннелей) можно вывести на экран следующим образом.

```
run sh pptp-server sessions
ifname | username | calling-sid |      ip      | type | comp | state  |  uptime
```

```
------+---------+-------------+-------------+------+------+--------+----------
ppp0   | test     | 10.1.1.99   | 192.168.0.10 | pptp | mppe | active | 00:00:58
```

Основные настройки сервера PPTP

- `set vpn pptp remote-access authentication local-users username <name> password <text-password>`

- `set vpn pptp remote-access authentication mode <local | radius>`

- `set vpn pptp remote-access client-ip-pool start <x.x.x.x>`

- `set vpn pptp remote-access client-ip-pool stop <x.x.x.x>`

- `set vpn pptp remote-access gateway-address <x.x.x.x>`

- `set vpn pptp remote-access outside-address <x.x.x.x>`

Сервер SSTP

Общая информация о сервере SSTP

Протокол SSTP - протокол построения VPN, обеспечивающий механизм передачи PPP трафика по SSL/TLS туннелю. SSL/TLS обеспечивает безопасность на транспортном уровне с согласованием ключей, шифрованием и проверкой целостности трафика. Использование SSL/TLS через TCP-порт 443 позволяет протоколу SSTP проходить практически через любой межсетевой экран и прокси-сервер, за исключением веб-прокси с аутентификацией.

Протокол SSTP доступен к использованию в операционных системах Linux, BSD и Windows.

Для обеспечения функциональности SSTP сервера в операционной системе Факел используется механизм accel-ppp. Данный механизм может использоваться как с локальной аутентификацией, так и совместно с подключенным сервером RADIUS.

Поскольку протокол SSTP позволяет передавать PPP трафик по SSL/TLS туннелю, для организации VPN необходимо использовать сертификаты с открытой подписью, а также инфраструктуру PKI.

> **Примечание**
>
> *Все сертификаты должны размещаться в операционной системе Факел по следующему пути /config/auth. Если сертификаты размещаются не в каталоге /config, они не будут перенесены при обновлении ПО.*

Пример настройки сервера SSTP

В данном примере рассматриваться порядок конфигурирования SSTP сервера в составе операционной системы Факел с учетом следующих особенностей:

- используется локальная учетная запись пользователя с логином *foo* и паролем *bar*;
- IP-адрес для клиента будет выделяться из пула, определенного подсетью *192.0.2.0/25*.

Подготовка сертификатов

Первым шагом необходимо подготовить сертификата УЦ, закрытого ключа и сертификата сервера SSTP.

Для генерации сертификата УЦ, закрытого ключа и сертификата сервера SSTP выполните следующие команды:

```
fakel@fakel:~$ mkdir -p /config/user-data/sstp
fakel@fakel:~$ openssl req -newkey rsa:4096 -new -nodes -x509 -days 3650 -keyout /config/user-data/sstp/server.key -out /config/user-data/sstp/server.crt


Generating a 4096 bit RSA private key
.........................++
.....................................................................++
Writing new private key to 'server.key'
[...]
Country Name (2 letter code) [RU]:
State or Province Name (full name) [Some-Region]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Some Company Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:
```

```
fakel@fakel:~$ openssl req -new -x509 -key /config/user-
data/sstp/server.key -out /config/user-data/sstp/ca.crt
[...]
Country Name (2 letter code) [RU]:
State or Province Name (full name) [Some-Region]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Some Company Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:
```

Настройка SSTP на маршрутизаторе

После того, как сертификаты будут готовы, можно переходить к настройке SSTP.

Команды для настройки SSTP на маршрутизаторе:

```
set vpn sstp authentication local-users username foo password
bar
set vpn sstp authentication mode local
set vpn sstp gateway-address 192.0.2.254
set vpn sstp client-ip-pool subnet 192.0.2.0/25
set vpn sstp name-server 10.0.0.1
set vpn sstp name-server 10.0.0.2
set vpn sstp ssl ca-cert-file /config/auth/ca.crt
set vpn sstp ssl cert-file /config/auth/server.crt
set vpn sstp ssl key-file /config/auth/server.key
```

Проверка работы SSTP

После конфигурирования SSTP сервера необходимо выполнить базовую проверку его работы. Для этого можно воспользоваться SSTP клиентом для операционной системы Linux - *sstpc*. Данные клиент потребует наличия соответствующего конфигурационного файла.

В примере далее показано использование для базовой проверки протокола MS-CHAP v2.

```
fakel@fakel:~$ cat /etc/ppp/peers/fakel
usepeerdns
#require-mppe
```

```
#require-pap
require-mschap-v2
noauth
lock
refuse-pap
refuse-eap
refuse-chap
refuse-mschap
#refuse-mschap-v2
nobsdcomp
nodeflate
debug
```

Теперь можно выполнить подключение клиента с помощью команды ***sstpc --log-level 4 --log-stderr --user foo --password bar vpn.example.ru -- call fakel***. Детальная информация по попытке подключения представлена в примере ниже.

```
fakel@fakel:~$ sstpc --log-level 4 --log-stderr --user foo --
password bar vpn.example.ru -- call fakel
Mar 22 13:29:12 sstpc[12344]: Resolved vpn.example.ru to 192.0.2.1
Mar 22 13:29:12 sstpc[12344]: Connected to vpn.example.ru
Mar 22 13:29:12 sstpc[12344]: Sending Connect-Request Message
Mar 22 13:29:12 sstpc[12344]: SEND SSTP CRTL PKT(14)
Mar 22 13:29:12 sstpc[12344]:    TYPE(1): CONNECT REQUEST,
ATTR(1):
Mar 22 13:29:12 sstpc[12344]:       ENCAP PROTO(1): 6
Mar 22 13:29:12 sstpc[12344]: RECV SSTP CRTL PKT(48)
Mar 22 13:29:12 sstpc[12344]:    TYPE(2): CONNECT ACK, ATTR(1):
Mar 22 13:29:12 sstpc[12344]:       CRYPTO BIND REQ(4): 40
Mar 22 13:29:12 sstpc[12344]: Started PPP Link Negotiation
Mar 22 13:29:15 sstpc[12344]: Sending Connected Message
Mar 22 13:29:15 sstpc[12344]: SEND SSTP CRTL PKT(112)
Mar 22 13:29:15 sstpc[12344]:    TYPE(4): CONNECTED, ATTR(1):
Mar 22 13:29:15 sstpc[12344]:       CRYPTO BIND(3): 104
Mar 22 13:29:15 sstpc[12344]: Connection Established


fakel@fakel:~$ ip addr show ppp0
164: ppp0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1452
qdisc fq_codel state UNKNOWN group default qlen 3
```

```
    link/ppp  promiscuity 0
    inet 100.64.2.2 peer 100.64.1.1/32 scope global ppp0
       valid_lft forever preferred_lft forever
```

Основные настройки сервера SSTP

- `set vpn sstp authentication local-users username <user> password <pass>`

Создает учетную запись пользователя с логином *<user>* для локальной аутентификации. Пароль пользователя будет иметь значение *<pass>*.

- `set vpn sstp authentication local-users username <user> disable`

Отключает учетную запись пользователя с логином *<user>*.

- `set vpn sstp authentication local-users username <user> static-ip <address>`

Назначает учетной записи пользователя с логином *<user>* статический IP-адрес *<address>*.

- `set vpn sstp authentication local-users username <user> rate-limit download <bandwidth>`

Ограничивает полосу пропускания при скачивании данных для учетной записи пользователя с логином *<user>* в единицах кбит/с (килобит в секунду).

- `set vpn sstp authentication local-users username <user> rate-limit upload <bandwidth>`

Ограничивает полосу пропускания при загрузке данных для учетной записи пользователя с логином *<user>* в единицах кбит/с (килобит в секунду).

- `set vpn sstp authentication protocols <pap|chap|mschap|mschap-v2>`

Задает требования аутентификации узла-участника VPN с помощью одного из следующих протоколов: PAP, CHAP, MS-CHAP, MS-CHAP v2.

- `set vpn sstp authentication mode <local|radius>`

Задает режим аутентификации *(<local>* - локальных пользователей, *<radius>* - пользователей RADIUS). В заданном режиме будут обрабатываться все запросы.

- *<local>* - все запросы на аутентификацию будут обрабатываться локально.

- *<radius>* - все запросы на аутентификацию будут обрабатываться сконфигурированным RADIUS сервером.

```
set vpn sstp gateway-address <gateway>
```

Задает IP-адрес узла *<gateway>*, который будет использоваться в качестве локального адреса для PPP интерфейса.

```
set vpn sstp client-ip-pool subnet <subnet>
```

Задает IP-адрес подсети *<subnet>*, которая будет использоваться в качестве пула адресов для всех подключающихся клиентов.

```
set vpn sstp client-ipv6-pool prefix <address> mask <number-of-bits>
```

Задает пул IPv6, из которого SSTP клиент будет получать IPv6 префикс заданной длины (маску) для определения конечной точки SSTP на своей стороне. Длина маски может быть задана в диапазоне от 48 до 128 бит, значение по умолчанию - 64 бит.

```
set vpn sstp client-ipv6-pool delegate <address> delegation-prefix <number-of-bits>
```

Настраивает механизма делегирования префикса DHCPv6 согласно RFC 3633 в контексте протокола SSTP. Для этого необходимо задать пул IPv6 и длину префикса делегирования. Из заданного пула будут выделяться IPv6 подсети определенной размерности. Длина префикса делегирования может быть задана от 32 до 64 бит.

```
set vpn sstp name-server <address>
```

При использовании данной команды подключенный клиент должен использовать адрес **<address>** в качестве адреса DNS сервера. Эта команда принимает адреса как IPv4, так и IPv6. Для IPv4 может быть настроено до двух DNS серверов, для IPv6 - до трех.

Настройка сертификатов SSL/TLS для сервера SSTP

```
set vpn sstp ssl ca-cert-file <file>
```

Задает путь до файла *<file>* сертификата УЦ.

```
set vpn sstp ssl cert-file <file>
```

Задает путь до файла *<file>* сертификата сервера (открытый ключ).

```
set vpn sstp ssl key-file <file>
```

Задает путь до файла *<file>* с закрытым ключом сервера.

Настройка PPP для сервера SSTP

```
set vpn sstp ppp-options lcp-echo-failure <number>
```

Задает максимальное количество *<number>* сообщений Echo Request по протоколу LCP, по достижении которого PPP сессия будет сброшена.

```
set vpn sstp ppp-options lcp-echo-interval <interval>
```

Задает периодичность *<interval>* в секундах отправки модулем PPP сообщения Echo Request по протоколу LCP. При значении 0 модуль PPP не будет отправлять сообщения Echo Request.

```
set vpn sstp ppp-options lcp-echo-timeout <timeout>
```

Задает время *<timeout>* ожидания активности любых узлов-участников VPN. При использовании опции lcp-echo-timeout для модуля PPP активируется режим адаптивной функциональности LCP Echo Request, а значение опции lcp-echo-failure игнорируется.

```
set vpn sstp ppp-options mppe <require|prefer|deny>
```

Определяет один из следующих режимов использования протокола шифрования MPPE:

- **require** - ожидать от клиента подтверждения использования протокола MPPE, если клиент такого подтверждения не дает, то соединение сбрасывается;
- **prefer** - ожидать от клиента подтверждения использования протокола MPPE, если клиент такого подтверждения не дает, то соединение сохраняется;
- **deny** - не использовать протокол MPPE.

Если режим не указан явно, то по умолчанию модуль PPP не запрашивает подтверждения от клиента, но допускает использование протокола MPPE по его инициативе.

> **Примечание**
>
> *Обратите внимание, что RADIUS сервер может переопределять значение данной опции значением атрибута MS-MPPE-Encryption-Policy.*

Настройка аутентификации RADIUS для SSTP сервера

```
set vpn sstp authentication radius server <server> port <port>
```

Определяет RADIUS сервер *<server>* и задает порт *<port>* для отправки запросов на аутентификацию.

- `set vpn sstp authentication radius server <server> key <secret>`

Определяет RADIUS сервер <*server*> и задает общий ключ <*secret*> для защиты коммуникаций с сервером.

- `set vpn sstp authentication radius server <server> fail-time <time>`

Помечает настроенный RADIUS сервер <*server*> как «не в сети» на заданный в секундах промежуток времени <*time*>.

- `set vpn sstp authentication radius server <server> disable`

Отключает настроенный RADIUS сервер <*server*>.

- `set vpn sstp authentication radius acct-timeout <timeout>`

Определяет время ожидания <*timeout*> в секундах ответа на пакеты Interim Update от RADIUS сервера. По умолчанию - 3 секунды.

- `set vpn sstp authentication radius dynamic-author server <address>`

Задает IP-адрес <*address*> для сервера Dynamic Authorization Extension (DM/CoA).

- `set vpn sstp authentication radius dynamic-author port <port>`

Задает порт <*port*> для подключения к серверу Dynamic Authorization Extension (DM/CoA).

- `set vpn sstp authentication radius dynamic-author key <secret>`

Задает общий ключ <*secret*> для защиты коммуникаций с сервером Dynamic Authorization Extension (DM/CoA).

- `set vpn sstp authentication radius max-try <number>`

Определяет количество <*number*> попыток передачи серверу запросов Access Request и Accounting Request.

- `set vpn sstp authentication radius timeout <timeout>`

Определяет время ожидания <*timeout*> в секундах ответа от сервера.

- `set vpn sstp authentication radius nas-identifier <identifier>`

Задает значение идентификатора NAS-Identifier, который будет передан RADIUS серверу соответствующим атрибутом и должен соответствовать запросам Dynamic Authorization Extension (DM/CoA).

- `set vpn sstp authentication radius nas-ip-address <address>`

Определяет значение адреса NAS-IP-Address, который будет передан RADIUS серверу соответствующим атрибутом и должен соответствовать запросам Dynamic Authorization Extension (DM/CoA). Также указанный адрес будет выбран в качестве адреса сервера Dynamic Authorization Extension (DM/CoA).

- `set vpn sstp authentication radius source-address <address>`

Задает IPv4 адрес *<address>* отправителя, который будет использоваться при формировании запросов на аутентификацию к RADIUS серверу.

- `set vpn sstp authentication radius rate-limit attribute <attribute>`

Определяет в каком из атрибутов *<attribute>* RADIUS сервера будет храниться предельное значение полосы пропускания. По умолчанию таким атрибутом является атрибут Filter-Id.

- `set vpn sstp authentication radius rate-limit enable`

Включает ограничение полосы пропускания средствами RADIUS сервера.

- `set vpn sstp authentication radius rate-limit vendor`

Включает использование справочника атрибутов RADIUS сервера, который должен располагаться по следующему пути */usr/share/accel-ppp/radius*.

DMVPN

Общая информация о DMVPN

DMVPN - технология динамического построения VPN, первоначально разработанная компанией Cisco. Хотя ее реализация и была в некоторой степени закрытой, лежащие в ее основе технологии в действительности основаны на вполне конкретных стандартах. В основе данной технологии лежит использование следующих протоколов:

- **NHRP** - RFC 2332;
- **mGRE** - RFC 1702;
- **IPSec** - RFC 4301.

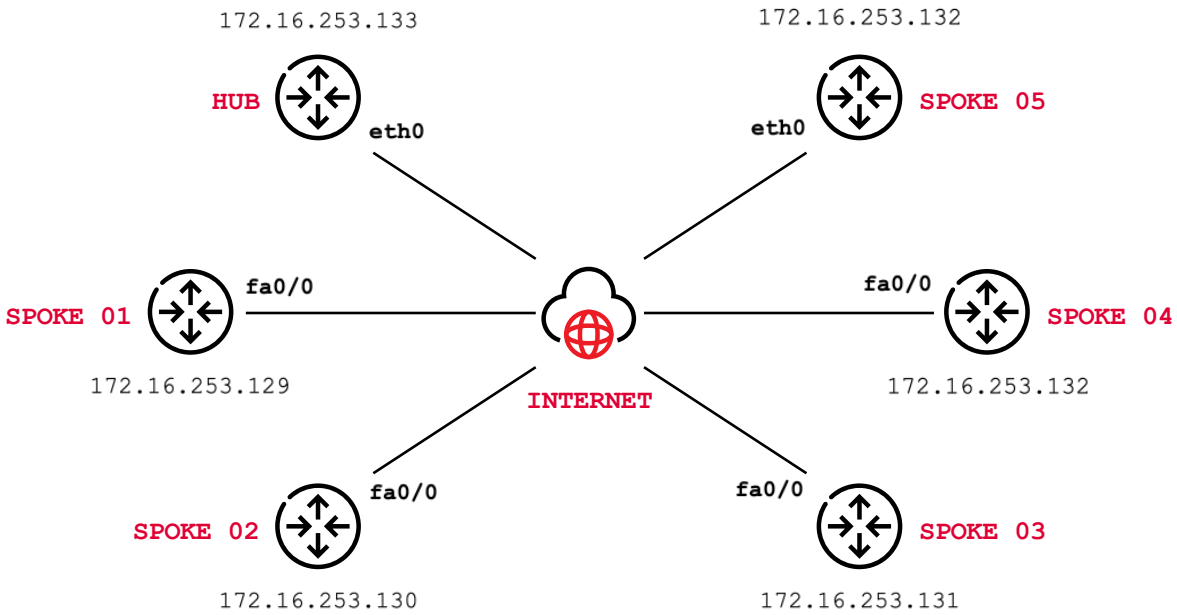
Протокол NHRP обеспечивает механизм динамического обнаружения конечных точек туннеля (поиск и их регистрация), протокол mGRE обеспечивает непосредственно инкапсуляцию данных, передаваемых по туннелю, а стек протоколов IPSec обеспечивает обмен ключами и криптографическую защиту передаваемых данных.

Другими словами, DMVPN обеспечивает возможность создания динамически связной VPN сети без предварительной (статической) настройки всех возможных конечных точек туннеля.

> **Примечание**
>
> *Технология DMVPN автоматизирует только процесс обнаружения и настройки конечных точек туннеля. Полноценное решение также включает в себя использование протокола маршрутизации. Особенно хорошо для использования в данной технологии подходит протокол динамической маршрутизации BGP.*

Пример настройки DMVPN

Данный пример подразумевает использование устройства под управлением операционной системы Факел в качестве центрального узла (*HUB*), устройства под управлением операционной системы Факел в качестве одного из оконечных узлов (*SPOKE 05*), а также устройств под управлением операционной системы Cisco IOS в качестве остальных оконечных узлов (*SPOKE 01–04*).



Каждый узел (*Hub, Spoke 01 - 05*) использует IP-адрес из подсети *172.16.253.128/29*.

Используемый в конфигурации ниже IP-адрес *192.0.2.1*, являющийся глобальным адресом, по которому доступен центральный узел (*HUB*) для каждого из оконечных узлов (*SPOKE*).

Команды для настройки Центрального узла «HUB»:

```
set interfaces ethernet eth0 address 192.0.2.1/24
set interfaces tunnel tun100 address '172.16.253.134/29'
set interfaces tunnel tun100 encapsulation 'gre'
set interfaces tunnel tun100 source-address '192.0.2.1'
set interfaces tunnel tun100 multicast 'enable'
set interfaces tunnel tun100 parameters ip key '1'
set protocols nhrp tunnel tun100 cisco-authentication 'secret'
set protocols nhrp tunnel tun100 holding-time '300'
set protocols nhrp tunnel tun100 multicast 'dynamic'
set protocols nhrp tunnel tun100 redirect
set protocols nhrp tunnel tun100 shortcut
set vpn ipsec esp-group ESP-HUB compression 'disable'
set vpn ipsec esp-group ESP-HUB lifetime '1800'
set vpn ipsec esp-group ESP-HUB mode 'transport'
set vpn ipsec esp-group ESP-HUB pfs 'dh-group2'
set vpn ipsec esp-group ESP-HUB proposal 1 encryption 'aes256'
set vpn ipsec esp-group ESP-HUB proposal 1 hash 'sha1'
set vpn ipsec esp-group ESP-HUB proposal 2 encryption '3des'
set vpn ipsec esp-group ESP-HUB proposal 2 hash 'md5'
set vpn ipsec ike-group IKE-HUB ikev2-reauth 'no'
set vpn ipsec ike-group IKE-HUB key-exchange 'ikev1'
set vpn ipsec ike-group IKE-HUB lifetime '3600'
set vpn ipsec ike-group IKE-HUB proposal 1 dh-group '2'
set vpn ipsec ike-group IKE-HUB proposal 1 encryption 'aes256'
set vpn ipsec ike-group IKE-HUB proposal 1 hash 'sha1'
set vpn ipsec ike-group IKE-HUB proposal 2 dh-group '2'
set vpn ipsec ike-group IKE-HUB proposal 2 encryption 'aes128'
set vpn ipsec ike-group IKE-HUB proposal 2 hash 'sha1'
set vpn ipsec ipsec-interfaces interface 'eth0'
set vpn ipsec profile NHRPVPN authentication mode 'pre-shared-
secret'
set vpn ipsec profile NHRPVPN authentication pre-shared-secret
'secret'
set vpn ipsec profile NHRPVPN bind tunnel 'tun100'
```

- set vpn ipsec profile NHRPVPN esp-group 'ESP-HUB'
- set vpn ipsec profile NHRPVPN ike-group 'IKE-HUB'

Команды для настройки оконечных узлов «SPOKE 01-04»:

```
crypto keyring DMVPN
   pre-shared-key address 192.0.2.1 key secret
!
crypto isakmp policy 10
   encr aes 256
   authentication pre-share
   group 2
crypto isakmp invalid-spi-recovery
crypto isakmp keepalive 30 30 periodic
crypto isakmp profile DMVPN
   keyring DMVPN
   match identity address 192.0.2.1 255.255.255.255
!
crypto ipsec transform-set DMVPN-AES256 esp-aes 256 esp-sha-hmac
   mode transport
!
crypto ipsec profile DMVPN
   set security-association idle-time 720
   set transform-set DMVPN-AES256
   set isakmp-profile DMVPN
!
interface Tunnel10
   ! individual spoke tunnel IP must change
   ip address 172.16.253.129 255.255.255.248
   no ip redirects
   ip nhrp authentication secret
   ip nhrp map 172.16.253.134 192.0.2.1
   ip nhrp map multicast 192.0.2.1
   ip nhrp network-id 1
   ip nhrp holdtime 600
   ip nhrp nhs 172.16.253.134
   ip nhrp registration timeout 75
   tunnel source FastEthernet0/0
```

```
   tunnel mode gre multipoint
   tunnel protection ipsec profile DMVPN
   tunnel key 1
!
interface FastEthernet0/0
   ip address dhcp
   duplex half
```

Примечание

Конфигурации отдельных оконечных узлов будут отличаться только локальным IP-адресом на интерфейсе tun10.

Команды для настройки оконечного узла «SPOKE 05»:

```
set interfaces ethernet eth0 address 'dhcp'
set interfaces tunnel tun100 address '172.16.253.133/29'
set interfaces tunnel tun100 source-address 0.0.0.0
set interfaces tunnel tun100 encapsulation 'gre'
set interfaces tunnel tun100 multicast 'enable'
set interfaces tunnel tun100 parameters ip key '1'
set protocols nhrp tunnel tun100 cisco-authentication 'secret'
set protocols nhrp tunnel tun100 holding-time '300'
set protocols nhrp tunnel tun100 map 172.16.253.134/29 nbma-address '192.0.2.1'
set protocols nhrp tunnel tun100 map 172.16.253.134/29 register
set protocols nhrp tunnel tun100 multicast 'nhs'
set protocols nhrp tunnel tun100 redirect
set protocols nhrp tunnel tun100 shortcut
set vpn ipsec esp-group ESP-HUB compression 'disable'
set vpn ipsec esp-group ESP-HUB lifetime '1800'
set vpn ipsec esp-group ESP-HUB mode 'transport'
set vpn ipsec esp-group ESP-HUB pfs 'dh-group2'
set vpn ipsec esp-group ESP-HUB proposal 1 encryption 'aes256'
set vpn ipsec esp-group ESP-HUB proposal 1 hash 'sha1'
set vpn ipsec esp-group ESP-HUB proposal 2 encryption '3des'
set vpn ipsec esp-group ESP-HUB proposal 2 hash 'md5'
```

```
set vpn ipsec ike-group IKE-HUB close-action 'none'
set vpn ipsec ike-group IKE-HUB ikev2-reauth 'no'
set vpn ipsec ike-group IKE-HUB key-exchange 'ikev1'
set vpn ipsec ike-group IKE-HUB lifetime '3600'
set vpn ipsec ike-group IKE-HUB proposal 1 dh-group '2'
set vpn ipsec ike-group IKE-HUB proposal 1 encryption 'aes256'
set vpn ipsec ike-group IKE-HUB proposal 1 hash 'sha1'
set vpn ipsec ike-group IKE-HUB proposal 2 dh-group '2'
set vpn ipsec ike-group IKE-HUB proposal 2 encryption 'aes128'
set vpn ipsec ike-group IKE-HUB proposal 2 hash 'sha1'
set vpn ipsec ipsec-interfaces interface 'eth0'
set vpn ipsec profile NHRPVPN authentication mode 'pre-shared-secret'
set vpn ipsec profile NHRPVPN authentication pre-shared-secret 'secret'
set vpn ipsec profile NHRPVPN bind tunnel 'tun100'
set vpn ipsec profile NHRPVPN esp-group 'ESP-HUB'
set vpn ipsec profile NHRPVPN ike-group 'IKE-HUB'
```

Основные настройки DMVPN

```
set protocols nhrp tunnel <tunnel> cisco-authentication <secret>
```

Активирует аутентификацию для коммуникаций по протоколу NHRP, аналогичную реализации DMVPN от компании Cisco. При этом в исходящие пакеты NHRP вставляется секретный ключ в виде открытого текста. Входящие пакеты NHRP на соответствующем интерфейсе отбрасываются, если в них нет данного ключа. Максимальная длина секретного ключа - 8 символов.

```
set protocols nhrp tunnel <tunnel> dynamic-map <address> nbma-domain-name <fqdn>
```

Определяет, что NBMA адреса серверов следующего узла определены в параметре nbma-domain-name. Для каждой записи механизм реализации протокола NHRP создает динамическую запись NHS.

Каждая динамическая запись NHS получает информацию о соседнем участнике VPN сети с настроенным сетевым адресом и обнаруженным адресом NBMA.

Первый запрос Registration Request посылается на широковещательный адрес, а реальный адрес сервера динамически определяется из первого ответа Registration Reply.

- `set protocols nhrp tunnel <tunnel> holding-time <timeout>`

Определяет время удержания запросов Registration Request и ответов Resolution Reply, отправляемых с данного интерфейса. Время удержания указывается в секундах и по умолчанию равно двум часам.

- `set protocols nhrp tunnel <tunnel> map cisco`

Данную команда используется, если соседний участник VPN сети является устройством под управлением операционной системы Cisco IOS. При использовании данной команды будет статически зафиксировано значение параметра Registration Request ID, чтобы в случае изменения адреса NBMA соответствующий запрос Purge Request был отправлен. Данная команда используется, как правило, для обхода ограничения устройства под управлением операционной системы Cisco IOS, которое требует соответствия между параметром Purge Request ID и оригинальным параметром Registration Request ID.

- `set protocols nhrp tunnel <tunnel> map nbma-address <address>`

Задает для соседнего участника VPN сети статическое соответствие адреса, используемого протоколом, адресу NBMA.

Если маска для префикса IP задана, то реализации протокола NHRP будет использовать данного участника VPN сети в качестве следующего узла при отправке запросов Resolution Requests, соответствующих указанной подсети.

Заданный адрес является IP-адресом или именем FQDN центрального узла.

- `set protocols nhrp tunnel <tunnel> map register`

Определяет, путем использования необязательного параметра register, что запрос Registration Request должен быть отправлен данному участнику VPN сети при запуске механизма. Параметр register, однако, является обязательным при настройке оконечного узла.

- `set protocols nhrp tunnel <tunnel> multicast <dynamic|nhs>`

Определяет, каким образом реализация протокола NHRP будет осуществлять «мягкое» переключение трафика группового вещания. На данный момент реализация протокола NHRP предусматривает перехват трафика группового вещания при помощи пакетного сокета и его отправки обратно соответствующему получателю, что требует значительных затрат ресурсов процессора.

Использование записей NHS приведет к тому, что весь трафик группового вещания будет перенаправлен каждому статически заданному следующему узлу.

Лучшие практики предписывают пересылать трафик группового вещания всем участникам VPN сети, с которыми установлено прямое соединение. Альтернативно можно использовать данную команду несколько раз для каждого адреса, используемого протоколом, на который должен быть отправлен трафик группового вещания.

- `set protocols nhrp tunnel <tunnel> non-caching`

Отключает кэширование информации об участниках VPN сети, получаемой из перенаправляемых пакетов NHRP Resolution Reply. Использование данной команды может потребление памяти на больших NBMA подсетях.

- `set protocols nhrp tunnel <tunnel> redirect`

Включает отправку пакетов NHRP Traffic Indication по аналогии с реализацией от компании Cisco. Если эта функция включена и реализация протокола NHRP обнаруживает перенаправленный пакет, то механизм отправит сообщение оригинальному отправителю сообщение с указанием на необходимость создания прямого соединения с получателем. Другими словами, данная функция является эквивалентом сообщения ICMP Redirect.

- `set protocols nhrp tunnel <tunnel> shortcut`

Создает «короткий» (shortcut) маршрут. Полученный пакет NHRP Traffic Indication инициирует процесс разрешения и установления «короткого» маршрута.

- `set protocols nhrp tunnel <tunnel> shortcut-destination`

Предписывает реализации протокола NHRP авторизованно отвечать на пакеты NHRP Resolution Requests, направленные на адреса данного интерфейса вместо перенаправления пакетов. Это позволяет создавать «короткие» маршруты к подсетям, расположенным за этим интерфейсом. Данная команда должна быть использована для каждого интерфейса в отдельности.

- `set protocols nhrp tunnel <tunnel> shortcut-target <address>`

Определяет префикс сети за пределами NBMA, для которого GRE интерфейс будет выступать в качестве шлюза. Такой подход является альтернативным определению локальных интерфейсов с флагом shortcut-destination.

- `set protocols nhrp tunnel <tunnel> shortcut-target <address> holding-time <timeout>`

Определяет время удержания запросов Registration Request и ответов Resolution Reply, отправляемых с данного интерфейса или целевого узла, доступного через «короткий» маршрут. Время удержания указывается в секундах и по умолчанию равно двум часам.

> **Примечание**
>
> *Для детального ознакомления со связанными настройками туннелей обратитесь к разделу документации* ***Ethernet интерфейс***
>
> *Для детального ознакомления со связанными настройками стека протоколов IPSec обратитесь к разделу документации* ***Стек протоколов IPsec***

Site-to-Site VPN

Общая информация о Site-to-Site VPN

Подход к построению VPN сети Site-to-Site подразумевает добавление удаленных участников VPN сети, которые могут быть сконфигурированы для обмена зашифрованной информацией между ними и устройством под управлением операционной системы Факел или подключенными к ним сетями.

Для конфигурирования Site-to-Site подключений необходимо добавить участников VPN сети с помощью команды ***set vpn ipsec site-to-site***.

Удаленный участник VPN сети может быть идентифицирован с помощью следующих данных:

- IPv4 или IPv6 адреса. Этот способ является самым простым и используется в случае назначения участникам VPN сети статических публичных IP-адресов.
- Имя хоста. Этот способ аналогичен предыдущему, только с одним отличием - указывается DNS имя участника VPN сети вместо IP-адреса. Используется в случае, если участник имеет публичный IP-адрес и DNS имя, но IP-адрес может иногда изменяться.
- Идентификатор удаленного участника VPN сети. Этот способ не предусматривает задания IP-адреса или DNS имени участника VPN сети. Используется в случае, если участник не имеет доступных публичных IP-адресов (например, в случае использования трансляции адресов - NAT) или IP-адреса могут быть изменены.

Каждый участник VPN сети Site-to-Site располагает следующим набором параметров:

- **authentication** - набор параметров аутентификации между устройством под управлением операционной системы Факел и удаленным участником VPN сети:

- **id** - локальный идентификатор устройства под управлением операционной системы Факел. Если данный параметр задан, то в процессе аутентификации он будет направлен удаленному участнику VPN сети;
- **mode** - режим аутентификации между устройством под управлением операционной системы Факел и удаленным участником VPN сети:
  - **pre-shared-secret** - последовательность символов, представляющая собой общий секретный ключ, который должен быть одинаковым как для локальной, так и для удаленной стороны защищенной коммуникации;
  - **rsa** - общий RSA ключ, который должен быть определен в секции set vpn rsa-keys конфигурации;
  - **x509** - инфраструктура сертификатов, используемая для аутентификации.
- **pre-shared-secret** - общий секретный ключ, используется в случае, если задан параметр mode pre-shared-secret;
- **remote-id** - идентификатор удаленного участника VPN сети, который используется вместо DNS имени или IP-адреса. Данный параметр рекомендуется использовать в случае, если удаленный участник VPN сети находится за устройством с настроенным механизмом трансляции адресов (NAT) или если задан параметр mode x509;
- **rsa-key-name** - общий RSA ключ, используемый для аутентификации, который должен быть определен в секции set vpn rsa-keys конфигурации;
- **use-x509-id** - признак использования локального идентификатора из сертификата X.509. Данный параметр не может быть использован совместно с заданным параметром id;
- **x509** - набор параметров аутентификации с использованием сертификатов X.509:
  - **ca-cert-file** - путь до файла сертификата удостоверяющего центра. Используется для аутентификации удаленного участника VPN сети;
  - **cert-file** - путь до файла сертификата, используемого для аутентификации локального участника VPN сети по отношению к удаленному участнику;
  - **crl-file** - путь до файла со списком отозванных сертификатов, который используется для проверки действительности сертификата удаленного участника VPN сети;

        - **key** - закрытый ключ, который используется для аутентификации локального участника VPN сети по отношению к удаленному участнику:
            - **file** - путь до контейнера с закрытым ключом;
            - **password** - пароль для доступа к контейнеру с закрытым ключом (опционально).
- **connection-type** - режим взаимодействия устройства под управлением операционной системы Факел с удаленными участниками VPN сети:
    - **initiate** - режим, при котором устройство пытается установить соединение с удаленным участником VPN сети сразу после применения конфигурации и после перезагрузки. В этом режиме соединение не будет установлено повторно в случае его разрыва. Рекомендуется использовать только совместно механизмом DPD или другими механизмами контроля соединений.
    - **respond** - режим, при котором устройство не пытается установить соединение с удаленным участником VPN сети. В этом режиме IPSec сессия будет установлена по инициативе удаленного участника. Рекомендуется использовать в случае, если отсутствует прямое подключение к удаленному участнику, например, из-за того, что он находится за межсетевым экраном или устройством с настроенным механизмом трансляции адресов (NAT).
- **default-esp-group** - группа алгоритмов для протокола ESP по умолчанию, которая будет использована при шифровании передаваемых данных. Группа может быть переопределена отдельными настройками туннеля или VTI интерфейса;
- **description** - краткое описание для локального участника VPN сети;
- **dhcp-interface** - признак использования IP-адреса, полученного локальным участником VPN сети по протоколу DHCP, для установления IPSec сессий вместо адреса, указанного в параметре local-address;
- **force-encapsulation** - признак принудительной инкапсуляции ESP пакетов в UDP. Рекомендуется использовать в случае, если между локальным и удаленным участниками VPN сети находится межсетевой экран или устройство с настроенным механизмом трансляции адресом (NAT), которое не может обрабатывать ESP пакеты;
- **ike-group** - группа алгоритмов для протокола IKE, которая будет использована при обмене ключами;
- **ikev2-reauth** - режим повторной аутентификации удаленного участника VPN сети при смене ключей, которая выполняется только в рамках протокола IKEv2:

  - **yes** - режим, при котором со сменой ключей осуществляется попытка повторного создания всех ранее созданных записей в новой таблице SA;
  - **no** - режим, при котором со сменой ключей не происходит изменений изменений в части записей таблицы SA;
  - **inherit** - режим, при котором со сменой ключей используется порядок действий по умолчанию, обусловленный выбранной группой алгоритмов для протокола IKE.
- **local-address** - локальный IP-адрес для подключения к данному участнику VPN сети. При значении any используется IP-адрес интерфейса, на который указывает маршрут по умолчанию.
- **tunnel** - классификатор трафика, при соответствии которому принимается решение о шифровании данных, перед их передачей другому участнике VPN сети:
  - **disable** - признак отключения данного туннеля;
  - **esp-group** - группа алгоритмов для протокола ESP по отношению к выбранному туннелю;
  - **local** - локальный отправитель трафика, для которого выполняется шифрование перед передачей данному участнику VPN сети:
    - **port** - номер порта отправителя, учитывается только при совместном использовании с параметром prefix;
    - **prefix** - IP-адрес сети на локальной стороне.
  - **protocol** - протокол, для которого выполняется шифрование перед передачей данному участнике VPN сети;
  - **remote** - удаленный получатель трафика, для которого выполняется шифрование перед передачей данному участнику VPN сети:
    - **port** - номер порта отправителя, учитывается только при совместном использовании с параметром prefix;
    - **prefix** - IP-адрес сети на удаленной стороне.
- **vti** - набор параметров использования VTI интерфейса для шифрования трафика. Любой трафик, который будет отправлен на VTI интерфейс, будет зашифрован и отправлен данному участнику VPN сети. Использование VTI интерфейса упрощает настройку механизма IPSec и предоставляет некоторую гибкость в сложных сценариях использования, а также позволяет динамически добавлять и удалять сети, доступные через удаленного участника VPN сети, так как в этом режиме для устройства не требуется создание дополнительных записей в таблице SA или политик SP для каждой удаленной сети:

- **bind** - параметр, который позволяет выбрать и привязать к данному участнике VPN сети определенный VTI интерфейс;
- **esp-group** - группа алгоритмов для протокола ESP, которая будет использована для шифрования трафика, проходящего через данный VTI интерфейс.

Пример настройки Site-to-Site VPN

Протокол IKEv1

В данном примере рассматривается настройка взаимодействия двух устройств,

которые для удобства обозначены Левой стороной и Правой стороной соответственно, при следующих условиях:

- в качестве внешнего (WAN) интерфейса рассматривается интерфейс *eth1*;
- левая подсеть с адресом *192.168.0.0/24* рассматривается как сеть головного офиса;
- локальный IP-адрес левой стороны *198.51.100.3* рассматривается как внешний (WAN) IP-адрес головного офиса;
- правая подсеть с адресом *10.0.0.0/24* рассматривается как сеть удаленного филиала;
- локальный IP-адрес правой стороны *203.0.113.2* рассматривается как внешний (WAN) IP-адрес удаленного филиала.

Список команд для настройки левого маршрутизатора (LEFT):

```
set vpn ipsec esp-group office-srv-esp compression 'disable'
set vpn ipsec esp-group office-srv-esp lifetime '1800'
set vpn ipsec esp-group office-srv-esp mode 'tunnel'
set vpn ipsec esp-group office-srv-esp pfs 'enable'
set vpn ipsec esp-group office-srv-esp proposal 1 encryption
'aes256'
set vpn ipsec esp-group office-srv-esp proposal 1 hash 'sha1'
set vpn ipsec ike-group office-srv-ike ikev2-reauth 'no'
set vpn ipsec ike-group office-srv-ike key-exchange 'ikev1'
set vpn ipsec ike-group office-srv-ike lifetime '3600'
set vpn ipsec ike-group office-srv-ike proposal 1 encryption
'aes256'
```

```
set vpn ipsec ike-group office-srv-ike proposal 1 hash 'sha1'
set vpn ipsec ipsec-interfaces interface 'eth1'
set vpn ipsec site-to-site peer 203.0.113.2 authentication mode 'pre-shared-secret'
set vpn ipsec site-to-site peer 203.0.113.2 authentication pre-shared-secret 'SomePreSharedKey'
set vpn ipsec site-to-site peer 203.0.113.2 ike-group 'office-srv-ike'
set vpn ipsec site-to-site peer 203.0.113.2 local-address '198.51.100.3'
set vpn ipsec site-to-site peer 203.0.113.2 tunnel 0 allow-nat-networks 'disable'
set vpn ipsec site-to-site peer 203.0.113.2 tunnel 0 allow-public-networks 'disable'
set vpn ipsec site-to-site peer 203.0.113.2 tunnel 0 esp-group 'office-srv-esp'
set vpn ipsec site-to-site peer 203.0.113.2 tunnel 0 local prefix '192.168.0.0/24'
set vpn ipsec site-to-site peer 203.0.113.2 tunnel 0 remote prefix '10.0.0.0/21'
```

Список команд для настройки правого маршрутизатора (RIGHT):

```
set vpn ipsec esp-group office-srv-esp compression 'disable'
set vpn ipsec esp-group office-srv-esp lifetime '1800'
set vpn ipsec esp-group office-srv-esp mode 'tunnel'
set vpn ipsec esp-group office-srv-esp pfs 'enable'
set vpn ipsec esp-group office-srv-esp proposal 1 encryption 'aes256'
set vpn ipsec esp-group office-srv-esp proposal 1 hash 'sha1'
set vpn ipsec ike-group office-srv-ike ikev2-reauth 'no'
set vpn ipsec ike-group office-srv-ike key-exchange 'ikev1'
set vpn ipsec ike-group office-srv-ike lifetime '3600'
set vpn ipsec ike-group office-srv-ike proposal 1 encryption 'aes256'
set vpn ipsec ike-group office-srv-ike proposal 1 hash 'sha1'
```

```
set vpn ipsec ipsec-interfaces interface 'eth1'
set vpn ipsec site-to-site peer 198.51.100.3 authentication mode 'pre-shared-secret'
set vpn ipsec site-to-site peer 198.51.100.3 authentication pre-shared-secret 'SomePreSharedKey'
set vpn ipsec site-to-site peer 198.51.100.3 ike-group 'office-srv-ike'
set vpn ipsec site-to-site peer 198.51.100.3 local-address '203.0.113.2'
set vpn ipsec site-to-site peer 198.51.100.3 tunnel 0 allow-nat-networks 'disable'
set vpn ipsec site-to-site peer 198.51.100.3 tunnel 0 allow-public-networks 'disable'
set vpn ipsec site-to-site peer 198.51.100.3 tunnel 0 esp-group 'office-srv-esp'
set vpn ipsec site-to-site peer 198.51.100.3 tunnel 0 local prefix '10.0.0.0/21'
set vpn ipsec site-to-site peer 198.51.100.3 tunnel 0 remote prefix '192.168.0.0/24'
```

После применения настроек, убедитесь что Site-ti-Site туннель между правой и левой стороной активны:

```
fakel@hq:~$ show vpn ike sa
Peer ID / IP                            Local ID / IP
------------                            -------------
203.0.113.2                             198.51.100.3

State   Encrypt   Hash   D-H Grp   NAT-T   A-Time   L-Time
-----   -------   ----   -------   -----   ------   ------
up      aes256    sha1   5         no      734      3600

fakel@hq:~$ show vpn ipsec sa
Peer ID / IP                            Local ID / IP
------------                            -------------
203.0.113.2                             198.51.100.3

Tunnel   State   Bytes Out/In   Encrypt   Hash   NAT-T   A-Time   L-Time   Proto
------   -----   ------------   -------   ----   -----   ------   ------   -----
```

```
0        up      7.5M/230.6K    aes256    sha1    no      567      1800     all
```

При настроенных правилах трансляции адреса отправителя (SNAT) на интерфейсе eth1 необходимо добавить исключающее правило.

Список команд для настройки правил трансляции на левом маршрутизаторе (LEFT):

```
set nat source rule 10 destination address '10.0.0.0/24'
set nat source rule 10 'exclude'
set nat source rule 10 outbound-interface 'eth1'
set nat source rule 10 source address '192.168.0.0/24'
```

Список команд для настройки правил трансляции на правом маршрутизаторе (RIGHT):

```
set nat source rule 10 destination address '192.168.0.0/24'
set nat source rule 10 'exclude'
set nat source rule 10 outbound-interface 'eth1'
set nat source rule 10 source address '10.0.0.0/24'
```

Для прохождения трафика, адресованного клиентским хостам, необходимо добавить соответствующие разрешающие правила фильтрации.

Список команд для настройки правил межсетевого экрана на левом маршрутизаторе (LEFT):

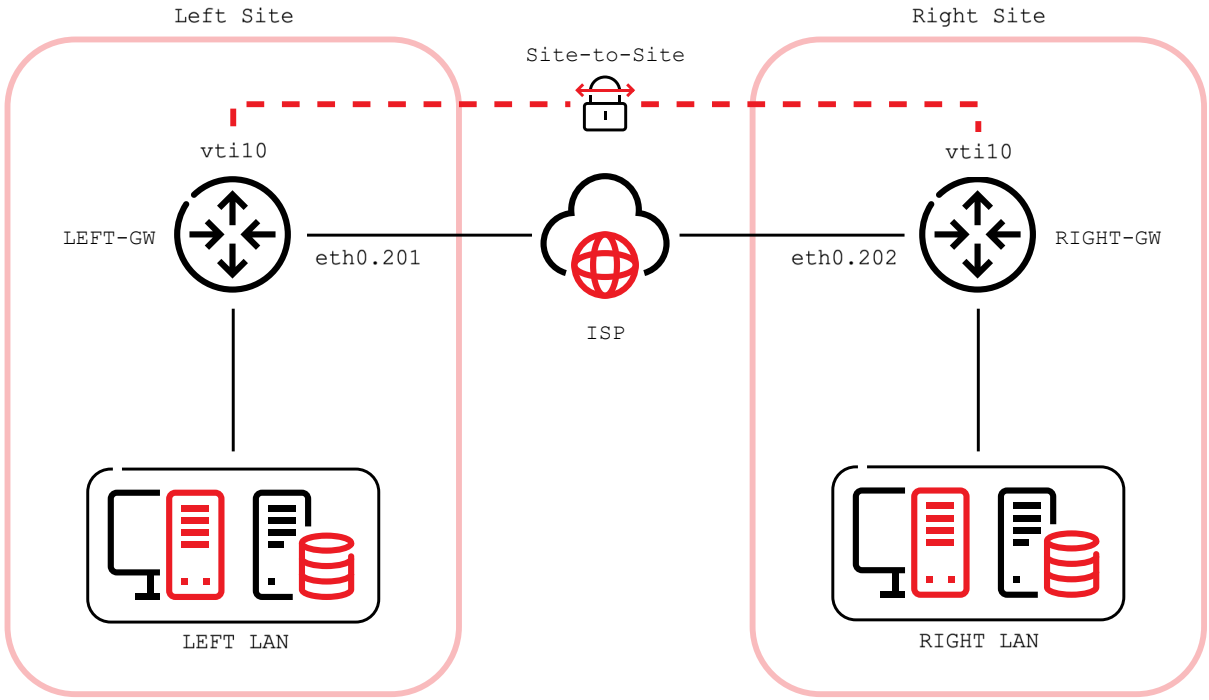
```
set firewall name OUTSIDE-LOCAL rule 32 action 'accept'
set firewall name OUTSIDE-LOCAL rule 32 source address
'10.0.0.0/24'
```

Список команд для настройки правил трансляции на правом маршрутизаторе (RIGHT):

```
set firewall name OUTSIDE-LOCAL rule 32 action 'accept'
set firewall name OUTSIDE-LOCAL rule 32 source address
'192.168.0.0/24'
```

Протокол IKEv2

Рассмотрим следующую топологию сет



В данном примере рассматривается настройка взаимодействия двух устройств при следующих условиях:

- *10.0.0.2/31* – IP-адрес виртуального интерфейса *vti10* на маршрутизаторе *LEFT-GW*;
- *10.0.0.3/31* – IP-адрес виртуального интерфейса *vti10* на маршрутизаторе *RIGHT-GW*;
- *172.18.201.10/24* – IP-адрес внешнего WAN интерфейса *eth0.210* на маршрутизаторе *LEFT-GW*;
- *172.18.202.10/24* – IP-адрес внешнего WAN интерфейса *eth0.220* на маршрутизаторе *RIGHT-GW*;
- *172.18.201.254/24* – IP-адрес шлюза по умолчанию для маршрутизатора *LEFT-GW*;
- *172.18.202.254/24* – IP-адрес шлюза по умолчанию для маршрутизатора *RIGHT-GW*.

Список команд для настройки левого маршрутизатора (LEFT):

```
set interfaces vti vti10 address '10.0.0.2/31'
set vpn ipsec esp-group ESP_DEFAULT compression 'disable'
set vpn ipsec esp-group ESP_DEFAULT lifetime '3600'
```

```
set vpn ipsec esp-group ESP_DEFAULT mode 'tunnel'
set vpn ipsec esp-group ESP_DEFAULT pfs 'dh-group19'
set vpn ipsec esp-group ESP_DEFAULT proposal 10 encryption
'aes256gcm128'
set vpn ipsec esp-group ESP_DEFAULT proposal 10 hash 'sha256'
set vpn ipsec ike-group IKEv2_DEFAULT dead-peer-detection
action 'hold'
set vpn ipsec ike-group IKEv2_DEFAULT dead-peer-detection
interval '30'
set vpn ipsec ike-group IKEv2_DEFAULT dead-peer-detection
timeout '120'
set vpn ipsec ike-group IKEv2_DEFAULT ikev2-reauth 'no'
set vpn ipsec ike-group IKEv2_DEFAULT key-exchange 'ikev2'
set vpn ipsec ike-group IKEv2_DEFAULT lifetime '10800'
set vpn ipsec ike-group IKEv2_DEFAULT mobike 'disable'
set vpn ipsec ike-group IKEv2_DEFAULT proposal 10 dh-group '19'
set vpn ipsec ike-group IKEv2_DEFAULT proposal 10 encryption
'aes256gcm128'
set vpn ipsec ike-group IKEv2_DEFAULT proposal 10 hash 'sha256'
set vpn ipsec ipsec-interfaces interface 'eth0.201'
set vpn ipsec site-to-site peer 172.18.202.10 authentication id
'172.18.201.10'
set vpn ipsec site-to-site peer 172.18.202.10 authentication
mode 'pre-shared-secret'
set vpn ipsec site-to-site peer 172.18.202.10 authentication
pre-shared-secret 'secretkey'
set vpn ipsec site-to-site peer 172.18.202.10 authentication
remote-id '172.18.202.10'
set vpn ipsec site-to-site peer 172.18.202.10 connection-type
'initiate'
set vpn ipsec site-to-site peer 172.18.202.10 ike-group
'IKEv2_DEFAULT'
set vpn ipsec site-to-site peer 172.18.202.10 ikev2-reauth
'inherit'
set vpn ipsec site-to-site peer 172.18.202.10 local-address
'172.18.201.10'
```

```
set vpn ipsec site-to-site peer 172.18.202.10 vti bind 'vti10'
set vpn ipsec site-to-site peer 172.18.202.10 vti esp-group 'ESP_DEFAULT'
```

Список команд для настройки правого маршрутизатора (RIGHT):

```
set interfaces vti vti10 address '10.0.0.3/31'
set vpn ipsec esp-group ESP_DEFAULT compression 'disable'
set vpn ipsec esp-group ESP_DEFAULT lifetime '3600'
set vpn ipsec esp-group ESP_DEFAULT mode 'tunnel'
set vpn ipsec esp-group ESP_DEFAULT pfs 'dh-group19'
set vpn ipsec esp-group ESP_DEFAULT proposal 10 encryption 'aes256gcm128'
set vpn ipsec esp-group ESP_DEFAULT proposal 10 hash 'sha256'
set vpn ipsec ike-group IKEv2_DEFAULT dead-peer-detection action 'hold'
set vpn ipsec ike-group IKEv2_DEFAULT dead-peer-detection interval '30'
set vpn ipsec ike-group IKEv2_DEFAULT dead-peer-detection timeout '120'
set vpn ipsec ike-group IKEv2_DEFAULT ikev2-reauth 'no'
set vpn ipsec ike-group IKEv2_DEFAULT key-exchange 'ikev2'
set vpn ipsec ike-group IKEv2_DEFAULT lifetime '10800'
set vpn ipsec ike-group IKEv2_DEFAULT mobike 'disable'
set vpn ipsec ike-group IKEv2_DEFAULT proposal 10 dh-group '19'
set vpn ipsec ike-group IKEv2_DEFAULT proposal 10 encryption 'aes256gcm128'
set vpn ipsec ike-group IKEv2_DEFAULT proposal 10 hash 'sha256'
set vpn ipsec ipsec-interfaces interface 'eth0.202'
set vpn ipsec site-to-site peer 172.18.201.10 authentication id '172.18.202.10'
set vpn ipsec site-to-site peer 172.18.201.10 authentication mode 'pre-shared-secret'
set vpn ipsec site-to-site peer 172.18.201.10 authentication pre-shared-secret 'secretkey'
```

- set vpn ipsec site-to-site peer 172.18.201.10 authentication remote-id '172.18.201.10'
- set vpn ipsec site-to-site peer 172.18.201.10 connection-type 'initiate'
- set vpn ipsec site-to-site peer 172.18.201.10 ike-group 'IKEv2_DEFAULT'
- set vpn ipsec site-to-site peer 172.18.201.10 ikev2-reauth 'inherit'
- set vpn ipsec site-to-site peer 172.18.201.10 local-address '172.18.202.10'
- set vpn ipsec site-to-site peer 172.18.201.10 vti bind 'vti10'
- set vpn ipsec site-to-site peer 172.18.201.10 vti esp-group 'ESP_DEFAULT'

Основные настройки Site-to-Site VPN

- set vpn ipsec esp-group <text> compression <enable | disable>

- set vpn ipsec esp-group <text> lifetime <30-86400>

- set vpn ipsec esp-group <text> mode <tunnel | transport>

- set vpn ipsec esp-group <text> pfs <enable | disable | dh-groupN>

- set vpn ipsec esp-group <text> proposal <1-65535> encryption <type>

- set vpn ipsec esp-group <text> proposal <1-65535> hash <type>

- set vpn ipsec ike-group <text> ikev2-reauth <yes | no>

- `set vpn ipsec ike-group <text> key-exchange <ikev1 | ikev2>`

- `set vpn ipsec ike-group <text> lifetime <30-86400>`

- `set vpn ipsec ike-group <text> proposal <1-65535> encryption <type>`

- `set vpn ipsec ike-group <text> proposal <1-65535> hash <type>`

- `set vpn ipsec ipsec-interfaces interface <ethN>`

- `set vpn ipsec site-to-site peer <x.x.x.x> authentication mode <pre-shared-secret | rsa | x509>'`

- `set vpn ipsec site-to-site peer <x.x.x.x> authentication pre-shared-secret <text>`

- `set vpn ipsec site-to-site peer <x.x.x.x> ike-group <text>`

- `set vpn ipsec site-to-site peer <x.x.x.x> local-address <x.x.x.x>`

- `set vpn ipsec site-to-site peer <x.x.x.x> tunnel <0-4294967295> allow-nat-networks <enable | disable>`

- `set vpn ipsec site-to-site peer <x.x.x.x> tunnel <0-4294967295> esp-group <text>`

- `set vpn ipsec site-to-site peer <x.x.x.x> tunnel <0-4294967295> local prefix <x.x.x.x/x>`

- `set vpn ipsec site-to-site peer <x.x.x.x> tunnel <0-4294967295> remote prefix <x.x.x.x/x>`

- `set interfaces vti <vtiN> address <x.x.x.x/x>`

- `set vpn ipsec ike-group <text> dead-peer-detection action <hold | clear | restart>`

- `set vpn ipsec ike-group <text> dead-peer-detection interval <2-86400>`

- `set vpn ipsec ike-group <text> dead-peer-detection timeout <2-86400>`

- `set vpn ipsec ike-group <text> mobike <enable | disable>`

- `set vpn ipsec ike-group <text> proposal <1-65535> dh-group <number>`

- `set vpn ipsec site-to-site peer <x.x.x.x> authentication id <x.x.x.x>`

- `set vpn ipsec site-to-site peer <x.x.x.x> authentication remote-id <x.x.x.x>`

- `set vpn ipsec site-to-site peer <x.x.x.x> connection-type <initiate | respond>`

- `set vpn ipsec site-to-site peer <x.x.x.x> ike-group <text>`

- `set vpn ipsec site-to-site peer <x.x.x.x> ikev2-reauth <yes | no | inherit>`

- `set vpn ipsec site-to-site peer <x.x.x.x> local-address <x.x.x.x>`

- `set vpn ipsec site-to-site peer <x.x.x.x> vti bind <vtiN>`

- `set vpn ipsec site-to-site peer <x.x.x.x> vti esp-group <text>`

Контроль трафика

В ПО **Факел** контроль трафика реализован следующими механизмами:

- Межсетевой экран (Firewall)
- Трансляция адресов (NAT)

Межсетевой экран

Общая информация о работе межсетевого экрана

ПО **Факел** использует для фильтрации сетевых пакетов стандартную подсистему netfilter, которая интегрирована во все операционные системы Linux.

Межсетевой экран предоставляет возможность создания групп портов, адресов и подсетей с помощью механизма ipset, а также возможность задания политики фильтрации, основанной на интерфейсах или зонах. Межсетевой экран в составе операционной системы Факел предоставляет возможность использования в правилах фильтрации групп - списков IP адресов, подсетей или портов. После создания группа может быть указана в правилах в качестве отправителя или получателя. Записи могут добавляться в группу или удалятся из нее без необходимости изменения самой группы или правил, в которых она указана.

Группы должны иметь уникальные имена, даже если они содержат идентичные значения IPv4 или IPv6 адресов.

Межсетевой экран в части политики фильтрации оперирует такими объектами как in, out и local. Пользователи, знакомые с подсистемой netfilter часто ошибочно ассоциируют объект in с цепочкой INPUT, а объект out - с цепочкой OUTPUT. В действительности же оба этих объекта ассоциированы с цепочкой FORWARD, но обозначают входящий и исходящий сетевой интерфейс по отношению к проходящему через устройство трафику соответственно. Цепочка INPUT, которая определяет порядок обработки трафика, адресованного локальным сервисам операционной системы Факел, ассоциирована с объектом local, но с учетом определенного входящего сетевого интерфейса.

Пример конфигурации межсетевого экрана:

```
firewall {
    all-ping enable
    broadcast-ping disable
    config-trap disable
    group {
        network-group BAD-NETWORKS {
            network 198.51.100.0/24
            network 203.0.113.0/24
        }
        network-group GOOD-NETWORKS {
            network 192.0.2.0/24
        }
        port-group BAD-PORTS {
            port 65535
        }
    }
    name FROM-INTERNET {
        default-action accept
        description "From the Internet"
        rule 10 {
            action accept
            description "Authorized Networks"
            protocol all
            source {
                group {
                    network-group GOOD-NETWORKS
                }
            }
        }
        rule 11 {
            action drop
            description "Bad Networks"
            protocol all
            source {
                group {
                    network-group BAD-NETWORKS
                }
```

```
            }
        }
        rule 30 {
            action drop
            description "BAD PORTS"
            destination {
                group {
                    port-group BAD-PORTS
                }
            }
            log enable
            protocol all
        }
    }
}
interfaces {
    ethernet eth1 {
        address dhcp
        description OUTSIDE
        duplex auto
        firewall {
            in {
                name FROM-INTERNET
            }
        }
    }
}
```

Глобальные настройки межсетевого экрана

- `set firewall all-ping enable`

Активирует механизм отправки ответа на каждое сообщение ICMP Echo Request, адресованное локальным сервисам. В случае наличия других правил для блокировки данного трафика операционная система Факел не будет отвечать на сообщения ICMP Echo Request.

- `set firewall all-ping disable`

Отключает механизм отправки ответа на сообщение ICMP Echo Request, адресованное локальным сервисам.

> **Примечание**
>
> *Команда firewall all-ping влияет только на локальный трафик и всегда используется для ограничения доступа.По умолчанию операционная система Факел при получении сообщения ICMP Echo Request в свой адрес будет отвечать сообщением ICMP Echo Reply до тех пор, пока данный трафик не будет заблокирован с помощью межсетевого экрана.*

- `set firewall broadcast-ping [enable | disable]`

Включает/отключает возможность ответа на широковещательные ICMP сообщения.

При использовании команды изменяются значения следующих системных параметров:

- *net.ipv4.icmp_echo_ignore_broadcasts*

- `set firewall ip-src-route [enable | disable]`
- `set firewall ipv6-src-route [enable | disable]`

Включают/отключают возможность приема IP пакетов с опцией Source Route в заголовке.

При использовании команд изменяются значения следующих системных параметров

- *net.ipv4.conf.all.accept_source_route*
- *net.ipv6.conf.all.accept_source_route*

- `set firewall receive-redirects [enable | disable]`
- `set firewall ipv6-receive-redirects [enable | disable]`

Включают/отключают возможность приема сообщений ICMP Redirect.

При использовании команд изменяются значения следующих системных параметров:

- *net.ipv4.conf.all.accept_redirects*
- *net.ipv6.conf.all.accept_redirects*

- `set firewall send-redirects [enable | disable]`

Включает/отключает возможность передачи сообщений ICMP Reidrect.

При использовании команды изменяются значения следующих системных параметров:

- *net.ipv4.conf.all.send_redirects*

- `set firewall log-martians [enable | disable]`

Включает/отключает регистрацию прохождения видимых в общих сетях (Интернет) IPv4 пакетов - Martian Packets.

При использовании команды изменяются значения следующих системных параметров:

- *net.ipv4.conf.all.log_martians*

- `set firewall source-validation [strict | loose | disable]`

Включает/отключает возможность валидации IPv4 адреса отправителя - фактически защиту от подмены IP адресов - IP Spoofing.

При использовании данной команды изменяются значения следующих системных параметров:

- *net.ipv4.conf.all.rp_filter*

- `set firewall syn-cookies [enable | disable]`

Включает/отключает возможность использования механизма TCP SYN Cookies, используемого для противодействия DoS атакам типа SYN Flood.

При использовании команды изменяются значения следующих системных параметров:

- *net.ipv4.tcp_syncookies*

- `set firewall twa-hazards-protection [enable | disable]`

Обеспечивает соответствие операционной системы Факел положениям спецификации RFC 1337.

При использовании команды изменяются значения следующих системных параметров:

- *net.ipv4.tcp_rfc1337*

- `set firewall state-policy established action [accept | drop | reject]`

Определяет политику фильтрации для установленных соединений (соединений, для которых состояние - ESTABLISHED).

- `set firewall state-policy established log enable`

Включает запись журнала событий политики фильтрации для установленных соединений (соединений, для которых состояние - ESTABLISHED).

- `set firewall state-policy invalid action [accept | drop | reject]`

Определяет политику фильтрации для некорректных пакетов - пакетов, не прошедших проверку Packet Sanity.

- `set firewall state-policy invalid log enable`

Включает запись журнала событий политики фильтрации для некорректных пакетов - пакетов, не прошедших проверку Packet Sanity.

- `set firewall state-policy related action [accept | drop | reject]`

Определяет политику фильтрации для связанных соединений (соединений, для которых состояние - RELATED).

- `set firewall state-policy related log enable`

Включает запись журнала событий политики фильтрации для связанных соединений (соединений, для которых состояние - RELATED).

Настройка группы адресов для межсетевого экрана

Группа адресов представляет собой список IP адресов хостов или диапазонов IP адресов.

- `set firewall group address-group <name> address [address | address range]`

Создает группу <*name*> IPv4 адресов <*address* | *address range*>.

- `set firewall group ipv6-address-group <name> address <address>`

Создает группу <*name*> IPv6 адресов <*address*>.

- `set firewall group address-group <name> description <text>`

Добавляет текстовое описание <*text*> для созданной группы адресов IPv4 <*name*>.

- `set firewall group ipv6-address-group <name> description <text>`

Добавляет текстовое описание <*text*> для созданной группы адресов IPv6 <*name*>.

Настройка группы подсетей для межсетевого экрана

Группа подсетей представляет собой список IP подсетей в нотации CIDR. Определенные адреса могут быть добавлены в качестве 32-разрядного префикса. Если есть необходимость включить в группу одновременно адреса хостов и адреса подсетей, то рекомендуется использовать именно данный тип группы.

- `set firewall group network-group <name> network <CIDR>`

Создает группу <*name*> подсетей IPv4 <*CIDR*>.

- `set firewall group ipv6-network-group <name> network <CIDR>`

Создает группу <*name*> подсетей IPv6 <*CIDR*>.

- `set firewall group network-group <name> description <text>`

Добавляет текстовое описание <*text*> для созданной группы подсетей IPv4 <*name*>.

- `set firewall group ipv6-network-group <name> description <text>`

Добавляет текстовое описание <*text*> для созданной группы подсетей IPv6 <*name*>.

Настройка группы портов для межсетевого экрана

Группы портов представляют собой список только номеров портов, без указания протокола. Группы портов могут быть указаны в правиле фильтрации только совместно с протоколами TCP и UDP. Рекомендуется отдельно группировать порты для протоколов TCP и UDP, чтобы избежать случайной блокировки портов, доступ к которым необходимо сохранить. Диапазон номеров портов может быть определен с помощью символа «-».

- `set firewall group port-group <name> port [portname | portnumber | startport-endport]`

Создает группу портов. Значение параметра portname должно быть символьным именем сервиса из списка, определенного в файле /etc/services, например http.

- `set firewall group port-group <name> description <text>`

Добавляет текстовое описание <*text*> для созданной группы портов <*name*>.

Настройка набора правил для межсетевого экрана

Набор правил - именованный список правил фильтрации, которые могут быть применены к интерфейсу или зоне. Каждое правило имеет порядковый номер,

классификатор, на предмет соответствие которому проверяются проходящие через межсетевой экран сетевые пакеты, и действие, которое будет применено к пакету в случае срабатывания правила. При поступлении сетевого пакета выполняется проход по списку правил в заданном порядке (1 - 999999), срабатывает первое подошедшее правило.

```
set firewall name <name> description <text>
```

Добавляет текстовое описание *<text>* для набора привил фильтрации трафика IPv4 *<name>*.

```
set firewall ipv6-name <name> description <text>
```

Добавляет текстовое описание *<text>* для набора привил фильтрации трафика IPv6 *<name>*.

```
set firewall name <name> default-action [drop | reject |
accept]
```

Задает действие по умолчанию для созданного набора правил фильтрации трафика IPv4 - действия, которое будет применено к пакету в случае, если ни одно правило не подошло.

```
set firewall ipv6-name <name> default-action [drop | reject |
accept]
```

Задает действие по умолчанию для созданного набора правил фильтрации трафика IPv6 - действия, которое будет применено к пакету в случае, если ни одно правило не подошло.

```
set firewall name <name> enable-default-log
```

Включает запись событий правил фильтрации трафика IPv4, к которому применено действие по умолчанию.

```
set firewall ipv6-name <name> enable-default-log
```

Включает запись событий правил фильтрации трафика IPv6, к которому применено действие по умолчанию.

```
set firewall name <name> rule <1-999999> action [drop | reject
| accept]
```

Задет действие для выбранного правила *<1-999999>* из созданного набора правил фильтрации трафика IPv4 *<name>*.

- `set firewall ipv6-name <name> rule <1-999999> action [drop | reject | accept]`

Задет действие для выбранного правила *<1-999999>* из созданного набора правил фильтрации трафика IPv6 *<name>*.

- `set firewall name <name> rule <1-999999> description <text>`

Добавляет текстовое описание *<text>* для выбранного правила *<1-999999>* из набора правил фильтрации трафика IPv4 *<name>*.

- `set firewall ipv6-name <name> rule <1-999999> description <text>`

Добавляет текстовое описание *<text>* для выбранного правила *<1-999999>* из набора правил фильтрации трафика IPv6 *<name>*.

- `set firewall name <name> rule <1-999999> log [disable | enable]`

Включает запись событий правил фильтрации трафика IPv4, для выбранного правила *<1-999999>* из созданного набора *<name>*.

- `set firewall ipv6-name <name> rule <1-999999> log [disable | enable]`

Включает запись событий правил фильтрации трафика IPv6, для выбранного правила *<1-999999>* из созданного набора *<name>*.

- `set firewall name <name> rule <1-999999> disable`

Отключает выбранное правило *<1-999999>* из набора правил фильтрации трафика IPv4 *<name>*. После отключения правило фильтрации остается в конфигурации операционной системы.

- `set firewall ipv6-name <name> rule <1-999999> disable`

Отключает выбранное правило *<1-999999>* из набора правил фильтрации трафика IPv6 *<name>*. После отключения правило фильтрации остается в конфигурации операционной системы.

Настройка классификатора правил для межсетевого экрана

Межсетевой экран в составе операционной системы Факел позволяет задать различные варианты классификаторов в правиле фильтрации, на предмет соответствия которым будет проверяться сетевой пакет.

- `set firewall name <name> rule <1-999999> source address [address | addressrange | CIDR]`

Задает классификатор проверки соответствия адреса источника *[address | addressrange | CIDR]* для правила фильтрации трафика IPv4 *<1-999999>*. Можно задавать отрицание (инверсию) с помощью символа «**!**».

- `set firewall name <name> rule <1-999999> destination address [address | addressrange | CIDR]`

Задает классификатор проверки соответствия адреса получателя *[address | addressrange | CIDR]* для правила фильтрации трафика IPv4 *<1-999999>*. Можно задавать отрицание (инверсию) с помощью символа «!».

- `set firewall ipv6-name <name> rule <1-999999> source address [address | addressrange | CIDR]`

Задает классификатор проверки соответствия адреса источника *[address | addressrange | CIDR]* для правила фильтрации трафика IPv6 *<1-999999>*. Можно задавать отрицание (инверсию) с помощью символа «*!*».

- `set firewall ipv6-name <name> rule <1-999999> destination address [address | addressrange | CIDR]`

Задает классификатор проверки соответствия адреса получателя *[address | addressrange | CIDR]* для правила фильтрации трафика IPv6 *<1-999999>*. Можно задавать отрицание (инверсию) с помощью символа «*!*».

- `set firewall name <name> rule <1-999999> source mac-address <mac-address>`

Задает классификатор проверки соответствия MAC адреса источника *<mac-address>* для правила фильтрации трафика IPv4 *<1-999999>*.

- `set firewall ipv6-name <name> rule <1-999999> source mac-address <mac-address>`

Задает классификатор проверки соответствия MAC адреса источника *<mac-address>* для правила фильтрации трафика IPv6 *<1-999999>*.

- `set firewall name <name> rule <1-999999> source port [1-65535 | portname | start-end]`

Задает классификатор проверки соответствия порта источника *[1-65535 | portname | start-end]* для правила фильтрации трафика IPv4 *<1-999999>*.

- `set firewall name <name> rule <1-999999> destination port [1-65535 | portname | start-end]`

Задает классификатор проверки соответствия порта получателя *[1-65535 | portname | start-end]* для правила фильтрации трафика IPv4 *<1-999999>*.

- `set firewall ipv6-name <name> rule <1-999999> source port [1-65535 | portname | start-end]`

Задает классификатор проверки соответствия порта источника *[1-65535 | portname | start-end]* для правила фильтрации трафика IPv6 *<1-999999>*.

- `set firewall ipv6-name <name> rule <1-999999> destination port [1-65535 | portname | start-end]`

Задает классификатор проверки соответствия порта получателя *[1-65535 | portname | start-end]* для правила фильтрации трафика IPv6 *<1-999999>*.

Примечание

При определении классификатора вы можете использовать номер порта или символьное имя сервиса из списка, определенного в файле /etc/services. Можно также задавать несколько значений в виде списка значений, разделенных символом «,». Весь список значений может быть инвертирован с помощью символа «!».

- `set firewall name <name> rule <1-999999> source group address-group <name>`

Задает классификатор проверки соответствия адреса источника из группы адресов *<name>* для правила фильтрации трафика IPv4 *<1-999999>*.

- `set firewall name <name> rule <1-999999> destination group address-group <name>`

Задает классификатор проверки соответствия адреса получателя из группы адресов *<name>* для правила фильтрации трафика IPv4 *<1-999999>*.

- `set firewall ipv6-name <name> rule <1-999999> source group address-group <name>`

Задает классификатор проверки соответствия адреса источника из группы адресов *<name>* для правила фильтрации трафика IPv6 *<1-999999>*.

- `set firewall ipv6-name <name> rule <1-999999> destination group address-group <name>`

Задает классификатор проверки соответствия адреса получателя из группы адресов *<name>* для правила фильтрации трафика IPv6 *<1-999999>*.

```
set firewall name <name> rule <1-999999> source group network-group <name>
```

Задает классификатор проверки соответствия подсети источника из группы подсетей *<name>* для правила фильтрации трафика IPv4 *<1-999999>*.

```
set firewall name <name> rule <1-999999> destination group network-group <name>
```

Задает классификатор проверки соответствия подсети получателя из группы подсетей *<name>* для правила фильтрации трафика IPv4 *<1-999999>*.

```
set firewall ipv6-name <name> rule <1-999999> source group network-group <name>
```

Задает классификатор проверки соответствия подсети источника из группы подсетей *<name>* для правила фильтрации трафика IPv6 *<1-999999>*.

```
set firewall ipv6-name <name> rule <1-999999> destination group network-group <name>
```

Задает классификатор проверки соответствия подсети получателя из группы подсетей *<name>* для правила фильтрации трафика IPv6 *<1-999999>*.

```
set firewall name <name> rule <1-999999> source group port-group <name>
```

Задает классификатор проверки соответствия порта источника из группы портов *<name>* для правила фильтрации трафика IPv4 *<1-999999>*.

```
set firewall name <name> rule <1-999999> destination group port-group <name>
```

Задает классификатор проверки соответствия порта получателя из группы портов *<name>* для правила фильтрации трафика IPv4 *<1-999999>*.

```
set firewall ipv6-name <name> rule <1-999999> source group port-group <name>
```

Задает классификатор проверки соответствия порта источника из группы портов *<name>* для правила фильтрации трафика IPv6 *<1-999999>*.

```
set firewall ipv6-name <name> rule <1-999999> destination group port-group <name>
```

Задает классификатор проверки соответствия порта получателя из группы портов *<name>* для правила фильтрации трафика IPv6 *<1-999999>*.

```
set firewall name <name> rule <1-999999> protocol [<text> | <0-255> | all | tcp_udp]
```

Задает классификатор проверки соответствия протокола *[<text> | <0-255> | all | tcp_udp]* для правила фильтрации трафика IPv4 *<1-999999>*.

```
set firewall ipv6-name <name> rule <1-999999> protocol [<text> | <0-255> | all | tcp_udp]
```

Задает классификатор проверки соответствия протокола *[<text> | <0-255> | all | tcp_udp]* для правила фильтрации трафика IPv6 *<1-999999>*.

> **Примечание**
>
> *При определении классификатора вы можете использовать номер протокола или его символьное имя из списка, определенного в файле /etc/protocols. Также для использования доступны специальные обозначения* ***all*** *(все протоколы)****,*** *tcp_udp (только протоколы TCP и UDP) и «!» (инверсия списка заданных значений).*

```
set firewall name <name> rule <1-999999> tcp flags <text>
```

Задает классификатор проверки соответствия флагов в TCP заголовке *<text>* для правила фильтрации трафика IPv4 *<1-999999>*.

```
set firewall ipv6-name <name> rule <1-999999> tcp flags <text>
```

Задает классификатор проверки соответствия флагов в TCP заголовке *<text>* для правила фильтрации трафика IPv6 *<1-999999>*.

> **Примечание**
>
> *При определении классификатора вы можете указать флаги в заголовке TCP. Доступны следующие значения: SYN, ACK, FIN, RST, URG, PSH, ALL. Можно также задавать несколько значений в виде списка значений, разделенных символом «,». Каждое отдельное значение в списке может быть инвертировано с помощью символа «!».*

```
set firewall name <name> rule <1-999999> state [established | invalid | new | related] [enable | disable]
```

Задает классификатор проверки соответствия состояния соединений *[established | invalid | new | related]* для правила фильтрации трафика IPv4 *<1-999999>*.

```
set firewall ipv6-name <name> rule <1-999999> state [established | invalid | new | related] [enable | disable]
```

Задает классификатор проверки соответствия состояния соединений *[established | invalid | new | related]* для правила фильтрации трафика IPv6 *<1-999999>*.

```
set firewall name <name> rule <1-999999> recent count <1-255>
set firewall name <name> rule <1-999999> recent time <second | minute | hour>
```

Задает классификатор проверки количество соединений (recent count) *<1-255>* за определенный промежуток времени - (time) *<second | minute | hour>* для правила фильтрации трафика IPv4. Классификатор, заданный таким образом, может быть использован для блокировки атак типа Brute Force.

```
set firewall ipv6-name <name> rule <1-999999> recent count <1-255>
set firewall ipv6-name <name> rule <1-999999> recent time <second | minute | hour>
```

Задает классификатор проверки количество соединений (recent count) *<1-255>* за определенный промежуток времени - (time) *<second | minute | hour>* для правила фильтрации трафика IPv6. Классификатор, заданный таким образом, может быть использован для блокировки атак типа Brute Force.

Применение набора правил для межсетевого экрана к сетевому интерфейсу

Набор правил может быть применен к каждому сетевому интерфейсу устройства одним из следующих способов:

- **in** - набор правил для транзитного трафика на входящем интерфейсе;
- **out** - набор правил для транзитного трафика на исходящем интерфейсе;
- **local** - набор правил для входящего трафика, получателем которого является само устройство.

```
set interface ethernet <ethN> firewall [in | out | local] [name | ipv6-name] <rule-set>
```

Применяет набор правил *<rule-set>* к сетевому интерфейсу *<ethN>* с учетом пути прохождения трафика *[in | out | local]*.

Пример назначения набора правил сетевому интерфейсу:

```
set interface ethernet eth1 vif 100 firewall in name LANv4-IN
```

```
set interface ethernet eth1 vif 100 firewall out name LANv4-OUT
set interface bonding bond0 firewall in name LANv4-IN
set interfaces openvpn vtun1 firewall in name Lanv4-IN
```

Пример, представленный выше, показывает, что вы также можете применить один и тот же набор правил к нескольким сетевым интерфейсам, но к отдельному сетевому интерфейсу может быть применен только один набор правил.

Мониторинг работы межсетевого экрана

```
show firewall
```

Выводит на экран общие сведения о межсетевом экране.

```
show firewall summary
```

Выводит на экран общие сведения о межсетевом экране в разрезе наборов правил и групп.

```
show firewall statistics
```

Выводит на экран статистику по всем наборам правил с момента последней загрузки.

```
show firewall [name | ipv6name] <name> rule <1-999999>
```

Выводит на экран сведения об отдельном правиле в составе определенного набора.

```
show firewall group <name>
```

Выводит на экран сведения об определенной группе: ее тип, состав и в каких правилах она используется.

```
show firewall [name | ipv6name] <name>
```

Выводит на экран сведения об определенном наборе правил.

```
show firewall [name | ipv6name] <name> statistics
```

Выводит на экран статистику по определенному набору правил с момента последней загрузки.

```
show zone-policy zone <name>
```

Выводит на экран сведения об определенной зоне.

- `show log firewall [name | ipv6name] <name>`

Выводит на экран список событий, соответствующих срабатываниям определенного набора правил.

> **Примечание**
>
> *Полный список событий, зарегистрированных в ходе работы межсетевого экрана, доступен для просмотра в файле /var/logs/messages.*

Фиксация TCP MSS

Широко распространенный в сети Интернет механизм Path MTU Discovery в действительности работает крайне редко. Поэтому иногда приходится фиксировать определенное значение параметра TCP MSS. Это поле используется в заголовке TCP совместно с флагом SYN. Устанавливая определенное значение TCP MSS, вы таким образом сообщаете удаленному участнику коммуникации, чтобы он не направлял вам блоки данных, превышающие по объему данное значение. Данный подход называется TCP MSS Clamping

Операционная система Факел содержит параметр межсетевого экрана, который позволяет зафиксировать значение TCP MSS для протоколов IPv4 и IPv6.

> **Примечание**
>
> *Обычно значение TCP MSS рассчитывается по следующей формуле: TCP MSS = MTU - IP Header - TCP Header, где IP Header = 20 байт и TCP Header = 20 байт. Поэтому для MTU = 1492 байт в результате получается значение TCP MSS = 1452 байт.*

Фиксация для IPv4

- `set firewall options interface <interface> adjust-mss <mss | clamp-mss-to-pmtu>`

Задает значение TCP MSS для транзитного IPv4 трафика на определенном сетевом интерфейсе. Возможны значения в диапазоне 536 - 65535 байт.

Вместо задания числового значения TCP MSS вручную вы можете установить флаг clamp-mss-to-pmtu. Тогда значение TCP MSS будет определено системой автоматически на основе результатов работы механизма Path MTU Discovery.

Фиксация для IPv6

- `set firewall options interface <interface> adjust-mss6 <mss | clamp-mss-to-pmtu>`

Задает значение TCP MSS для транзитного IPv6 трафика на определенном интерфейсе. Возможны значения в диапазоне 1220 - 65535 байт.

Вместо задания числового значения TCP MSS вручную вы можете установить флаг clamp-mss-to-pmtu. Тогда значение TCP MSS будет определено системой автоматически на основе результатов работы механизма Path MTU Discovery.

Политика фильтрации на основе зон

Общая информация о работе политике фильтрации на основе зон

В концепции политики фильтрации на основе зон интерфейсы назначаются зонам, а политика проверки применяется к трафику, проходящему между зонами, и действует в соответствии с правилами межсетевого экрана. Зона — это группа интерфейсов, имеющих схожие функции или возможности. Она устанавливает границы безопасности сети. Зона определяет границу, на которую накладываются ограничения политики при переходе трафика в другую область сети.

Ключевые понятия:

- Зона должна быть сконфигурирована до назначения ей интерфейса, а интерфейс может быть назначен только одной зоне.
- Весь трафик, входящий на интерфейс и исходящий с интерфейса разрешен в зоне.
- Весь трафик между зонами регулируется существующими политиками.
- Трафик не может проходить между интерфейсом, входящим в зону, и любым интерфейсом, не входящим в нее.
- Для определения движения трафика необходимы два отдельных межсетевых экрана - по одному на каждое направление движения.

Далее представлен пример с предоставлением доступа по протоколу SSH к маршрутизатору с установленным ПО **Факел**.

Пример настройки политики фильтрации на основе зон

Пример создания правил межсетевого экрана для предоставления доступа по протоколу SSH к маршрутизатору с установленным ПО **Факел**.

Создание правил межсетевого экрана:

```
set firewall name lan-local default-action 'drop'
set firewall name lan-local rule 1 action 'accept'
set firewall name lan-local rule 1 state established 'enable'
set firewall name lan-local rule 1 state related 'enable'
set firewall name lan-local rule 2 action 'drop'
set firewall name lan-local rule 2 state invalid 'enable'
```

```
set firewall name lan-local rule 2 log enable
set firewall name lan-local rule 100 action 'accept'
set firewall name lan-local rule 100 destination port '22'
set firewall name lan-local rule 100 log 'enable'
set firewall name lan-local rule 100 protocol 'tcp'
set firewall name local-lan default-action 'drop'
set firewall name local-lan rule 1 action 'accept'
set firewall name local-lan rule 1 state established 'enable'
set firewall name local-lan rule 1 state related 'enable'
set firewall name local-lan rule 2 action 'drop'
set firewall name local-lan rule 2 state invalid 'enable'
set firewall name local-lan rule 2 log enable
set firewall name local-lan rule 100 action 'accept'
set firewall name local-lan rule 100 destination address
'192.168.0.0/24'
set firewall name local-lan rule 100 log 'enable'
set firewall name local-lan rule 100 protocol 'tcp'
```

Настройка политики фильтрации на основе зон:

```
set zone-policy zone lan default-action 'drop'
set zone-policy zone lan description 'Local Area Network'
set zone-policy zone lan interface 'eth2'
set zone-policy zone lan from local firewall name 'lan-local'
set zone-policy zone local default-action 'drop'
set zone-policy zone local description 'system-defined zone'
set zone-policy zone local from lan firewall name 'local-lan'
set zone-policy zone local local-zone
```

Создание зоны

Прежде, чем применить набор правил к зоне, необходимо ее создать путем определения состава сетевых интерфейсов, входящих в нее.

```
set zone-policy zone <name> interface <interfacenames>
```

Определяет состав сетевых интерфейсов *<interfacenames>*, входящих в зону *<name>*. Каждая зона может содержать несколько сетевых интерфейсов, но каждый сетевой интерфейс может входить только в одну зону.

```
set zone-policy zone <name> local-zone
```

Определяет зону *<name>* как локальную. Локальная зона не имеет интерфейсов и соответствует самому устройству.

```
set zone-policy zone <name> default-action [drop | reject]
```

Определяет действие по умолчанию *[drop | reject]* для зоны *<name>*.

```
set zone-policy zone <name> description <text>
```

Добавляет текстовое описание *<text>* для созданной зоны *<name>*.

Применение набора правил к зоне

```
set zone-policy zone <name-1> from <name-2> firewall name <rule-set>
```

Применяет набор правил фильтрации IPv4 трафика *<rule-set>* из одной зоны *<name-1>* в другую *<name-2>*.

```
set zone-policy zone <name> from <name> firewall ipv6-name <rule-set>
```

Применяет набор правил фильтрации IPv6 трафика *<rule-set>* из одной зоны *<name-1>* в другую *<name-2>*.

> **Примечание**
>
> *Набор правил всегда применяется к определенной зоне с учетом другой зоны, откуда ожидается трафик, то есть к паре зон. Рекомендуется применять по одному набору правил к каждой паре зон.*

Трансляция адресов

Общая информация о трансляции адресов

NAT — распространенный метод преобразования одного адресного пространства IP в другое путем модификации информации о сетевом адресе в заголовке IP пакетов во время их прохождения через устройство маршрутизации трафика. Изначально этот метод использовался для того, чтобы избежать необходимости настройки трансляции для каждого узла при перемещении сети. В настоящее время метод стал популярным и необходимым инструментом для сохранения глобального адресного пространства в

условиях исчерпания адресов IPv4. Один IP адрес NAT шлюза может быть использован для всей частной сети.

IP маскарад — это технология, которая позволяет скрыть все адресное пространство IP, обычно состоящее из частных IP адресов, за одним IP адресом в другом, обычно публичном адресном пространстве. Скрытые адреса меняются на один (публичный) IP адрес в качестве адреса отправителя исходящих IP пакетов, так что создается впечатление, что они исходят не от скрытого хоста, а от самого устройства маршрутизации. В связи с популярностью этого метода экономии адресного пространства IPv4 термин «NAT» стал практически синонимом термина «IP маскарад».

Поскольку трансляция сетевых адресов изменяет информацию об IP адресе в пакетах, реализация NAT может отличаться по своему поведению при различных вариантах адресации и влиянию на сетевой трафик. Особенности поведения NAT обычно не документируются производителями оборудования, содержащего NAT.

Хосты внутренней сети могут использовать любой из адресов, зарезервированных в реестре IANA для частной адресации (см. RFC 1918). Эти зарезервированные IP адреса не используются в Интернете, поэтому внешняя машина не сможет напрямую проложить к ним маршрут. Следующие адреса зарезервированы для частного использования:

- 10.0.0.0 - 10.255.255.255 (CIDR: 10.0.0.0/8);
- 172.16.0.0 - 172.31.255.255 (CIDR: 172.16.0.0/12);
- 192.168.0.0 - 192.168.255.255 (CIDR: 192.168.0.0/16).

Если провайдер развертывает CGN и использует адресное пространство согласно RFC 1918 для идентификации шлюзов клиентов, то есть риск возникновения коллизии адресов и, соответственно, возникновения сбоев в маршрутизации, когда в сети клиента уже используется адресное пространство согласно RFC 1918.

Это побудило некоторых провайдеров разработать в рамках консорциума ARIN политику выделения нового частного адресного пространства для CGN, но консорциум ARIN перед реализацией этой политики обратился к организации IETF, указав, что речь идет не о типичном выделении, а о резервировании адресов для технических целей уже согласно RFC 2860.

Организация IETF опубликовала RFC 6598, в котором подробно описано разделяемое адресное пространство для использования провайдерами в CGN сетях, которые могут обрабатывать одни и те же сетевые префиксы, встречающиеся как на входящих, так и на исходящих интерфейсах. После этого консорциум ARIN вернул адресное пространство в реестр IANA для этого распределения.

Данное распределение содержит адресный блок 100.64.0.0/10.

Устройства, оценивающие, является ли IPv4 адрес публичным, должны быть обновлены, чтобы распознать новое адресное пространство. Выделение большего количества частного адресного пространства IPv4 для устройств, выполняющих функцию NAT, может замедлить переход на IPv6.

Типы трансляции адресов

Трансляция адреса отправителя (SNAT)

SNAT является наиболее распространенной формой NAT и обычно обозначается просто как NAT. Точнее то, что большинство людей называют NAT, на самом деле является процессом PAT, или перегрузкой NAT. SNAT обычно используется внутренними пользователями или хостами с частной адресацией для доступа в сеть Интернет-адрес отправителя транслируется, оставаясь таким образом скрыт для узлов в сети Интернет.

Трансляция адреса получателя (DNAT)

DNAT изменяет адрес получателя пакетов, проходящих через маршрутизатор, а Трансляция адреса отправителя (SNAT) изменяет адрес отправителя пакетов. DNAT обычно используется, когда внешний хост, располагающийся в открытой сети, должен инициировать сеанс связи с внутренним хостом, располагающимся в защищаемой сети. Клиенту необходимо получить доступ к определенному сервису в защищаемой сети за публичным IP адресом маршрутизатора. С публичным IP адресом маршрутизатора устанавливается соединение на известный порт, и таким образом для всего трафика на этот порт адрес получателя заменяется на адрес внутреннего хоста.

Двусторонняя трансляция (Bidirectional)

В данном распространенном сценарии одновременно настроены Трансляция адреса отправителя (SNAT) и Трансляция адреса получателя (DNAT). Он обычно используется в тех случаях, когда внутренним, хостам располагающимся в защищаемой сети, необходимо установить соединение с внешними ресурсами, а внешним хостам, располагающимся в открытой сети - получить доступ к внутренним ресурсам.

Наборы правил трансляции адресов

NAT, как и межсетевой экран, настраивается посредством создания набора правил. Операционная система Факел учитывает порядок следования правил, поэтому у каждого правила в наборе есть порядковый номер. Номера правил могут быть изменены с помощью команд rename и copy в режиме конфигурирования.

Примечание

Изменения, внесенные в конфигурацию NAT применяются только к вновь устанавливаемым соединениям. Установленные ранее соединения не затрагиваются.

Подсказка

При разработке набора правил NAT рекомендуется оставлять пространство между их номерами для последующего расширения набора. Например, набор может начаться с номеров 10, 20, 30 и так далее, что позволит в дальнейшем поместить новые правила между существующими.

Обычно создаются отдельные правила для трансляции адреса отправителя (SNAT) и трансляции адреса получателя (DNAT). Если необходима Двусторонняя трансляция (Bidirectional), то должны быть созданы аналогичные правила как для трансляции адреса отправителя (SNAT), так и для трансляции адреса получателя (DNAT).

Классификатор правила трансляции адресов

Классификаторы правила используются для того, что определить, к каким пакетам должны быть применены те или иные правила NAT. Для использования доступные следующие классификаторы:

- **outbound-interface** - применим только для трансляции адреса отправителя (SNAT). Позволяет указать исходящий интерфейс для трафика, к которому должны быть применены правила NAT.
- **inbound-interface** - применим только для трансляции адреса получателя (DNAT). Позволяет указать входящий интерфейс для трафика, к которому должны быть применены правила NAT.
- **protocol** - позволяет указать протоколы передачи трафика, к которому должны быть применены правила NAT. По умолчанию классификатор имеет значение all, что соответствует всем поддерживаемым протоколам.
- **source** - позволяет указать IP адрес и/или номер порта отправителя трафика, к которому должны быть применены правила NAT. Если данный классификатор не указан, правило NAT будет применяться к любому адресу и/или порту отправителя.
- **destination** - позволяет указать IP адрес и/или номер порта получателя трафика, к которому должны быть применены правила NAT. Если данный классификатор не указан, правило NAT будет применяться к любому адресу и/или порту получателя.

Преобразование адресов

В каждом правиле NAT определена команда трансляции **translation address**. Адрес, определенный для трансляции, является адресом, используемым при замене адресной информации в пакете.

Адрес отправителя

Для правил трансляции адреса отправителя (SNAT) адрес отправителя в заголовке IP пакета будет заменен на адрес, указанный в команде трансляции translation address. Также может быть указана необходимость трансляции порта, который является частью адреса трансляции.

При трансляции адреса отправителя необходимо либо указывать в качестве адреса трансляции один из доступных адресов на исходящем интерфейсе, заданном в классификаторе outbound-interface, либо использовать директиву masquerade, которая

подразумевает, что в качестве адреса трансляции будет по умолчанию использован основной адрес исходящего интерфейса.

При использовании NAT для большого количества хостов рекомендуется использовать не менее одного внешнего IP адреса на каждые 256 внутренних хостов. Это связано с ограничением в 65000 номеров портов, доступных для уникальных трансляций, и резервированием в среднем 200 - 300 сессий на один хост. Для ориентировочно 8000 хостов рекомендуется использовать пул из не менее чем 32 IP адресов, в которые необходимо выполнять трансляцию.

Пул адресов может быть задан путем использования символа - между левым и правым IP адресами.

```
set nat source rule 100 translation address '203.0.113.32-203.0.113.63'
```

Пример:

```
set nat source rule 20 translation address 100.64.0.1
set nat source rule 30 translation address 'masquerade'
set nat source rule 40 translation address 100.64.0.10-100.64.0.20
```

Адрес получателя

Для правил трансляции адреса получателя (DNAT) адрес получателя в заголовке IP пакета будет заменен на адрес, указанный в команде трансляции **translation address**.

Пример:

```
set nat destination rule 10 translation address 192.0.2.10
```

Пример настройки трансляции адресов

Трансляция адреса источника SNAT

Пример настройки правил трансляции адреса источника SNAT, для предоставления доступа из внутренних сетей LAN и DMZ к ресурсам из внешней сети WAN. Правила трансляции адресов будут заменять адрес источника из локальной сети 172.16.100.0/24 или сети DMZ 10.150.0.0/24 на адрес, настроенный на внешнем интерфейсе eth2 по типу MASQUERADE.

Список команд для настройки SNAT:

Настройка правил трансляции для источника из локальной сети

```
set nat source rule 10 description "LAN to WAN translation"
set nat source rule 10 source address '172.16.100.0/24'
set nat source rule 10 outbound-interface 'eth2'
set nat source rule 10 translation address masquerade
```

Настройка правил трансляции для источника из сети DMZ

```
set nat source rule 20 description "DMZ to WAN translation"
set nat source rule 20 source address '10.150.0.0/24'
set nat source rule 20 outbound-interface 'eth2'
set nat source rule 20 translation address masquerade
```

Трансляция адреса источника DNAT

Пример настройки правил трансляции адреса назначения DNAT, для предоставления доступа из внешней сети WAN к веб серверу из сети DMZ. Правила трансляции адресов будут заменять адрес внешнего интерфейса 87.156.23.78 на адрес веб сервера 10.150.0.78 при обращении к адресу внешнего интерфейса из сети WAN по протоколу HTTPS.

Список команд для настройки DNAT:

```
set nat destination rule 10 description "Port Forward: 443 to 10.150.0.78"
set nat destination rule 10 destination address '87.156.23.78'
set nat destination rule 10 destination port '443'
set nat destination rule 10 inbound-interface 'eth2'
set nat destination rule 10 protocol 'tcp'
set nat destination rule 10 translation address '10.150.0.78'
set nat destination rule 10 translation options address-mapping 'persistent'
```

В разделе **Примеры конфигураций NAT** представлено описание других примеров для настройки трансляции NAT в операционной системе Факел.

Основные настройки трансляции адресов

```
set nat source rule <rule>
```

Создает правило трансляции <*rule*> типа SNAT.

- `set nat source rule <rule> description <text>`

Задает описание <*text*> для правила трансляции <*rule*> типа SNAT.

- `set nat source rule <rule> source address <address>`

Определяет адрес <*address*> источника для правила трансляции <*rule*> типа SNAT.

- `set nat source rule <rule> outbound-interface <ethN>`

Определяет исходящий интерфейс <*ethN*> для правила трансляции <*rule*> типа SNAT.

- `set nat source rule <rule> translation address masquerade`

Устанавливает тип трансляции *masquerade* для правила трансляции <*rule*> типа SNAT.

- `set nat source rule <rule> translation address <x.x.x.x>`

Определяет адрес <*x.x.x.x*>, который будет использован для трансляции в правиле <*rule*> типа SNAT.

- `set nat source rule <rule> translation address <x.x.x.x>-<x.x.x.x>`

Определяет диапазон адресов <*x.x.x.x*>-<*x.x.x.x*>, который будет использован для трансляции в правиле <*rule*> типа SNAT.

- `set nat source rule <rule> translation address <x.x.x.x/x>`

Определяет подсеть <*x.x.x.x/x*>, который будет использован для трансляции в правиле <*rule*> типа SNAT.

- `set nat source rule <rule> log <enable|disable>`

Включает/отключает запись в журнал событий <*enable*|*disable*> правила фильтрации SNAT <***rule>***.

- `set nat source rule <rule> disable`

Отключает правило фильтрации SNAT <*rule*>. Данная команда не удаляет правило из конфигурации операционной системы. После отключения правило остается в конфигурации операционной системы.

- `set nat destination rule <rule>`

Создает правило трансляции <*rule*> типа DNAT.

- `set nat destination rule <rule> description <text>`

Задает описание <*text*> для правила трансляции <*rule*> типа DNAT.

- `set nat destination rule <rule> destination address <address>`

Определяет адрес <*address*> получателя для правила трансляции <*rule*> типа DNAT.

- `set nat destination rule <rule> destination port <port>`

Определяет порт <*port*> получателя для правила трансляции <*rule*> типа DNAT.

- `set nat destination rule <rule> inbound-interface <ethN>`

Определяет входящий интерфейс <*ethN*> для правила трансляции <*rule*> типа DNAT.

- `set nat destination rule <rule> protocol <tcp|udp|tcp_udp>`

Определяет протокол <*tcp|udp|tcp_udp*> для правила трансляции <*rule*> типа DNAT.

- `set nat destination rule <rule> translation address <address>`

Определяет адрес <*address*>, который будет использован для трансляции в правиле <*rule*> типа DNAT.

- `set nat destination rule <rule> translation options address-mapping <persistent|random>`

Определяет тип механизма сопоставления адресов <*persistent|random*> для трансляции в правиле <*rule*> типа DNAT.

- `set nat destination rule <rule> log <enable|disable>`

Включает/отключает запись в журнал событий <*enable|disable*> правила фильтрации DNAT <*rule*>.

- `set nat destination rule <rule> disable`

Отключает правило фильтрации DNAT <*rule*>. Данная команда не удаляет правило из конфигурации операционной системы. После отключения правило остается в конфигурации операционной системы.

Мониторинг работы трансляции адресов

- `show nat source statistics`

Выводит на экран статистику работы механизма SNAT.

```
show nat source rules
```

Выводит на экран список настроенных правил SNAT.

```
show nat source translations
```

Выводит на экран информацию об активных трансляциях типа SNAT.

```
show nat destination statistics
```

Выводит на экран статистику работы механизма DNAT.

```
show nat destination rules
```

Выводит на экран список настроенных правил DNAT.

```
show nat destination translations
```

Выводит на экран информацию об активных трансляциях типа DNAT.

NAT Reflection

Распространенной проблемой при использовании NAT и размещении публичных серверов является возможность обращения внутренних хостов к внутреннему серверу по его внешнему IP адресу. Решением этой проблемы обычно является использование подхода **Split DNS** для корректного указания хостам внутреннего адреса при внутренних запросах. Однако в сетях небольшого масштаба зачастую отсутствует инфраструктура DNS, поэтому для управления трафиком используется обходной путь: выполняется трансляция адреса отправителя запросов от внутренних хостов в адрес внутреннего интерфейса на межсетевом экране.

Такой подход называется обратной трансляцией (NAT Reflection, Hairpin NAT).

Пример NAT Reflection:

- Трафик по протоколу RDP, поступающий на внешний (WAN) интерфейс из открытой сети перенаправляется с помощью трансляции адреса получателя (DNAT), заданной в правиле с номером *100* внутреннему хосту с адресом *192.0.2.40* в защищаемой сети.
- Трафик по протоколу RDP, поступающий на внутренний (LAN) интерфейс из защищаемой сети перенаправляется с помощью трансляции адреса получателя (DNAT), заданной в правиле с номером *110* внутреннему хосту с адресом *192.0.2.40* в защищаемой сети. Для трафика в обратную сторону также необходимо предусмотреть трансляцию адреса отправителя (SNAT) в правиле с

номером *110*. Адрес защищаемой сети - *192.0.2.0/24*. Защищаемая сеть доступна через интерфейс *eth0.10*.

Список команд для настройки обратной трансляции:

```
set nat destination rule 100 description 'Regular destination NAT from external'
set nat destination rule 100 destination port '3389'
set nat destination rule 100 inbound-interface 'pppoe0'
set nat destination rule 100 protocol 'tcp'
set nat destination rule 100 translation address '192.0.2.40'
set nat destination rule 110 description 'NAT Reflection: INSIDE'
set nat destination rule 110 destination port '3389'
set nat destination rule 110 inbound-interface 'eth0.10'
set nat destination rule 110 protocol 'tcp'
set nat destination rule 110 translation address '192.0.2.40'
set nat source rule 110 description 'NAT Reflection: INSIDE'
set nat source rule 110 destination address '192.0.2.0/24'
set nat source rule 110 outbound-interface 'eth0.10'
set nat source rule 110 protocol 'tcp'
set nat source rule 110 source address '192.0.2.0/24'
set nat source rule 110 translation address 'masquerade'
```

В результате выполнения данных команд будет создана следующая конфигурация:

```
fakel@fakel# show nat
destination {
   rule 100 {
      description "Regular destination NAT from external"
      destination {
         port 3389
      }
      inbound-interface pppoe0
      protocol tcp
```

```
      translation {
         address 192.0.2.40
      }
   }
   rule 110 {
      description "NAT Reflection: INSIDE"
      destination {
         port 3389
      }
      inbound-interface eth0.10
      protocol tcp
      translation {
         address 192.0.2.40
      }
   }
}
source {
   rule 110 {
      description "NAT Reflection: INSIDE"
      destination {
         address 192.0.2.0/24
      }
      outbound-interface eth0.10
      protocol tcp
      source {
         address 192.0.2.0/24
      }
      translation {
```

```
            address masquerade
        }
    }
}
```

Трансляция адреса получателя (DNAT)

Трансляция адреса получателя (DNAT) обычно связана также с трансляцией портов. При использовании устройства под управлением операционной системы Факел в качестве маршрутизатора с функцией NAT и межсетевого экрана как правило стоит задача перенаправить входящий трафик на хост, защищаемый межсетевым экраном.

Для конфигурирования трансляции адреса получателя (DNAT) необходимо определить:

- На каком интерфейсе ожидается поступление трафика;
- Какой протокол и какой номер порта необходимо перенаправлять;
- На какой IP адрес внутреннего хоста необходимо перенаправить трафик.

В данном примере необходимо выполнить трансляцию HTTP трафика на внутренний веб-сервер по адресу *192.168.0.100*. HTTP трафик подразумевает использование транспортного протокола TCP с номером порта 80. Перед использование других протоколов и номеров портов рекомендуется ознакомиться с реестром IANA Port Number Registry.

Список команд для настройки DNAT:

```
set nat destination rule 10 description 'Port Forward: HTTP to 192.168.0.100'
set nat destination rule 10 destination port '80'
set nat destination rule 10 inbound-interface 'eth0'
set nat destination rule 10 protocol 'tcp'
set nat destination rule 10 translation address '192.168.0.100'
```

В результате выполнения данных команд будет создана следующая конфигурация:

```
nat {
    destination {
        rule 10 {
            description "Port Forward: HTTP to 192.168.0.100"
```

```
            destination {
               port 80
            }
            inbound-interface eth0
            protocol tcp
            translation {
               address 192.168.0.100
            }
         }
      }
   }
```

Примечание

В случае перенаправления трафика на порт отличный от того, на который он поступил изначально, необходимо также сконфигурировать трансляцию портов с помощью команды set nat destination rule <number> translation port.

Таким образом в результате мы получим правило с трансляцией портов, однако если была задана политика межсетевого экрана, то трафик, скорее всего, будет заблокирован.

При задании правил межсетевого экрана важно понимать, что трансляция адреса получателя (DNAT) будет выполнена прежде, чем трафик пройдет через межсетевой экран.

Другими словами, адрес получателя уже был изменен на *192.168.0.100*, так что теперь необходимо разрешить трафик, поступающий на внешний (WAN) интерфейс и TCP порт с номером 80, а также предназначенный для адреса *192.168.0.100*.

Список команд для настройки правил межсетевого экрана:

```
set firewall name OUTSIDE-IN rule 20 action 'accept'
set firewall name OUTSIDE-IN rule 20 destination address
'192.168.0.100'
set firewall name OUTSIDE-IN rule 20 destination port '80'
set firewall name OUTSIDE-IN rule 20 protocol 'tcp'
set firewall name OUTSIDE-IN rule 20 state new 'enable'
```

В результате выполнения данных команд будет создана следующая конфигурация:

```
rule 20 {
   action accept
   destination {
      address 192.168.0.100
      port 80
   }
   protocol tcp
   state {
      new enable
   }
}
```

Примечание

Изменения, внесенные в конфигурацию NAT применяются только к вновь устанавливаемым соединениям. Установленные ранее соединения не затрагиваются.

Если ранее уже была задана некоторая политика *INSIDE-OUT*, то необходимо добавить дополнительные правила для разрешения входящего трафика после трансляции адресов (NAT).

Трансляция «один-к-одному»

Другим понятием, часто ассоциируемым с трансляцией адреса получателя (DNAT), является трансляция «один-к-одному». Для этого необходимо использовать одновременно трансляцию адреса получателя (DNAT) и адреса отправителя (SNAT) для всего трафика от внешнего IP адреса к внутреннему IP адресу и наоборот.

Обычно при трансляции «один-к-одному» порт получателя не задается (рассматриваются все возможные порты), а в качестве используемого протокола указывается либо значение all, либо ip.

Затем необходимо создать соответствующее правило трансляции адреса отправителя (SNAT) для исходящего трафика от внутреннего IP адреса к внешнему. Таким образом выделяется определенный внешний IP адрес для определенного внутреннего, что может быть необходимо в случае использования протоколов без портов (например, GRE).

Далее представлен простой пример конфигурации трансляции «один-к-одному» с одним внутренним и одним внешним интерфейсом.

Список команд для настройки трансляции «один-к-одному»:

```
set interfaces ethernet eth0 address '192.168.1.1/24'
set interfaces ethernet eth0 description 'Inside interface'
set interfaces ethernet eth1 address '192.0.2.30/24'
set interfaces ethernet eth1 description 'Outside interface'
set nat destination rule 2000 description '1-to-1 NAT example'
set nat destination rule 2000 destination address '192.0.2.30'
set nat destination rule 2000 inbound-interface 'eth1'
set nat destination rule 2000 translation address
'192.168.1.10'
set nat source rule 2000 description '1-to-1 NAT example'
set nat source rule 2000 outbound-interface 'eth1'
set nat source rule 2000 source address '192.168.1.10'
set nat source rule 2000 translation address '192.0.2.30'
```

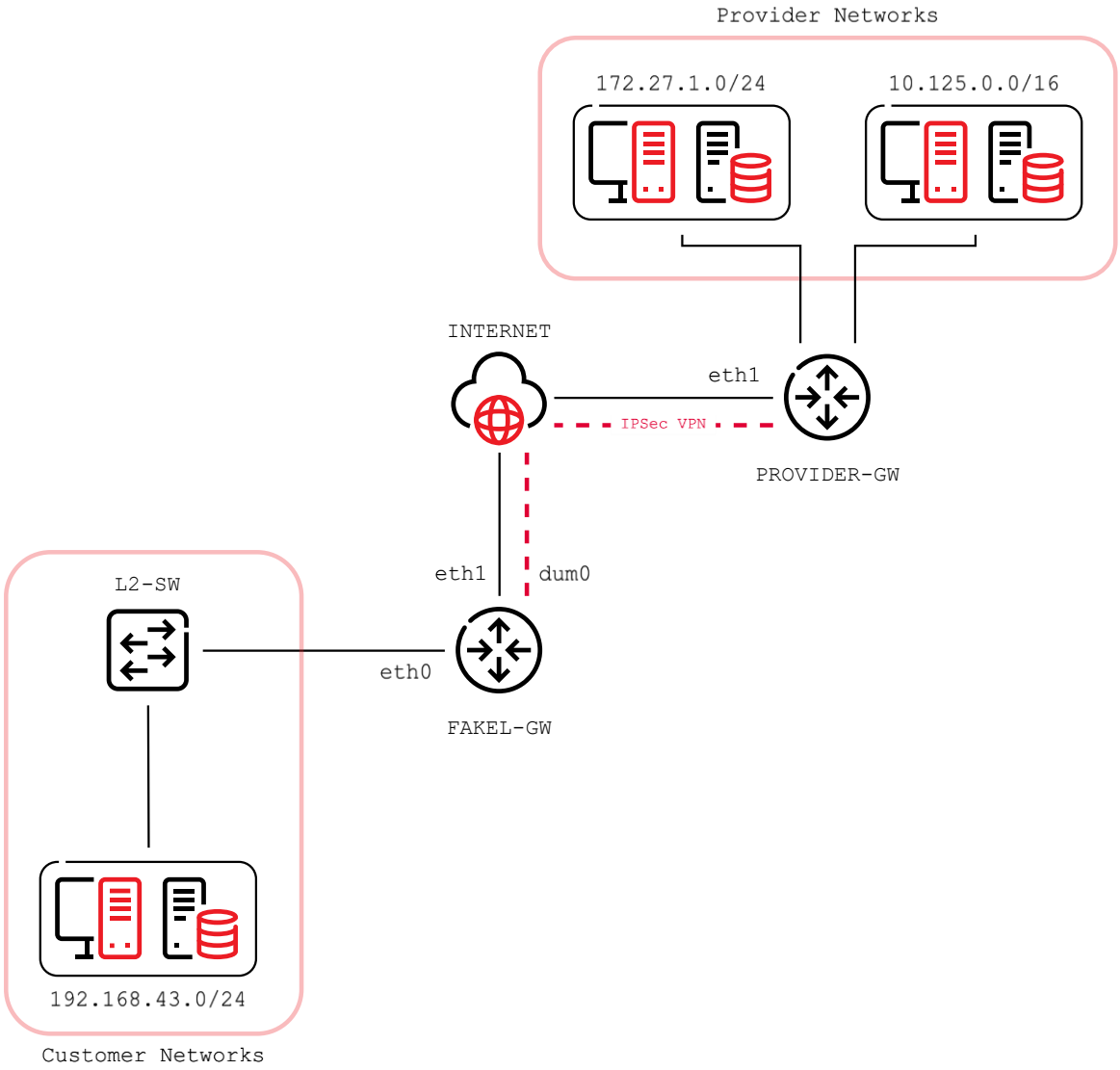
Правила межсетевого экрана задаются обычным образом с указанием внешних IP адресов в качестве адресов отправителей для исходящих правил и адресов получателей для входящий правил.

Трансляция перед VPN

Некоторые провайдеры приложений (ASP) используют VPN шлюз для обеспечения доступа к своим внутренним ресурсам и требуют, чтобы подключающаяся организация выполняла трансляцию адресов для всего трафика, направленного в сеть поставщика услуг, в адрес отправителя, предоставленного ASP.

Сетевая инфраструктура

Далее представлен пример сетевого окружения для ASP. По требованиям ASP все подключения от компании должны осуществляться с адреса 172.29.41.89 - адреса, который назначен на стороне ASP и не используется на клиентской стороне.



Конфигурацию следует рассматривать в разрезе следующих основных составляющих:

- *192.168.43.0/24* – адрес пользовательской сети за маршрутизатором *FAKEL-GW*;
- *172.21.1.0/24* – адрес сети провайдера за маршрутизатором *PROVIDER-GW*;
- *10.125.0.0/16* – адрес сети провайдера за маршрутизатором *PROVIDER-GW*;
- *192.168.43.1* – IP-адрес внутреннего интерфейса *eth0* на маршрутизаторе *FAKEL-GW*;
- *200.0.113.46* – IP-адрес внешнего интерфейса *eth1* на маршрутизаторе *FAKEL-GW*;
- *172.29.41.89* – IP-адрес виртуального интерфейса *dum0* на маршрутизаторе *FAKEL-GW*;
- *198.51.100.243* – IP-адрес внешнего интерфейса *eth1* на маршрутизаторе *PROVIDER-GW*;

- использования интерфейса Dummy для назначенного провайдером IP адреса;
- использования трансляции адресов (NAT), особенно трансляции адреса отправителя (SNAT);
- использования групп алгоритмов для протоколов IKE и ESP в рамках стека протоколов IPSec;
- использования туннелей IPSec VPN.

Интерфейс Dummy представляет собой некоторый аналог интерфейса Loopback в операционной системе Cisco IOS - внутренний интерфейс, которому можем быть назначен IP адрес, но который в действительности никак не соотносится с реальной сетью.

Для настройки интерфейса Dummy необходимо выполнить следующую команду:

```
set interfaces dummy dum0 address '172.29.41.89/32'
```

Список команд для настройки трансляции адресов:

```
set nat source rule 110 description 'Internal to ASP'
set nat source rule 110 destination address '172.27.1.0/24'
set nat source rule 110 outbound-interface 'any'
set nat source rule 110 source address '192.168.43.0/24'
set nat source rule 110 translation address '172.29.41.89'
set nat source rule 120 description 'Internal to ASP'
set nat source rule 120 destination address '10.125.0.0/16'
set nat source rule 120 outbound-interface 'any'
set nat source rule 120 source address '192.168.43.0/24'
set nat source rule 120 translation address '172.29.41.89'
```

Конфигурация IPSec (IKE и ESP)

Далее представлены группы алгоритмов, которые необходимо использовать по требованиям ASP.

Для протокола IKE:

- шифрование с помощью AES-256;
- контроль целостности с помощью SHA-256.

Для протокола ESP:

- шифрование с помощью AES-256;

- контроль целостности с помощью SHA-256;
- выработка секретного ключа с помощью DH Group 14.

Дополнительно необходимо указать, что построение туннелей будет выполняться только с внешнего интерфейса *eth1* согласно схеме, представленной выше.

Список команд для настройки IKE и ESP групп:

```
set vpn ipsec ike-group my-ike ikev2-reauth 'no'
set vpn ipsec ike-group my-ike key-exchange 'ikev1'
set vpn ipsec ike-group my-ike lifetime '7800'
set vpn ipsec ike-group my-ike proposal 1 dh-group '14'
set vpn ipsec ike-group my-ike proposal 1 encryption 'aes256'
set vpn ipsec ike-group my-ike proposal 1 hash 'sha256'
set vpn ipsec esp-group my-esp compression 'disable'
set vpn ipsec esp-group my-esp lifetime '3600'
set vpn ipsec esp-group my-esp mode 'tunnel'
set vpn ipsec esp-group my-esp pfs 'disable'
set vpn ipsec esp-group my-esp proposal 1 encryption 'aes256'
set vpn ipsec esp-group my-esp proposal 1 hash 'sha256'
set vpn ipsec ipsec-interfaces interface 'eth1'
```

Конфигурация IPSec (туннели)

Для данной сети VPN будут использоваться группы алгоритмов для протоколов IKE и ESP, представленные выше. Так как доступ необходим к двум различным подсетям на удаленной стороне, понадобятся два разных туннеля. При выборе на предыдущем шаге групп алгоритмов, отличных от представленных в примере, убедитесь, что далее будут использоваться те же самые группы.

Список команд для настройки IKE и ESP групп:

```
set vpn ipsec site-to-site peer 198.51.100.243 authentication
mode 'pre-shared-secret'
set vpn ipsec site-to-site peer 198.51.100.243 authentication
pre-shared-secret 'PASSWORD IS HERE'
set vpn ipsec site-to-site peer 198.51.100.243 connection-type
'initiate'
set vpn ipsec site-to-site peer 198.51.100.243 default-esp-
group 'my-esp'
```

```
set vpn ipsec site-to-site peer 198.51.100.243 ike-group 'my-ike'
set vpn ipsec site-to-site peer 198.51.100.243 ikev2-reauth 'inherit'
set vpn ipsec site-to-site peer 198.51.100.243 local-address '203.0.113.46'
set vpn ipsec site-to-site peer 198.51.100.243 tunnel 0 local prefix '172.29.41.89/32'
set vpn ipsec site-to-site peer 198.51.100.243 tunnel 0 remote prefix '172.27.1.0/24'
set vpn ipsec site-to-site peer 198.51.100.243 tunnel 1 local prefix '172.29.41.89/32'
set vpn ipsec site-to-site peer 198.51.100.243 tunnel 1 remote prefix '10.125.0.0/16'
```

Проверка работы

Если все предыдущие шаги выполнены, то теперь можно убедиться в корректности работы системы согласно заданной конфигурации.

Стоит начать с проверки записей в таблице SA.

```
fakel@fakel:~$ show vpn ipsec sa
Peer ID / IP                            Local ID / IP
--------------                          -------------
198.51.100.243                          203.0.113.46

Tunnel  State  Bytes Out/In  Encrypt  Hash    NAT-T  A-Time  L-Time  Proto
------  -----  ------------  -------  ------  -----  ------  ------  ----
0       up     0.0/0.0       aes256   sha256  no     1647    3600    all
1       up     0.0/0.0       aes256   sha256  no     865     3600    all
```

Из представленных данных видно, что два туннеля успешно установлены и функционируют.

NAT для IPv6 (NPTv6)

NPTv6 - разновидность трансляции адресов (NAT) для протокола IPv6. Данная технология описана в спецификации RFC 6296.

Использование NPTv6 может быть полезным при использовании протокола IPv6 и нескольких подключения к провайдерам. Также NPTv6 обычно используется, когда внешний префикс IPv6 является динамическим, так как исключает необходимость настройки трансляции для внутренних узлов при изменении внешнего префикса.

Пример настройки NAT для IPv6

В качестве примера рассмотрим следующую конфигурацию сети:

- *eth0* - внутренний интерфейс *LAN*;
- *eth1* - внешний интерфейс *WAN-1* с префиксом *2001:db8:e1::/48* и соответствующей маршрутизацией;
- *eth2* - внешний интерфейс *WAN-2* с префиксом *2001:db8:e2::/48* и соответствующей маршрутизацией.

В данной конфигурации может возникнуть ряд вопросов:

- Почему при обработке трафика осуществляется выбор в пользу префикса 2001:db8:e1::/48 вместо префикса 2001:db8:e2::/48
- Что произойдет при подключении к новому провайдеру с другой маршрутизируемой IPv6 подсетью

Ответом на эти вопросы может стать назначение хостам уникального адреса ULA и настройка трансляции префиксов в соответствующую подсеть при прохождении трафика от хостов через маршрутизатор.

- *fc00:dead:beef::/48* – адрес внутренней подсети *LAN*;
- *2001:db8:e1::/48* – адрес внешней подсети *WAN-1*;
- *2001:db8:e2::/48* – адрес внешней подсети *WAN-2*;
- *fc00:dead:beef::1/48* – IP-адрес интерфейса *eth0*;
- *2001:db8:e1::1/48* – IP-адрес интерфейса *eth1*;
- *2001:db8:e2::1/48* – IP-адрес интерфейса *eth2*.

В ПО **Факел** настройка NPTv6 выполняется в секции nat nptv6.

Список команд для настройки трансляции адресов:

```
set nat nptv6 rule 10 source prefix 'fc00:dead:beef::/48'
```

```
set nat nptv6 rule 10 outbound-interface 'eth1'
set nat nptv6 rule 10 translation prefix '2001:db8:e1::/48'
set nat nptv6 rule 20 source prefix 'fc00:dead:beef::/48'
set nat nptv6 rule 20 outbound-interface 'eth2'
set nat nptv6 rule 20 translation prefix '2001:db8:e2::/48'
```

Результатом выполнения команд, представленных выше, является следующие правила межсетевого экрана (ip6tables):

```
Chain VYOS_DNPT_HOOK (1 references)
   pkts  bytes  target  prot  opt  in    out  source    destination
   0     0      NETMAP  all        eth1  any  anywhere  2001:db8:e1::/48  to:fc00:dead:beef::/48
   0     0      NETMAP  all        eth2  any  anywhere  2001:db8:e2::/48  to:fc00:dead:beef::/48
   0     0      RETURN  all        any   any  anywhere  anywhere
Chain VYOS_SNPT_HOOK (1 references)
   pkts  bytes  target  prot  opt  in   out   source              destination
   0     0      NETMAP  all        any  eth1  fc00:dead:beef::/48  anywhere     to:2001:db8:e1::/48
   0     0      NETMAP  all        any  eth2  fc00:dead:beef::/48  anywhere     to:2001:db8:e2::/48
   0     0      RETURN  all        any  any   anywhere            anywhere
```

Основные настройки NAT для IPv6

```
set nat nptv6 rule <rule>
```

Создает правило трансляции <*rule*> типа NPTv6.

```
set nat nptv6 rule <rule> description <text>
```

Задает описание <*text*> для правила трансляции <*rule*> типа NPTv6.

```
set nat nptv6 rule <rule> outbound-interface <ethN>
```

Определяет исходящий интерфейс <*ethN*> для правила трансляции <*rule*> типа NPTv6.

```
set nat nptv6 rule <rule> source prefix <h:h:h:h:h:h:h:h/x>
```

Задает префикс <*h:h:h:h:h:h:h:h/x*> источника для правила трансляции <*rule*> типа NPTv6.

```
set nat nptv6 rule <rule> translation prefix
<h:h:h:h:h:h:h:h/x>
```

Определяет префикс <*h:h:h:h:h:h:h:h/x*>, который будет использован для трансляции в правиле <*rule*> типа NPTv6.

```
set nat nptv6 rule <rule> disable
```

Отключает правило фильтрации NPTv6 <*rule*>. Данная команда не удаляет правило из конфигурации операционной системы. После отключения правило остается в конфигурации операционной системы.

Приоритезация трафика (QoS)

Понятие QoS включает, как правило, шейпинг (ограничение и распределение пропускной способности) трафика, распределение пакетов по очередям или их отбрасывание, которые необходимы, например, для устранения узких мест в канале путем установления более высокого приоритета для определенного класса трафика перед другим классом.

Утилита tc представляет собой мощный инструмент для использования механизма QoS на уровне ядра операционной системы под управлением ПО **Факел**. Однако работа с ней напрямую вызывает затруднения у неподготовленного администратора. Поэтому для работы с ним предусмотрен специальный набор команд в CLI.

Общие принципы работы QOS

Для того, чтобы механизм QoS в составе ПО **Факел** функционировал корректно, необходимо выполнить следующие шаги:

- создать политику обработки трафика;
- применить политику обработки трафика к входящему или исходящему интерфейсу.

Перед изучением возможностей настройки политики необходимо разобраться с единицами измерения трафика, а также с понятием класса трафика.

Единицы измерения QOS

При настройке политики обработки трафика необходимо задавать скорость передачи данных. При этом следует выбирать корректные единицы измерения для соответствия фактического результата работы ожидаемому. Операционная система Факел оперирует различными префиксами и суффиксами единиц измерения, описание которых представлено в разделах ниже.

Префиксы

Префиксы могут указываться как в десятичном в формате, так и в двоичном формате.

```
kbit   (10^3)       kilobit per second

mbit   (10^6)       megabit per second

gbit   (10^9)       gigabit per second

tbit   (10^12)      terabit per second


kbps   (8*10^3)     kilobyte per second
```

```
mbps   (8*10^6)     megabyte per second
gbps   (8*10^9)     gigabyte per second
tbps   (8*10^12)    terabyte per second


      kibit (2^10 = 1024)      kibibit per second
      mibit (2^20 = 1024^2)    mebibit per second
      gibit (2^30 = 1024^3)    gibibit per second
tbit   (2^40 = 1024^4)    tebibit per second


      kibps (1024*8)           kibibyte (KiB) per second
      mibps (1024^2*8)         mebibyte (MiB) per second
gibps (1024^3*8)         gibibyte (GiB) per second
tibps (1024^4*8)         tebibyte (TiB) per second
```

Суффиксы

Суффиксы могут указываться как в единицах бит (для краткого обозначения используется bit), так и в единицах байт (для обозначения используется b).

```
kbit (kilobits per second)
mbit (megabits per second)
gbit (gigabits per second)
      tbit (terabits per second)


kbps (kilobytes per second)
mbps (megabytes per second)
gbps (gigabytes per second)
```

Классы QOS

Политики QOS позволяют классифицировать трафик в соответствии с заданными параметрами. Таким образом, класс — это определенный тип трафика, который можно указывать в политике.

В общем случае классификация трафика выполняется для обеспечения корректной его обработки.

Классификация трафика QOS

Для того чтобы определить, какой трафик ассоциируется с тем или иным классом, определяются критерии соответствия (классификатор). Пакет ассоциируется с определенным классом в том случае, если обнаруживается правило с подходящим классификатором (аналогично тому, как функционирует межсетевой экран).

В ПО **Факел** каждый класс обладает числовым идентификатором, который задается при его настройке.

> **Примечание**
>
> *Необходимо понимать, что назначение одного и того же идентификатора класса может быть разным для разных типов политики обработки трафика. Обычно при задании политики требуется указать числовой идентификатор класса, значение которого само по себе не несет никакого смысла, однако это не является обязательным условием для всех политик. Числовой идентификатор класса в очереди определенного приоритета не только идентифицирует сам класс, но также определяет приоритет.*

```
set traffic-policy <policy> <policy-name> class <class-ID>
match <class-matching-rule-name>
```

В приведенной выше команде мы задаем тип политики, с которой будем работать, и выбранное для нее имя; класс (чтобы можно было различать некоторый трафик) и идентифицируемый номер для этого класса; затем мы настраиваем правило соответствия (или фильтр) и имя для него.

В представленной выше команде задается тип политики обработки трафика, имя, класс (для дифференциации определенного трафика) и идентификатор данного класса. Затем для правила задается классификатор и имя правила.

Класс может иметь несколько классификаторов.

```
set traffic-policy shaper MY-SHAPER class 30 match HTTP
set traffic-policy shaper MY-SHAPER class 30 match HTTPs
```

Классификатор может содержать множественные критерии соответствия, а трафик будет считаться подошедшим, если подходят все заданные критерии одновременно.

Пример:

```
set traffic-policy shaper MY-SHAPER class 30 match HTTP ip
protocol tcp
```

```
set traffic-policy shaper MY-SHAPER class 30 match HTTP ip
source port 80
```

В данном примере показано задание классификатора для трафика по протоколу TCP с номером порта отправителя 80.

При задании классификатора доступны следующие параметры, определяющие подходящий трафик:

- **интерфейс** – имя интерфейса;
- **протокол Ethernet** – номер протокола, MAC адрес отправителя, MAC адрес получателя;
- **протокол IPv4** – метка DSCP, максимальная длина пакета, номер протокола, IP адрес отправителя, IP адрес получателя, номер порта отправителя, номер порта получателя, флаги TCP;
- **протокол IPv6** – метка DSCP, максимальная длина тела пакета, номер протокола, IP адрес отправителя, IP адрес получателя, номер порта отправителя, номер порта получателя, флаги TCP;
- **метка межсетевого экрана**;
- **идентификатор VLAN** – VLAN ID.

При задании классификатора можно использовать клавишу «**Tab**» для вывода на экран информации о доступных для использования параметрах.

```
fakel@fakel# set traffic-policy shaper MY-SHAPER class 30 match MY-
FIRST-FILTER

Possible completions:

   description  Description for this match

>  ether        Ethernet header match

   interface    Interface name for this match

>  ip           Match IP protocol header

>  ipv6         Match IPV6 header

   mark         Match on mark applied by firewall

   vif          Virtual Local Area Network (VLAN) ID for this match
```

Для классификатора также можно указать текстовое описание.

```
set traffic-policy shaper MY-SHAPER class 30 match MY-FIRST-FILTER
description "My filter description"
```

> **Примечание**
>
> *Классификатору с указанными параметрами протокола IPv4 (например, по протоколу TCP) будут соответствовать IPv4 пакеты с заголовком длиной 20 байт, которые, как правило, составляют большую часть трафика IPv4.*
>
> *Классификатору с указанными параметрами протокола IPv6 будут соответствовать IPv6 пакеты без расширения заголовка.*

Класс по умолчанию

Довольно часто возникает необходимость конфигурирования обработки трафика по умолчанию. Такому трафику может соответствовать и класс по умолчанию в силу характера обработки трафика. К классу по умолчанию относится любой трафик, который невозможно отнести ни к одному из других определенных классов, поэтому можно сказать, что данный класс - открытый, то есть он является классом, для которого явно не задан ни один из классификаторов.

Действие над трафиком

После того как для класса настроен фильтр, необходимо также определить, что делать с трафиком этого класса, какой специфический режим обработки трафика ему назначить. При задании разных политик обработки трафика операционной системой Факел предоставляются разные возможности.

```
fakel@fakel# set traffic-policy shaper MY-SHAPER class 30

Possible completions:

   bandwidth      Bandwidth used for this class

   burst          Burst size for this class (default: 15kb)

   ceiling        Bandwidth limit for this class

   codel-quantum  fq-codel - Number of bytes used as 'deficit' (default 1514)

   description    Description for this traffic class

   flows          fq-codel - Number of flows (default 1024)

   interval       fq-codel - Interval (milliseconds) used to measure the delay (default 100)

+> match          Class matching rule name

   priority       Priority for usage of excess bandwidth

   queue-limit    Maximum queue size (packets)

   queue-type     Queue type for this class
```

```
set-dscp       Change the Differentiated Services (DiffServ) field in the IP header

target         fq-codel - Acceptable minimum queue delay (milliseconds)
```

Например, с помощью команды ***set traffic-policy shaper MY-SHAPER class 30 set-dscp EF*** можно изменить значение метки DSCP для пакетов данного класса на Expedite Forwarding.

Допустимые значения для метки DSCP описаны в спецификациях *RFC 2474* и *RFC 4595*.

Встраивание политики

Часто возникает необходимость встроить одну политику в другую. То же самое можно сделать и в отношение политик обработки трафика, привязывая новую политику к определенному классу. Например, можно применить разные политики к разным классам сконфигурированной ранее политики Round Robin.

Типовым примером встраивания является применение ряда политик к интерфейсу, который является узким местом в контексте проходящего через него трафика. Если же маршрутизатор не подключен непосредственно к узкому месту, а находится за несколько узлов от него, то можно выполнить эмуляцию узкого место, встроив политику без шейпинга, в политику с шейпингом на основе классов, чтобы она начала действовать.

Привязать политику обработки трафика к классу можно с помощью параметра queue-type.

```
set traffic-policy shaper FQ-SHAPER bandwidth 4gbit
set traffic-policy shaper FQ-SHAPER default bandwidth 100%
set traffic-policy shaper FQ-SHAPER default queue-type fq-codel
```

Из примера выше видно, что параметр queue-type допускает различные комбинации. Также его можно использовать в нескольких политиках.

> **Примечание**
>
> *Некоторые политики уже содержат внутри себя другие встроенные политики. Такой политикой является политика «Shaper», каждый из классов которой использует политику FQ-CoDel до тех пор, пока она не будет переопределена.*

Создание политики обработки трафика

Операционная система Факел позволяет обрабатывать трафик различными способами:

- Политика **Drop Tail**;
- Политика **Fair Queue**;

- Политика **FQ-CoDel**;
- Политика **Limiter**;
- Политика **Network Emulator**;
- Политика **Priority Queue**;
- Политика **Random Detect**;
- Политика **Rate Control**;
- Политика **Round Robin**;
- Политика **Shaper**.

Можно настроить любое количество политик, но для каждого интерфейса, входящего или исходящего с учетом направления трафика, можно применить только одну политику.

Некоторые политики обработки трафика могут быть объединены с помощью встраивания, после чего встроенная политика будет применена к классу трафика, соответствующему основной политике.

> **Подсказка**
>
> *Если необходимо выбрать политику для исходящего трафика, но при этом нет понимания, какая именно политика необходима в конкретном случае, скорее всего стоит остановить выбор на политике Shaper, оставив по умолчанию встроенную в ее очереди политику FQ-CoDel.*

Если необходимо выбрать политику для исходящего трафика, но при этом нет понимания, какая именно политика необходима в конкретном случае, скорее всего стоит остановить выбор на политике Shaper, оставив по умолчанию встроенную в ее очереди политику FQ-CoDel.

Политика Drop Tail

Дисциплина очередей: First In First Out (FIFO).

Применяется к: исходящему трафику.

Очереди политики *Drop Tail* - самый простой вариант реализации, который можно применить к трафику. Перед отправкой трафик должен пройти через конечную очередь. Необходимо определить, сколько пакетов должна содержать конечная очередь.

Перед отправкой пакет должен пройти через конечную очередь, поэтому он перемещается на последнюю позицию в очереди. Когда пакет полностью пройдет через очередь, он покинет ее, освободив место, и в итоге будет передан сетевой карте для отправки.

Несмотря на то, что политика *Drop Tail* не замедляет передачу пакетов, при большом их количестве часть из них может быть отброшена при поступлении на последнюю позицию очереди. Это может произойти, если в очереди нет достаточного количества свободных места из-за того, что пакеты занимают первые позиции в очереди.

Данная политика требует наименьших ресурсов при одном и том же объеме трафика, но используется крайне редко из-за наименьшей пользы от нее при обработке трафика.

- `set traffic-policy drop-tail <policy-name> queue-limit <number-of-packets>`

Создает политику фильтрации трафика типа Drop Tail. Необходимо указать уникальное имя для данной политики, а также - размер очереди, то есть количество пакетов, которое она может содержать (максимальный размер - 4294967295).

Политика Fair Queue

Дисциплина очередей: Stochastic Fairness Queuing (SFQ).

Применяется к: исходящему трафику.

Политика *Fair Queue* подразумевает использование планировщика пакетов, который распределяет пакеты по очередям на основе потоков, то есть балансирует трафик в равной степени между дочерними очередями, чтобы пакеты из каждого потока отправлялись строго по очереди, не позволяя таким образом одному потоку преобладать над другими.

Для распределения трафика по очередям политика *Fair Queue* использует классификатор, основанный на адресе отправителя, адресе получателя и номере порта отправителя. Алгоритм распределения использует специальные хэш-корзины. Каждый пакет должен принадлежать уникальному потоку. Поскольку несколько потоков могут попасть в одну и ту же хэш-корзину, алгоритм хэширования задействуется через настраиваемые интервалы времени, поэтому преобладание одного потока над другим возможно, но только в течение некоторого времени. Однако при использовании алгоритма хэширования возможно переупорядочение пакетов. Рекомендуемое значение временного интервала составляет 10 секунд.

Один из вариантов использования политики *Fair Queue* - снижение эффективности атак типа DoS.

Список команд для настройки

- `set traffic-policy fair-queue <policy-name>`

Задает уникальное имя для политики фильтрации трафика типа Fair Queue.

- `set traffic-policy fair-queue <policy-name> hash-interval <seconds>`

Задает время в секундах, через которое будет задействован алгоритм хэширования (максимальное время - 4294967295).

> **Примечание**
>
> *При завершении очереди каждая хэш-корзина с данными используется в кольцевом порядке следования. Можно также сконфигурировать длину очереди.*

- `set traffic-policy fair-queue <policy-name> queue-limit <limit>`

Задает максимальное количество пакетов, ожидающих обработки в очереди. Любые другие пакеты будут отброшены.

> **Примечание**
>
> *Политика Fair Queue - политика без шейпинга, поэтому она будет полезна только в случае, если исходящий интерфейс действительно перегружается трафиком. В противном случае ПО Факел не будет оперировать очередью и политика не будет иметь никакого эффекта. Если в канале есть доступная для использования полоса пропускания, то можно встроить политику Fair Queue в другую политику с шейпингом на основе классов, чтобы убедиться в том, что ПО Факел оперирует очередью.*

Политика FQ-CoDel

Дисциплина очередей: Fair/Flow Queue CoDel (FQ-CoDel).

Применяется к: исходящему трафику.

Политика *FQ-CoDel* распределяет трафик по 1024 FIFO очередям и направлена на обеспечение наилучшего качества между ними. При использовании данной политики по возможности сохраняется небольшая длина очередей.

Политика *FQ-CoDel* направлена на борьбу с переполнение буфера и на снижение задержек без необходимости задания сложной конфигурации. Данная политика используется по умолчанию для интерфейсов в некоторых дистрибутивах операционной систем Linux.

Политика использует случайную модель распределения для классификации входящих пакетов на различные потоки и используется для обеспечения равных долей пропускной способности всем потокам, использующим очередь. Каждый поток обрабатывается в соответствии с дисциплиной CoDel. Переупорядочение пакетов в составе потока исключено, так как дисциплина CoDel использует внутреннюю очередь FIFO.

В основе политики *FQ-CoDel* лежит модифицированный планировщик очередей Deficit Round Robin (DRR) с алгоритмом CoDel Active Queue Management (AQM), работающим с каждой очередью.

> **Примечание**
>
> *Политика FQ-Codel - политика без шейпинга, поэтому она будет полезна только в том случае, если исходящий интерфейс действительно перегружается трафиком. В противном случае операционная система Факел не будет оперировать очередью, и политика не будет иметь никакого эффекта. Если в канале есть доступная для использования полоса пропускания, то можно встроить FQ-Codel в другую политику с шейпингом на основе классов, чтобы убедиться в том, что ПО Факел оперирует очередью. Рекомендуется встраивать политику FQ-CoDel в политику Shaper.*

Политика *FQ-CoDel* может эффективно использоваться с параметрами по умолчанию на скорости 10 Гбит/сек. Для других скоростей может потребоваться изменение параметров.

При скорости 1 Гбит/сек и ниже рекомендуется уменьшить значение параметра *queue-limit* до 1000 пакетов или меньше.

При скорости 10 Мбит/сек рекомендуется установить значение параметра *queue-limit* в 600 пакетов.

При скорости 100 Мбит/сек и выше, если политика *FQ-CoDel* встроена в политику Shaper рекомендуется установить значение параметра *quantum* в 8000 байт, чтобы снизить нагрузку на процессор.

При скорости менее 40 Мбит/сек рекомендуется установить уменьшить значение параметра *quantum* до 300 байт.

При скорости менее 3 Мбит/сек кроме задания значения параметра *quantum* в 300 байт рекомендуется также увеличить значение параметра *target* до 15 миллисекунд и значение параметра *interval* до 150 миллисекунд.

Пример использования политики FQ-CoDel, встроенной в политику Shaper:

```
set traffic-policy shaper FQ-CODEL-SHAPER bandwidth 2gbit
set traffic-policy shaper FQ-CODEL-SHAPER default bandwidth 100%
set traffic-policy shaper FQ-CODEL-SHAPER default queue-type fq-codel
```

Список команд для настройки

```
set traffic-policy fq-codel <policy name>
```

Задает уникальное имя для политики фильтрации трафика типа FQ-CoDel.

```
set traffic-policy fq-codel <policy name> codel-quantum <bytes>
```

Задает максимальное количество байт (по умолчанию - 1514), извлекаемых из очереди за один раз.

- `set traffic-policy fq-codel <policy name> flows <number-of-flows>`

Задает количество дочерних очередей (по умолчанию - 1024), в которые будут помещаться классифицированные пакеты.

- `set traffic-policy fq-codel <policy name> interval <miliseconds>`

Задает период времени в миллисекундах (по умолчанию - 100), используемый управляющим циклом CoDel для обнаружения устоявшихся очередей, чтобы исключить значительное увеличение минимально определенной задержки.

- `set traffic-policy fq-codel <policy-name> queue-limit <number-of-packets>`

Задает предельный размер очереди в пакетах (по умолчанию - 10240). При достижении предельного размера все новые пакеты отбрасываются.

- `set traffic-policy fq-codel <policy-name> target <miliseconds>`

Задает минимально допустимую задержку в миллисекундах (по умолчанию - 5) для ожидающей/устоявшейся очереди. Задержка определяется путем отслеживания минимальной задержки отдельной очереди, с которой сталкиваются проходящие через нее пакеты.

Политика Limiter

Дисциплина очередей: Ingress Policer.

Применяется к: входящему трафику.

Политика *Limiter* использует классы (на самом деле дисциплина очередей, применяемая для обработки входящего трафика, является бесклассовой политикой, но классификаторы могут успешной применяться совместно с ней).

Политика *Limiter* подразумевает простое ограничение входящих потоков трафика. Можно определить несколько классов трафика и применить к каждому классу свои ограничения. Несмотря на то, что внутри ограничителя используется механизм токен-корзин, он не имеет возможность удерживать пакеты, как это делает механизм шейпинга. Трафик, превышающий установленные ограничения полосы пропускания, отбрасывается напрямую. Также для политики *Limiter* может быть настроен максимально допустимый всплеск (burst) трафика.

Вы можете создать классы в количестве до 4090 штук с различными параметрами, а также политику по умолчанию, которая будет применена к любому трафику, не подошедшему под созданные классы.

Список команд для настройки

- `set traffic-policy limiter <policy-name>`

Задает уникальное имя для политики фильтрации трафика типа Limiter

- `set traffic-policy limiter <policy-name> class <class ID>`

Задает числовой идентификатор класса (1 - 4090).

- `set traffic-policy limiter <policy-name> class <class ID> match <match-name> description <description>`

Задает имя соответствующему классу правила, а также его текстовое описание.

- `set traffic-policy limiter <policy-name> class <class-ID> bandwidth <rate>`

Задает максимально допустимую полосу пропускания для класса.

- `set traffic-policy limiter <policy-name> class <class-ID> burst <burst-size>`

Задает размер всплеска (burst) трафика в байтах для класса (по умолчанию - 15).

- `set traffic-policy limiter <policy-name> default bandwidth <rate>`

Задает максимально допустимую полосу пропускания для политики по умолчанию.

- `set traffic-policy limiter <policy-name> default burst <burst-size>`

Задает размер всплеска (burst) трафика в байтах для политики по умолчанию (по умолчанию - 15).

- `set traffic-policy limiter <policy-name> class <class ID> priority <value>`

Задает приоритет соответствующего классу правила (0 - 20, значение по умолчанию - 20). Чем ниже значение, тем выше приоритет.

Политика Network Emulator

Дисциплина очередей: Token Bucket Filter (TBF).

Применяется к: исходящему трафику.

Политика *Network Emulator* имитирует условия эксплуатации реальной сети. При использовании данной политики есть возможность сконфигурировать такие параметры, как пропускную способность, всплеск (burst) трафика, задержку при передаче данных, потери пакетов, долю неправильных или пакетов с нарушением порядка следования.

Данная политика может быть полезна при определении поведения приложения в определенных условиях эксплуатации сети.

Список команд для настройки

- `set traffic-policy network-emulator <policy-name>`

Задает уникальное имя для политики фильтрации трафика типа Network Emulator

- `set traffic-policy network-emulator <policy-name> bandwidth <rate>`

Задает пропускную способность для политики фильтрации трафика типа Network Emulator

- `set traffic-policy network-emulator <policy-name> burst <burst-size>`

Задает размер всплеска (burst) трафика в байтах (по умолчанию - 15), который будет учитываться только при заданной пропускной способности.

- `set traffic-policy network-emulator <policy-name> network-delay <delay>`

Задает задержку в миллисекундах при прохождении пакетов через интерфейс (по умолчанию - 50), которая будет учитываться только при заданной пропускной способности.

- `set traffic-policy network-emulator <policy-name> packet-corruption <percent>`

Задает в процентном соотношении количество неправильных пакетов. На случайно позиции в структуре таких пакетов будет размещаться случайная ошибки в соответствии с заданной долей.

- `set traffic-policy network-emulator <policy-name> packet-loss <percent>`

Задает значение допустимой потери пакетов в процентах.

- `set traffic-policy network-emulator <policy-name> packet-reordering <percent>`

Задает количество пакетов, порядок следования которых нарушается, в процентах.

- `set traffic-policy network-emulator <policy-name> queue-limit <limit>`

Задает количество пакетов (1 - 4294967295), которое может содержать очередь в отдельный момент времени.

Политика Priority Queue

Дисциплина очередей: PRIO.

Применяется к: исходящему трафику.

Политика *Priority Queue* является политикой на основе классов, которая не вносит задержек в движение пакетов, так как она не является политикой с шейпингом. При использовании данной политики пакеты извлекаются из очередей в соответствии с заданным приоритетом.

> **Примечание**
>
> *Политика Priority Queue, как и любая другая политика без шейпинга, полезна только в случае, если достигнут предел пропускной способности исходящего интерфейса. Если этого не произошло, ПО **Факел** не использует очередь, и политика не будет действовать. Однако даже если физический канал располагает доступной полосой пропускания, можно встроить политику Priority Queue в другую политику с шейпингом на основе классов, чтобы убедиться в том, что ПО **Факел** оперирует очередью. В этом случае пакеты могут получать приоритеты согласно метке DSCP.*

Можно назначить до 7 очередей, определенных как классы, с различными приоритетами. Пакет помещаются в очереди в соответствии с заданным классификатором. Пакеты извлекаются из очередей в указанном порядке. Если классы с более высоким приоритетом переполняются, то пакеты из классов с приоритетом ниже, будут переданы дальше, только после снижения нагрузки на классы с более высоким приоритетом.

> **Примечание**
>
> *Политика Priority Queue предусматривает задание числового идентификатора класса для обозначения приоритета (1 - 7). Чем ниже значение, тем выше приоритет.*

Как и в случае с другими политиками, можно указать различные типы соответствующих классам правил.

```
fakel@fakel# set traffic-policy priority-queue MY-PRIO class 3 match
MY-MATCH-RULE

Possible completions:

   description  Description for this match

>  ether        Ethernet header match

   interface    Interface name for this match

>  ip           Match IP protocol header

>  ipv6         Match IPV6 header

   mark         Match on mark applied by firewall

   vif          Virtual Local Area Network (VLAN) ID for this match
```

Как и в случае с другими политиками, можно встраивать в классы (включая класс по умолчанию) политики Priority Queue другие политики с помощью параметра queue-type.

```
fakel@fakel# set traffic-policy priority-queue MY-PRIO class 3 queue-
type

Possible completions:

   fq-codel       Fair Queue Codel

   fair-queue     Stochastic Fair Queue (SFQ)

   drop-tail      First-In-First-Out (FIFO)

   priority       Priority queueing based on DSCP

   random-detect  Random Early Detection (RED)
```

Список команд для настройки

- `set traffic-policy priority-queue <policy-name> class <class-ID> queue-limit <limit>`

Создает политику *Priority Queue*. Задает уникальное имя для данной политики и числовой идентификатор класса (1 - 7). Можно указать максимально допустимый размер очереди, по достижении которого все новые пакеты отбрасываются.

Политика Random Detect

Дисциплина очередей: Random Early Drop (RED).

Применяется к: исходящему трафику.

При использовании политики *Random Detect* пакеты случайным образом отбрасываются из очереди перед тем, как она будет заполнена. Такое поведение особенно актуально для коммуникаций по протоколу TCP, так как отбрасывание пакетов может служить сигналом их отправителю, чтобы он снизил скорость передачи.

В отличие от стандартной дисциплины RED политика *Random Detect* обеспечивает различные виртуальные очереди на основе значения параметра IP Precedence таким образом, что некоторые виртуальные очереди отбрасывают большее количество пакетов, чем другие.

Это достигается путем использования первых трех бит поля **ToS** для определения потоков данных. Решение принимается в соответствии с установленным значением параметра IP Precedence, то есть установленным приоритетом.

Параметр IP Precedence определен в спецификации RFC 791.

Приоритет	Описание
7	Network Control
6	Internetwork Control
5	CRITIC / ECP
4	Flash Override
3	Flash
2	Immediate
1	Priority
0	Routine

Политика *Random Detect* может быть полезной при «тяжелых» профилях трафика, особенно на магистральных каналах, чтобы избежать их перегрузки, но только при условии, что все коммуникации используют протокол TCP (отброшенные пакеты будут переданы повторно, но с пониженной интенсивностью).

Если средний размер очереди ниже значения параметра min-threshold, поступающий пакет будет помещен в очередь.

Если средний размер очереди находится в диапазоне значений параметров min-threshold и max-threshold, поступающий пакет будет либо помещен в очередь, либо отброшен. Результат будет определен значением параметра mark-probability.

Если текущий размер очереди превышает значение параметра queue-limit, пакеты буду отбрасываться. Средний размер очереди зависит от предыдущего ее размера и текущего.

Если значение параметра max-threshold установлено, а значение параметра min-threshold - нет, последний определяется как 50% от значения параметра max-threshold.

В общем случае рекомендуется устанавливать следующие значения для данных параметров:

min-threshold < max-threshold < queue-limit.

Список команд для настройки

- `set traffic-policy random-detect <policy-name>`

Задает уникальное имя для политики фильтрации трафика типа Random Detect.

- `set traffic-policy random-detect <policy-name> bandwidth <bandwidth>`

Задает значение доступной полосы пропускания для политики фильтрации трафика типа Random Detect, значение которой будет использоваться для расчета среднего размера очереди после некоторого времени ее простоя. Рекомендуется указывать полосу пропускания в соответствии с полосой пропускания сетевого интерфейса. Политика Random Detect не является политикой с шейпингом.

- `set traffic-policy random-detect <policy-name> precedence <IP-precedence-value>`

Задает значение параметра IP Precedence для виртуальной очереди.

Примечание

Чем больше значение параметра IP Precedence, тем выше приоритет.

- `set traffic-policy random-detect <policy-name> precedence <IP-precedence-value> average-packet <bytes>`

Задает средний размер пакета в байтах (по умолчанию - 1024).

- `set traffic-policy random-detect <policy-name> precedence <IP-precedence-value> mark-probability <value>`

Задает вероятность отбрасывания пакетов путем указания значения N в распределении 1/N (по умолчанию - 10).

- `set traffic-policy random-detect <policy-name> precedence <IP-precedence-value> maximum-threshold <packets>`

Задает максимальное пороговое количество пакетов для случайного отбрасывания (0 - 4096, по умолчанию - 18). При заданном пороговом количестве вероятность отбрасывания будет максимальной.

- `set traffic-policy random-detect <policy-name> precedence <IP-precedence-value> minimum-threshold <packets>`

Задает минимальное пороговое количество пакетов для случайного отбрасывания (0 - 4096). Если заданное пороговое значение превышается, пакеты становятся кандидатами на отбрасывание.

- `set traffic-policy random-detect <policy-name> precedence <IP-precedence-value> queue-limit <packets>`

Задает максимальный размер очереди в пакетах (1 - 4294967295), при достижении которого все новые пакеты отбрасываются.

Политика Rate Control

Дисциплина очередей: Token Bucket Filter (TBF).

Применяется к: исходящему трафику.

Политика *Rate Control* является бесклассовой политикой, ограничивающей поток пакетов установленной скоростью, то есть по сути это - политика с шейпингом, которая не распределяет трафик по очередям. Обработка трафика осуществляется путем контроля расхода токенов. Токены в некоторой степени соответствуют количеству байт обработанных данных.

Допускаются кратковременные всплески трафика, которые приводят к превышению заданного лимита. При создании политика *Rate Control* накапливает токены, количество которых соответствует объему всплеска трафика при однократном проходе трафика через механизм TBF. Токены поступают с постоянной скоростью до тех пор, пока токен-корзина не будет наполнена.

Политика *Rate Control* незначительно влияет на загрузку процессора. Рекомендуется использовать данную политику, если есть необходимость замедлить движение (обработку) трафика.

Список команд для настройки

- `set traffic-policy rate-control <policy-name>`

Задает уникальное имя для политики фильтрации трафика типа Rate Control.

- `set traffic-policy rate-control <policy-name> bandwidth <rate>`

Задает лимит скорости для политики фильтрации трафика типа Rate Control.

- `set traffic-policy rate-control <policy-name> burst <burst-size>`

Задает в байтах размер токен-корзины, которая будет использоваться для обработки всплесков трафика.

> **Примечание**
>
> *Для сетевых адаптеров компании Intel с технической пропускной способностью в 10 Мбит/сек необходимо иметь буфер емкостью в 100 КБ, чтобы была возможность достичь заданной скорости. Буфер меньшей емкости довольно скоро приведет к отбрасыванию пакетов.*

- `set traffic-policy rate-control <policy-name> latency`

Задает максимально допустимую задержку в миллисекундах пакетов в очереди (по умолчанию - 50).

Политика Round Robin

Дисциплина очередей: Deficit Round Robin (DRR).

Применяется к: исходящему трафику.

Политика *Round Robin* является распределяющей политикой на основе классов, которая распределяет трафик по различным классам (максимальное количество классов - 4096). Можно встроить политику *Round Robin* в каждый из доступных классов, включая класс по умолчанию.

Каждому классу назначается специальный счетчик дефицита данных (количество байт, которое может быть передано в рамках потока при его обработке), инициализируемый параметром - квантом. Квант — это настраиваемый параметр, который представляет собой фиксированное количество байт, предоставляемых счетчику на каждом цикле его работы. Затем политика *Round Robin* начинает перемещать указатель по очередям. Если счетчик дефицита данных превышает размер пакета в начале очереди, пакет будет отправлен, а значение счетчика - уменьшено на размер пакета. Затем размер следующего пакета снова сравнивается со значением счетчика, и процесс повторяется. Как только очередь окажется пустой или текущего значения счетчика будет недостаточно для сравнения, указатель будет перемещен к следующей очереди. Если очередь пустая, то значение счетчика дефицита данных сбрасывается к нулевому значению.

На каждом цикле работы счетчик дефицита данных добавляет квант, поэтому даже большие по размеру пакеты могут быть обработаны в очереди.

Политика *Round Robin* может быть встроена в класс другой политики с помощью параметра *queue-type*.

```
fakel@fakel# set traffic-policy round-robin DRR class 10 queue-type

Possible completions:

   fq-codel     Fair Queue Codel

   fair-queue   Stochastic Fair Queue (SFQ)

   drop-tail    First-In-First-Out (FIFO)

   priority     Priority queueing based on DSCP
```

Список команд для настройки

- `set traffic-policy round-robin <policy name>`

Задает уникальное имя для политики фильтрации трафика типа Round Robin.

- `set traffic-policy round-robin <policy name> class <class-ID>`

Задает числовой идентификатор класса для политики фильтрации трафика типа Round Robin.

- `set traffic-policy round-robin <policy name> class <class-ID> quantum <packets>`

Задает квант для определенного идентификатора класса, значение которого будет добавляться на каждом цикле работы счетчика дефицита данных.

- `set traffic-policy round-robin <policy name> class <class ID> queue-limit <packets>`

Задает размер очереди в пакетах.

Политика Shaper

Дисциплина очередей: Hierarchical Token Bucket (HTB).

Применяется к: исходящему трафику.

Политика *Shaper* не гарантирует низкую задержку, но гарантирует определенную полосу пропускания для различных классов трафика, а также позволяет распределить трафик при соблюдении данных гарантий.

Каждый класс предоставляет гарантированную часть полосы пропускания, определенную для политики целиком. Другие классы оперируют оставшейся полосой пропускания, доступной в политике.

Если гарантированная полоса пропускания обеспечена, а загрузка трафиком ее не превышает, то может быть использован параметр ceiling, который определяет выделяемую долю полосы пропускания. Если гарантированная полоса пропускания обеспечена, и несколько классов используют заданный для них параметр ceiling, то можно использовать параметр priority для распределения дополнительного трафика в определенном порядке. Параметр priority представляет собой число от 0 до 7. Чем ниже значение параметра priority, тем выше приоритет.

Политика *Shaper* может быть встроена в класс другой политики с помощью параметра *queue-type*.

```
fakel@fakel# set traffic-policy shaper HTB class 10 queue-type

Possible completions:

   fq-codel       Fair Queue Codel

   fair-queue     Stochastic Fair Queue (SFQ)

   drop-tail      First-In-First-Out (FIFO)

   priority       Priority queueing based on DSCP

   random-detect  Random Early Detection (RED)


fakel@fakel# set traffic-policy shaper HTB class 10

Possible completions:

   bandwidth      Bandwidth used for this class

   burst          Burst size for this class (default: 15kb)

   ceiling        Bandwidth limit for this class

   codel-quantum  fq-codel - Number of bytes used as 'deficit' (default 1514)

   description    Description for this traffic class

   flows          fq-codel - Number of flows (default 1024)

   interval       fq-codel - Interval (milliseconds) used to measure the delay
(default 100)

+> match          Class matching rule name

   priority       Priority for usage of excess bandwidth

   queue-limit    Maximum queue size (packets)

   queue-type     Queue type for this class
```

```
   set-dscp       Change the Differentiated Services (DiffServ) field in the IP
header

   target         fq-codel - Acceptable minimum queue delay (milliseconds)
```

Примечание

Если сконфигурирован класс для трафика, источником которого является технология VoIP, не рекомендуется задавать для него параметр ceiling. В противном случае новые звонки могут не пройти из-за того, что полоса пропускания уже поделена между остальными классами.

Далее представлен пример использования политики Shaper с приоритетами.

```
set traffic-policy shaper MY-HTB bandwidth '50mbit'
set traffic-policy shaper MY-HTB class 10 bandwidth '20%'
set traffic-policy shaper MY-HTB class 10 match DSCP ip dscp
'EF'
set traffic-policy shaper MY-HTB class 10 queue-type 'fq-codel'
set traffic-policy shaper MY-HTB class 20 bandwidth '10%'
set traffic-policy shaper MY-HTB class 20 ceiling '50%'
set traffic-policy shaper MY-HTB class 20 match PORT666 ip
destination port '666'
set traffic-policy shaper MY-HTB class 20 priority '3'
set traffic-policy shaper MY-HTB class 20 queue-type 'fair-
queue'
set traffic-policy shaper MY-HTB class 30 bandwidth '10%'
set traffic-policy shaper MY-HTB class 30 ceiling '50%'
set traffic-policy shaper MY-HTB class 30 match ADDRESS30 ip
source address '192.168.30.0/24'
set traffic-policy shaper MY-HTB class 30 priority '5'
set traffic-policy shaper MY-HTB class 30 queue-type 'fair-
queue'
set traffic-policy shaper MY-HTB default bandwidth '10%'
set traffic-policy shaper MY-HTB default ceiling '100%'
set traffic-policy shaper MY-HTB default priority '7'
set traffic-policy shaper MY-HTB default queue-type 'fair-
queue'
```

Список команд для настройки

- `set traffic-policy shaper <policy-name>`

Задает уникальное имя для политики фильтрации трафика типа Shaper.

- `set traffic-policy shaper <policy-name> bandwidth <rate>`

Устанавливает гарантированную полосу пропускания для всего трафика без учета классов.

- `set traffic-policy shaper <policy-name> class <class-ID>`

Задает числовой идентификатор класса для политики фильтрации трафика типа Shaper.

- `set traffic-policy shaper <policy-name> class <class-ID> bandwidth <rate>`

Устанавливает гарантированную полосу пропускания для трафика определенного класса.

- `set traffic-policy shaper <policy-name> class <class-ID> burst <bytes>`

Задает размер токен-корзины в байтах (по умолчанию - 15 КБ), которая будет использована при передаче трафика на скорости параметра ceiling.

- `set traffic-policy shaper <policy-name> class <class-ID> ceiling <bandwidth>`

Задает максимально возможную скорость для определенного класса.

- `set traffic-policy shaper <policy-name> class <class-ID> priority <0-7>`

Устанавливает приоритет (по умолчанию - 0) использования доступной полосы пропускания при условии, что обеспечена гарантированная полоса.

Применение политики QOS

После конфигурирования политики обработки трафика ее необходимо применить к соответствующему интерфейсу:

- `set interfaces etherhet eth0 traffic-policy out WAN-OUT`

К отдельному интерфейсу и для определенного направления движения трафика может быть применена только одна политика обработки трафика, но одну и ту же политику

можно применять к различным интерфейсам и для других направлений движения трафика.

```
set interfaces ethernet eth0 traffic-policy in WAN-IN
set interfaces etherhet eth0 traffic-policy out WAN-OUT
set interfaces etherhet eth1 traffic-policy in LAN-IN
set interfaces etherhet eth1 traffic-policy out LAN-OUT
set interfaces ethernet eth2 traffic-policy in LAN-IN
set interfaces ethernet eth2 traffic-policy out LAN-OUT
set interfaces etherhet eth3 traffic-policy in TWO-WAY-POLICY
set interfaces etherhet eth3 traffic-policy out TWO-WAY-POLICY
set interfaces etherhet eth4 traffic-policy in TWO-WAY-POLICY
set interfaces etherhet eth4 traffic-policy out TWO-WAY-POLICY
```

```
show queueing <interface-type> <interface-name>
```

Выводит информацию о результатах работы политик обработки трафика, примененных к интерфейсу. Информация содержит счетчики пакетов в разрезе каждой политики и каждого сконфигурированного класса.

Обработка входящего трафика

Применяется к: входящему трафику.

Для обработки входящего трафика на интерфейсе может быть использована только одна политика - политика *Limiter*. Отсутствует возможность применения какой-либо политики с шейпингом к входящему трафику на любом интерфейсе, так как их реализация ограничивается работой только с исходящим трафиком.

Однако есть сценарий, когда можно применить политику с шейпингом к входящему трафику путем его перенаправления на специальный виртуальный интерфейс (Intermediate Functional Block). Идея состоит в том, что к виртуальному интерфейсу применяется политика, в том числе политика с шейпингом, направленная на обработку исходящего трафика.

Далее представлен пример того, как можно организовать шейпинг входящего трафика:

```
set traffic-policy shaper MY-INGRESS-SHAPING bandwidth 1000kbit
set traffic-policy shaper MY-INGRESS-SHAPING default bandwidth
1000kbit
set traffic-policy shaper MY-INGRESS-SHAPING default queue-type
fair-queue
```

```
set interfaces input ifb0 traffic-policy out MY-INGRESS-SHAPING
set interfaces ethernet eth0 redirect ifb0
```

! Предупреждение

Механизм Intermediate Functional Block необходимо настраивать только после того, как будет заданы все остальные параметры конфигурации политики обработки трафика. В противном случае возникнет ошибка RTNETLINK answer: File exists, которую можно устранить только с помощью команды sudo ip link delete ifb0.

Отказоустойчивость

В ПО **Факел** отказоустойчивость обеспечивается путем использования следующих механизмов:

- VRRP;
- Балансировка WAN-каналов.

VRRP

Общая информация о протоколе VRRP

Протокол VRRP позволяет организовать избыточность на уровне маршрутизаторов в формате активный-резервный. Протокол VRRP объединяет несколько маршрутизаторов в отказоустойчивый кластер. Все маршрутизаторы, которые являются участниками отказоустойчивого кластера, имеют общий виртуальный адрес IP (VIP) адрес и общий номер группы или VRID (Virtual Router Identifier). Один маршрутизатор может состоять в нескольких группах, каждая из которых должна иметь свою уникальную пару VIP/VRID. Все маршрутизаторы делятся на два типа: VRRP Master и VRRP Backup.

- **VRRP Master** — это маршрутизатор, который занимается пересылкой пакетов для данного виртуальной группы.
- **VRRP Backup** — это маршрутизатор, который ожидает пакет от Master. Если пакеты от Master перестают приходить, Backup пытается перейти в состояние Master.

При запуске маршрутизаторы обмениваются сообщениями, выбирая таким образом между собой активного. Активным становится тот маршрутизатор, который имеет наивысший приоритет. Виртуальный IP адрес назначается активному маршрутизатору. Все остальные маршрутизаторы с приоритетом ниже становятся резервными. В процессе работы активный маршрутизатор рассылает остальным маршрутизаторам keepalive пакеты на широковещательный адрес 224.0.0.18, чтобы сообщить Backup маршрутизаторам, что он работает. Master отправляет сообщения согласно таймеру advertise-interval, равный по умолчанию 1 секунде. В качестве MAC адреса отправителя используется адрес группы 00:00:5E:00:01:xx, где xx — VRID в шестнадцатеричном формате. Если Backup маршрутизаторы не получают сообщения в течение трех advertise-interval (количество интервалов определяется параметром failure-count), то новым Master становится маршрутизатор с наибольшим приоритетом. Если в группе появится Backup маршрутизатор с более высоким приоритетом, то он перехватит роль Master у маршрутизатора с более низким приоритетом. Однако, когда у Backup маршрутизатора, который имеет более высокий приоритет активирован параметр no-preempt, такой Backup маршрутизатор не станет перехватывать роль у Master маршрутизатора. VRRP приоритет задается в значениях от 1 до 254. Значение 0 зарезервировано для случаев, когда Master необходимо снять с себя ответственность за маршрутизацию. Значение 255 устанавливается маршрутизатору владельцу VIP.

Если VRRP-маршрутизатор является владельцем VIP адреса, то он всегда перехватывает роль Master. Приоритет по умолчанию равен 100, но может задаваться административно. VRRP маршрутизатор может иметь три состояния:

- Initialize
- Backup
- Master

VRRP маршрутизатор изменяет свои состояния последовательное.

В состоянии Initialize маршрутизатор ожидает начала работы. Если этот маршрутизатор является владельцем VIP адреса (приоритет равен 255), то маршрутизатор отправляет сообщения о том, что он становится Master. Он также отправляет gratuitous ARP-запрос, в котором MAC-адрес источника равен адресу виртуального маршрутизатора. Затем он переходит в состояние Master. Если маршрутизатор не является владельцем VIP, то он переходит в состояние Backup.

В состоянии Backup маршрутизатор ожидает пакеты от Master маршрутизатора. Маршрутизатор в этом состоянии не отвечает на ARP-запросы от VIP-адреса. Также он не принимает пакеты, у которых в качестве адреса назначения стоит MAC-адрес виртуального маршрутизатора.

Если Backup не получает сообщения от Master в течение количества advertise-interval, заданных в параметре failure-count (по умолчанию установлено значение 3 интервала), то он отправляет VRRP сообщение о том, что собирается стать Master. Затем этот маршрутизатор отправляет широковещательное VRRP сообщение, в котором MAC-адрес источника равен адресу этого виртуального маршрутизатора. В данном сообщении маршрутизатор указывает свой приоритет.

В состоянии Master маршрутизатор обрабатывает пакеты, адресованные виртуальному маршрутизатору. Он так же отвечает на ARP-запросы к VIP. Master маршрутизатор рассылает VRRP сообщения каждые advertise-interval (по умолчанию значение равно 1 секунде), чтобы подтвердить что он работает.

Если к одному сетевому интерфейсу привязывается несколько VRRP групп, каждая из них должна иметь уникальное значение VRID, однако есть возможность указывать дублирующиеся между разными сетевыми интерфейсами значения VRID, хотя рекомендуется так не делать во избежание проблем с распознаванием групп.

Протокол VRRP выполняет свои функции в одном из двух режимов:

- С наследование роли Master - если маршрутизатор с наивысшим приоритетом выходит из строя, а затем восстанавливает свою работу, маршрутизатор с приоритетом ниже возвращает ему роль активного.
- Без наследования роли Master - маршрутизатор, которому была назначена роль активного, сохраняет ее и назначенный виртуальный IP адрес вне зависимости от поведения маршрутизатора с приоритетом выше.

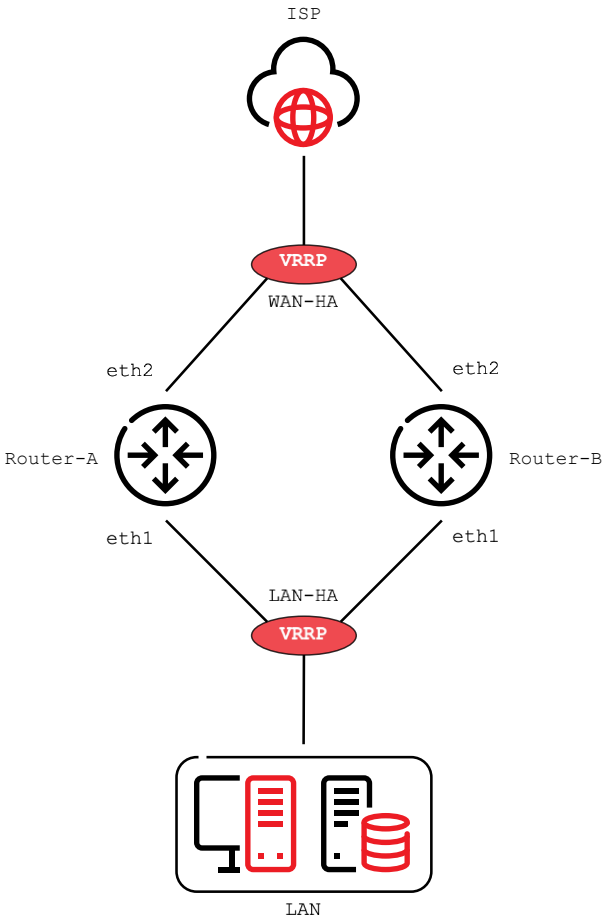
Алгоритм настройки VRRP группы

1) Указать IP адреса и задать описание интерфейсов физических интерфейсов маршрутизатора;
2) Задать имя и описание для VRRP группы;
3) Указать физический интерфейс маршрутизатора, который будет являться участником VRRP группы;
4) Определить приоритет маршрутизатора для VRRP группы;
5) Задать виртуальный адрес VIP для VRRP группы;
6) Указать общий номер VRRP группы VRID;
7) Выполнить аналогичные настройки на втором маршрутизаторе.

Пример настройки VRRP

В качестве примера рассмотрим ситуацию, когда есть два маршрутизатора (*Router-A* и *Router-B*) у которых внешние WAN интерфейсы объединены в одну VRRP группу, а внутренние LAN интерфейсы в другую VRRP группу. Маршрутизатор *Router-A* будет иметь наивысший приоритет и являть Master маршрутизатором. В случае выхода из строя маршрутизатора *Router-A* роль Master перейдет к маршрутизатору *Router-B*. Если маршрутизатор *Router-A* восстанавливает работоспособное состояние, роль Master снова переходит к нему, а маршрутизатору *Router-B* возвращается роль Backup.

Настройка маршрутизаторов будет выполнена в соответствии со схемой, представленной ниже:



Исходные данные для настройки VRRP группы

- *192.168.100.1/29* - адрес внутреннего LAN интерфейса *eth0* на первом маршрутизатора *Router-A*
- *10.250.44.1/29* - адрес внешнего WAN интерфейса *eth1* на первом маршрутизатора *Router-A*
- *192.168.100.2/29* - адрес внутреннего LAN интерфейса *eth0* на первом маршрутизатора *Router-B*
- *10.250.44.2/29* - адрес внешнего WAN интерфейса *eth1* на первом маршрутизатора *Router-B*
- *LAN-HA* - имя VRRP группы, которая объединяет внутренние интерфейсы маршрутизаторов *Router-A* и *Router-B*
- *WAN-HA* - имя VRRP группы, которая объединяет внешние интерфейсы маршрутизаторов *Router-A* и *Router-B*
- *192.168.100.6/29* - виртуальный адрес (VIP) для внутреннего LAN интерфейса VRRP группы *LAN-HA*
- *10.250.44.6/29* - виртуальный адрес (VIP) для внешнего WAN интерфейса VRRP группы *WAN-HA*

- *210* - номер VRID для VRRP группы *LAN-HA*
- *220* - номер VRID для VRRP группы *WAN-HA*
- *10* - приоритет для VRRP групп на маршрутизаторе *Router-A*
- *5* - приоритет для VRRP групп на маршрутизаторе *Router-B*

Команды для настройки VRRP группы

Список команд для настройки VRRP групп на маршрутизаторе Router-A:

```
set interfaces ethernet eth1 address '192.168.100.1/29'
set interfaces ethernet eth1 description 'LAN'
set interfaces ethernet eth2 address '10.250.44.1/29'
set interfaces ethernet eth2 description 'WAN'
set high-availability vrrp group LAN-HA description 'LAN Interface Hight Availability'
set high-availability vrrp group LAN-HA interface 'eth1'
set high-availability vrrp group LAN-HA priority '10'
set high-availability vrrp group LAN-HA virtual-address 192.168.100.6/29
set high-availability vrrp group LAN-HA vrid '210'
set high-availability vrrp group WAN-HA description 'LAN Interface Hight Availability'
set high-availability vrrp group WAN-HA interface 'eth2'
set high-availability vrrp group WAN-HA priority '10'
set high-availability vrrp group WAN-HA virtual-address 10.250.44.6/29
set high-availability vrrp group WAN-HA vrid '220'
```

Список команд для настройки VRRP групп на маршрутизаторе Router-B:

```
set interfaces ethernet eth1 address '192.168.100.2/29'
set interfaces ethernet eth1 description 'LAN'
set interfaces ethernet eth2 address '10.250.44.2/29'
set interfaces ethernet eth2 description 'WAN'
set high-availability vrrp group LAN-HA description 'LAN Interface Hight Availability'
```

```
set high-availability vrrp group LAN-HA interface 'eth1'
set high-availability vrrp group LAN-HA priority '5'
set high-availability vrrp group LAN-HA virtual-address
192.168.100.6/29
set high-availability vrrp group LAN-HA vrid '210'
set high-availability vrrp group WAN-HA description 'LAN
Interface Hight Availability'
set high-availability vrrp group WAN-HA interface 'eth2'
set high-availability vrrp group WAN-HA priority '5'
set high-availability vrrp group WAN-HA virtual-address
10.250.44.6/29
set high-availability vrrp group WAN-HA vrid '220'
```

Основные настройки VRRP группы

```
set high-availability vrrp group <name> vrid <id>
```

Задает имя <*name*> и идентификатор <*id*> для VRRP группы.

```
set high-availability vrrp group <name> interface <interface>
```

Добавляет сетевой интерфейс <*interface*> для VRRP группы <*name*>.

```
set high-availability vrrp group <>name> virtual-address
<address>
```

Добавляет виртуальный адрес <*address*> для VRRP группы <*name*>.

> **Примечание**
>
> *Значение параметра virtual-address может быть представлено как IPv4, так и IPv6 адресом, однако вы не можете использовать их комбинацию в рамках отдельной группы VRRP. Для этого необходимо создать разные группы VRRP с разными VRID, одна из которых будет использовать IPv4 адрес в качестве виртуального, а другая - IPv6 адрес.*

```
set high-availability vrrp group <name> disable
```

Отключает определенную группу VRRP <*name*>.

> **Примечание**
>
> *Дезактивированная группа VRRP будет полностью исключена из процесса VRRP, равно как и маршрутизатор, входящий в группу с данным VRID. Информация о*

дезактивированной группе VRRP также не будет присутствовать в выводе на экран результатов выполнения соответствующих команд в эксплуатационном режиме, а сама группа больше не будет выполнять роль резервной.

Дополнительные настройки VRRP группы

- `set high-availability vrrp group <name> priority <number>`

Задает приоритет <*number*> для VRRP группы <*name*>.

Примечание

Значение приоритета должно быть представлено целым числом в диапазон от 1 до 255. Наивысшее значение приоритета определяет, какой из маршрутизаторов будет определен как активный при очередной процедуре выбора.

- `set high-availability vrrp group <name> no-preempt`

Переводит протокол VRRP в режим без наследования роли. По умолчанию протокол VRRP выполняет свои функции в режиме с наследованием роли.

- `set high-availability vrrp group <name> preempt-delay <number>`

Задает временной интервал для задержки перед изменением роли наследования маршрутизатора с наивысшим приоритетом.

- `set high-availability vrrp group <name> peer-address <address>`

Задает адрес <*address*> для одноадресной (unicast) рассылки в VRRP группе <*name*>

- `set high-availability vrrp group <name> hello-source-address <address>`

Задает адрес источника <*address*> для рассылки hello пакетов в VRRP группе <*name*>

- `set high-availability vrrp group <name> rfc3768-compatibility`

Создает новый VRRP интерфейс, которому автоматически назначаются виртуальный IP адрес и виртуальный MAC адрес.

Примечание

RFC 3768 определяет порядок задания и использования виртуальных MAC адресов для каждого маршрутизатора в процессе VRRP. Виртуальный MAC адрес используется в

> *качестве адреса отправителя для всех периодических сообщений, источником которых является активный маршрутизатор.*

```
show vrrp
```

Выводит на экран информацию о созданной группе VRRP.

Использование скриптов

Функциональность протокола VRRP может быть расширена посредством пользовательских bash скриптов. ПО **Факел** поддерживает использования двух типов скриптов: скриптов контроля (health check) и скриптов состояния. Скрипты контроля (health check) подразумевают выполнение пользовательских проверок в дополнение к стандартной проверке доступности активного маршрутизатора. Скрипты состояния подразумевают изменение состояния, при котором активный маршрутизатор становится резервным или выходит из строя, и наоборот. Скрипты состояния могут быть использованы, например, для активации или дезактивации определенных сервисов.

Скрипты контроля

Пример, представленный ниже, показывает настройку процесса VRRP на использование скрипта контроля, размещенного по следующему пути */config/scripts/vrrp-check.sh script* каждый 60 секунд с последующим переводом группы VRRP в состояние неисправной, если результат проверки будет неудачный (то есть скрипт завершается с ненулевым состоянием) три раза подряд.

```
set high-availability vrrp group Foo health-check script
/config/scripts/vrrp-check.sh
set high-availability vrrp group Foo health-check interval 60
set high-availability vrrp group Foo health-check failure-count
3
```

Скрипты состояния

Скрипты состояния позволяют реализовать такие действия, как, например, запуск или остановку сервисов или даже изменение конфигурации операционной системы Факел в случае, если процесс VRRP фиксирует изменение состояния. Пример, представленный ниже, показывает настройку процесса VRRP на использование скрипта состояния, размещенного по следующему пути */config/scripts/vrrp-fail.sh* с аргументом *Foo* при переводе группы VRRP в состояние неисправной, а также скрипта, размещенного по следующему пути */config/scripts/vrrp-master.sh*, когда маршрутизатор становится активным.

```
set high-availability vrrp group Foo transition-script backup "/config/scripts/vrrp-fail.sh Foo"
set high-availability vrrp group Foo transition-script fault "/config/scripts/vrrp-fail.sh Foo"
set high-availability vrrp group Foo transition-script master "/config/scripts/vrrp-master.sh Foo"
```

Синхронизация состояния VRRP групп

VRRP Sync Group — это механизм, используемый для синхронизации состояний нескольких экземпляров VRRP, работающих на одном устройстве или в одной сети. Это позволяет координировать действия между различными группами VRRP, чтобы обеспечить более гладкое и согласованное переключение между основным и резервным маршрутизаторами в случае сбоя.

Ключевые аспекты VRRP Sync Group:

- **Синхронизация состояний**: Sync Group обеспечивает, чтобы все VRRP экземпляры в группе переключались между состояниями Master и Backup одновременно. Это предотвращает ситуации, когда часть трафика направляется через один маршрутизатор, а часть — через другой, что может привести к проблемам с маршрутизацией и производительностью.
- **Упрощение конфигурации**: Настройка нескольких VRRP экземпляров без синхронизации требует тщательного планирования и управления приоритетами для каждого экземпляра отдельно. Sync Group упрощает этот процесс, позволяя управлять всеми экземплярами как единым целым.
- **Повышение отказоустойчивости**: Использование Sync Group повышает общую надежность сети, так как в случае отказа одного маршрутизатора все релевантные VRRP экземпляры переключаются одновременно, минимизируя риск потери данных или прерывания связи.
- **Гибкость и масштабируемость**: Sync Group позволяет масштабировать сетевую инфраструктуру, добавляя новые экземпляры VRRP без необходимости полного перепланирования существующей конфигурации отказоустойчивости.

Настройка VRRP Sync Group требует тщательного понимания сетевой архитектуры и потребностей организации в отказоустойчивости. Правильно сконфигурированные Sync Groups могут значительно повышать устойчивость сетевых услуг к сбоям и обеспечивать непрерывность бизнес-процессов.

Пример:

```
edit high-availability vrrp
```

- `set sync-group MAIN member VLAN9`
- `set sync-group MAIN member VLAN20`

В примере далее представлена конфигурация, при которой в случае изменения состояния для группы *VLAN9* также изменяется состояние группы *VLAN20*.

```
vrrp {
    group VLAN9 {
        interface eth0.9
        virtual-address 10.9.1.1/24
        priority 200
        vrid 9
    }
    group VLAN20 {
        interface eth0.20
        priority 200
        virtual-address 10.20.20.1/24
        vrid 20
    }
    sync-group MAIN {
        member VLAN20
        member VLAN9
    }
}
```

! Предупреждение

Все элементы синхронной группы должны быть сконфигурированы одинаковым образом. Если для одной из групп VRRP в составе синхронной группы заданы значения приоритета или задержки перед наследованием роли (активный/резервный), отличные от значений для других групп, это может привести к бесконечному циклу изменения состояний групп.

Балансировка WAN-каналов

Общая информация о балансировке WAN-каналов

Исходящий по отношению к маршрутизатору трафик может быть распределен между двумя или более исходящими сетевыми интерфейсами. Если на одном из путей прохождения трафика происходит сбой, то такой путь исключается из таблицы маршрутизации, а трафик перераспределяется между оставшимися работоспособными путями. При восстановлении пути прохождения трафика после сбоя он снова добавляется в таблицу маршрутизации и используется сервисом балансировки трафика. Сервис балансировки трафика автоматически добавляет маршруты для каждого пути прохождения трафика в таблицу балансировки и распределяет трафик между всеми сконфигурированными сетевыми интерфейсами маршрутизатора, руководствуясь их текущим состоянием и заданным администратором весом.

Минимально необходимая конфигурация для балансировки WAN каналов должна включать:

- определенный сетевой интерфейс маршрутизатора с заданным адресом и адресом следующего узла;
- одно правило балансировки с заданными сетевыми интерфейсами LAN (inbound-interface) и WAN (interface) соответственно.

> **Примечание**
>
> *Балансировку WAN каналов не рекомендуется использовать совместно с протоколами динамической маршрутизации, так как сервис балансировки создает дополнительные таблицы маршрутизации и правила межсетевого экрана, которые могут быть не совместимы с протоколами динамической маршрутизации.*

Правила балансировки

Сетевые интерфейсы, их вес и тип трафика, подлежащего балансировке - эти данные указываются в наборах правил балансировки. Правила балансировки применяются в порядке их следования в списке к исходящим по отношению к маршрутизатору сетевым пакетам. При совпадении данных пакета и классификатора определенного правила балансировки этот пакет перенаправляется далее через сетевой интерфейс, указанный в этом правиле. Если для пакета не обнаружено ни одного совпадения, то он перенаправляется далее согласно маршрутам, указанным в системной таблице маршрутизации. Администратор может изменять номер правила балансировка, перемещая его таким образом в списке.

В следующем примере демонстрируется создание правила балансировки, которому может быть назначен номер в интервале от 1 до 9999.

```
fakel@fakel# set load-balancing wan rule 1

Possible completions:

description             Description for this rule

> destination           Destination

exclude                 Exclude packets matching this rule from wan load balance

failover                Enable failover for packets matching this rule from wan load balance

inbound-interface       Inbound interface name (e.g., "eth0") [REQUIRED]

+> interface            Interface name [REQUIRED]

> limit                 Enable packet limit for this rule

per-packet-balancing    Option to match traffic per-packet instead of the default, per-flow

protocol                Protocol to match

> source                Source information
```

Вес сетевого интерфейса

Пример, представленный выше, может быть расширен путем добавления веса сетевым интерфейсам, заданным в правиле балансировки. Предположим, что пропускная способность интерфейса *eth0* больше пропускной способности интерфейса *eth1*. По умолчанию весь исходящий по отношению к маршрутизатору трафик случайным образом распределяется между всеми доступными исходящими сетевыми интерфейсами. Веса могут быть использованы для того, чтобы сделать балансировку определенной.

```
set load-balancing wan rule 1 interface eth0 weight 2
set load-balancing wan rule 1 interface eth1 weight 1
```

Таким образом 66% исходящего трафика будет перенаправляться через сетевой интерфейс eth0, а 33% - через сетевой интерфейс eth1.

Ограничение пропускной способности

Пропускная способность для исходящего трафика может быть искусственно ограничена посредством использования параметра *limit* для отдельного правила балансировки.

```
set load-balancing wan rule <rule> limit <parameter>
```

Дополнительно с параметром *limit* могут быть указаны следующие параметры:

- **threshold** - пороговое значение пропускной способности. Указывается в формате ДО (below) и ОТ (above) определенного значения;

- **period** - период времени, за который выполняется расчет. Возможные значения - second (по умолчанию), minute, hour;
- **burst** - количество сетевых пакетов, допустимое для превышения порогового значения пропускной способности в рамках заданного периода времени (period);
- **rate** - количество сетевых пакетов. Значение по умолчанию - 5.

Балансировка пакетов или потоков

Для исходящего трафика по умолчанию выполняется балансировка потоков целиком. Для работы в таком режиме используется механизм отслеживания соединений по IP адресам отправителя и получателя, а также по номерам портов отправителя и получателя. Каждый поток ассоциируется с определенным сетевым интерфейсом в соответствии с заданными правилами балансировки, в результате чего все сетевые пакеты, соответствующие данному потоку, перенаправляются через этот интерфейс. Преимуществом данного режима работы является сохранение порядка пакетов при использовании разных скоростей передачи данных для разных подключений.

Балансировка пакетов для исходящего трафика позволяет достичь лучших показателей, но только в случае, если нарушение порядка пакетов не влияет на работоспособность сетевой инфраструктуры. Данный режим может быть активирован для отдельного правила балансировки с помощью параметра **per-packet-balancing**.

```
set load-balancing wan rule <rule> per-packet-balancing
```

Исключение трафика из балансировки

В некоторых случаях может возникнуть необходимость исключить трафик из балансировки. Это может быть сделано посредством использования параметра exclude для отдельного правила балансировки.

```
set load-balancing wan rule <rule> exclude
```

В результате при совпадении данных трафика и классификатора такого правила к этому трафику не применяется балансировки, а сам он маршрутизируется согласно записям в системной таблице маршрутизации.

Контроль работоспособности WAN каналов

Работоспособность WAN каналов (сетевых интерфейсов и путей прохождения трафика), используемых сервисом балансировки, периодически проверяется посредством:

- Отправки сообщения ICMP Echo Request удаленным хостам;
- Контроля времени жизни сетевого пакета (TTL);
- Выполнения специального пользовательского скрипта.

Если WAN канал не проходит проверку, он исключается из списка используемых сервисом балансировки. Для включения проверки WAN канала необходимо использовать команду ***set load-balancing wan interface-health <interface>***.

```
fakel@fakel# set load-balancing wan interface-health <interface>

Possible completions:

failure-count    Failure count

nexthop          Outbound interface nexthop address. Can be 'dhcp or ip address' [REQUIRED]

success-count    Success count

+> test          Rule number
```

Для обеспечения выполнения проверки необходимо указать адрес шлюза, через который будет доступен хост-получатель. Параметр ipv4-address может принимать значение dhcp.

- `set load-balancing wan interface-health <interface> nexthop <ipv4-address>`

Необходимо также задать количество неудачных результатов проверки, по достижении которого WAN канал будет помечен, как недоступный, и количество успешных проверок, по достижению которого WAN канал будет снова отмечен, как доступный. Количество данных проверок задается с помощью параметров *failure-count* и *success-count* соответственно. Допустимые значения данных параметров находятся в диапазоне от 1 до 10. Значение по умолчанию - 1.

- `set load-balancing wan interface-health <interface> failure-count <number>`
- `set load-balancing wan interface-health <interface> success-count <number>`

Каждая проверка настраивается для отдельного хоста-получателя и подразумевает выполнение соответствующего конфигурации теста. В случае, если необходимо выполнять проверку с использованием нескольких хостов-получателей, необходимо сконфигурировать несколько тестов, каждый из которых будет выполняться в определенном порядке.

```
fakel@fakel# set load-balancing wan interface-health eth1 test 0

Possible completions:

resp-time     Ping response time (seconds)

target        Health target address
```

```
test-script  Path to user defined script

ttl-limit    Ttl limit (hop count)

type         WLB test type
```

При настройке тестов необходимо задать следующие параметры:

- **resp-time** – максимальное время ожидания ответа на сообщение ICMP Echo Request в секундах. Допустимые значения находятся в диапазоне от 1 до 30. Значение по умолчанию - 5;
- **target** – идентификатор хоста-получателя, в адрес которого будут отправляться сообщения ICMP Echo Request. Допустимые значения - адрес IPv4 или символьное имя хоста;
- **test-script** – пользовательский скрипт, описывающий характер проверки. Скрипт должен возвращать 0 для обозначения успешного результата и отличное от 0 значение – для неудачного результата. Стандартное расположение скриптов - /config/scripts. При использования нестандартного расположения необходимо указывать полный путь до файла скрипта;
- **ttl-limit** – количество переходов между каждой парой шлюзов на пути к хосту-получателю, которое используется для тестов с использованием протокола UDP и атрибута TTL. Критерием успешности является получение в ответ сообщения ICMP Time Expired. Значение по умолчанию - 1;
- **type** – тип теста. Допустимое значение - одно из списка (ping, ttl, script).

Трансляция адреса отправителя

По умолчанию при балансировке WAN каналов IP адрес отправителя каждого исходящего сетевого пакета с запросом заменяется IP адресом соответствующего сетевого интерфейса для того, чтобы обеспечить получение ответа на тот же сетевой интерфейс. Это достигается за счет автоматической генерации правил трансляции адреса отправителя (SNAT), которые применяются только к трафику, к которому был применен механизм балансировки. В случаях, когда такое поведение является нежелательным, механизм автоматической генерации правил SNAT может быть отключен.

```
set load-balancing wan disable-source-nat
```

Привязанные соединения

Входящие соединения могут некорректно обрабатываться WAN каналами при отправке ответа на запрос клиента.

При получении на определенном сетевом интерфейсе входящего сетевого пакета с запросом от клиента может потребоваться, чтобы пакет с ответом отправлялся с того же сетевого интерфейса. Это достигается за счет использования параметра *sticky-connections* при настройке балансировке WAN каналов.

```
set load-balancing wan sticky-connections inbound
```

Отказоустойчивость

В режиме отказоустойчивости один из WAN каналов и соответствующий ему сетевой интерфейс объявляется активным, а остальные - резервными. В этом случае вместо балансировки трафика между всеми работоспособными WAN каналами трафик передается только по активному. В случае отказа активного WAN канала на эту роль выбирается другой из списка резервных на основе состояния и веса интерфейса. Оставшиеся WAN каналы становятся резервными по отношению к новому активному. Роли WAN каналов и соответствующих им интерфейсов могут быть также выбраны на основе порядка правил балансировки, для которых необходимо указывать параметр *failover*.

```
set load-balancing wan rule <number> failover
```

Так как перераспределение установленных ранее сессий между активным и резервными WAN каналами автоматически не выполняется, необходимо производить очистку таблицы соединений с помощью команды *flush-connections* при каждом изменении состояния соединения.

```
set load-balancing wan flush-connections
```

> **! Предупреждение**
>
> *Поскольку это симметричный ключ, его содержимое должно быть известно только вам и вашему удаленному устройству. Обеспечьте безопасное распространение ключа.*

Очистка таблица соединений приведет к тому, что будет выполняться балансировка пакетов вместо балансировки потоков до момента повторного установления соединений.

Пример настройки балансировки WAN-каналов

В качестве примера рассмотрим ситуацию, когда есть два сетевых интерфейса WAN, сконфигурированных по протоколу DHCP, и один LAN интерфейс (eth2).

```
set load-balancing wan interface-health eth0 nexthop 'dhcp'
set load-balancing wan interface-health eth1 nexthop 'dhcp'
```

```
set load-balancing wan rule 1 inbound-interface 'eth2'
set load-balancing wan rule 1 interface eth0
set load-balancing wan rule 1 interface eth1
```

Основные настройки балансировки WAN-каналов

```
set load-balancing wan interface-health <interface>
```

Активирует механизм проверки WAN канала для внешнего интерфейса маршрутизатора <*interface*>.

```
set load-balancing wan interface-health <interface> nexthop
<address|dhcp>
```

Задает адрес следующего маршрутизатора <*address|dhcp*>, который будет использован для проверки состояния канала.

```
set load-balancing wan rule <rule>
```

Создает правило <*id*> для балансировки WAN интерфейсов.

```
set load-balancing wan rule <rule> inbound-interface
<interface>
```

Определяет внутренний LAN интерфейс <*interface*> для правила балансировки <*rule*>.

```
set load-balancing wan rule <rule> interface <interface>
```

Определяет внешний WAN интерфейс <*interface*> для правила балансировки <*rule*>.

```
set load-balancing wan rule <rule> interface <interface> weight
<number>
```

Задает вес <*number*> внешнего WAN интерфейса <*interface*> для правила балансировки <rule>.

```
set load-balancing wan rule <rule> limit <parameter>
```

Задает значение <*parameter*> ограничения пропускной способности для исходящего трафика в правиле балансировки <*rule*>.

```
set load-balancing wan rule <rule> per-packet-balancing
```

Активирует механизм балансировки пакетов для исходящего трафика в определенном правиле балансировки <*rule*>

- `set load-balancing wan rule <rule> exclude`

Исключает трафик, определенный в правиле *<rule>* из балансировки.

- `set load-balancing wan interface-health <interface> failure-count <number>`

Задает количество неудачных результатов проверки *<number>*, по достижении которого WAN канал будет помечен, как недоступный. Допустимые значения данного параметра находятся в диапазоне от 1 до 10. Значение по умолчанию - 1.

- `set load-balancing wan interface-health <interface> success-count <number>`

Задает количество успешных результатов проверки *<number>*, по достижению которого WAN канал будет снова отмечен, как доступный. Допустимые значения данного параметра находятся в диапазоне от 1 до 10. Значение по умолчанию - 1.

- `set load-balancing wan interface-health <interface> test <number>`

Создает тест *<number>* для проверки работоспособности канала внешнего интерфейса маршрутизатора *<interface>*.

- `set load-balancing wan interface-health <interface> test <id> resp-time <number>`

Задает значение максимального времени ожидания ответа на сообщение ICMP Echo Request в секундах. Допустимые значения находятся в диапазоне от 1 до 30. Значение по умолчанию - 5.

- `set load-balancing wan interface-health <interface> test <id> target <text>`

Задает идентификатор хоста-получателя, в адрес которого будут отправляться сообщения ICMP Echo Request. Допустимые значения - адрес IPv4 или символьное имя хоста.

- `set load-balancing wan interface-health <interface> test <id> test-script <text>`

Добавляет пользовательский скрипт *<text>*, описывающий характер проверки. Скрипт должен возвращать 0 для обозначения успешного результата и значение отличное от 0 - для неудачного результата. Стандартное расположение скриптов - */config/scripts*. При использования нестандартного расположения необходимо указывать полный путь до файла скрипта.

- `set load-balancing wan interface-health <interface> test <id> ttl-limit <number>`

Определяет количество переходов между каждой парой шлюзов на пути к хосту-получателю, которое используется для тестов с использованием протокола UDP и атрибута TTL. Критерием успешности является получение в ответ сообщения ICMP Time Expired. Значение по умолчанию - 1;

- `set load-balancing wan interface-health <interface> test <id> type <ping|ttl|script>`

Определяет тип теста. Допустимое значение - ping, ttl, script.

- `set load-balancing wan disable-source-nat`

Отключает настройки правил SNAT для механизма балансировки WAN каналов.

- `set load-balancing wan sticky-connections inbound`

Активирует механизм привязки соединения к конкретному интерфейсу. При получении на определенном сетевом интерфейсе входящего сетевого пакета с запросом от клиента пакет с ответом отправлялся с того же сетевого интерфейса.

- `set load-balancing wan rule <number> failover`

Активирует режим отказоустойчивости для правила балансировки <*number*>.

- `set load-balancing wan flush-connections`

Выполняет очистку таблицы соединений при изменении состояния активного канала WAN.

- `restart wan-load-balance`

Выполняет перезапуск сервиса балансировки.

Мониторинг работы балансировки WAN-каналов

- `show wan-load-balance`

Выводит информацию о работе сервиса балансировки, включая типы тестов и список хостов-получателей.

При выводе на экран соответствующей информации используются следующие символы для обозначения результатов тестов:

- «+» – тест пройден успешно;

- «-» – тест не пройден из-за ошибки;
- **пустое значение** – ни один из тестов не пройден

```
show wan-load-balance connection
```

Выводит информацию о трафике, к которому был применен механизм балансировки.

Сервисные механизмы

Удаленный доступ

Общая информация об удаленном доступе

Удаленный доступ к ПО **Факел** выполняется по средствам подключения к устройству по протоколу SSH. SSH - сетевой протокол прикладного уровня, который используется для удалённого управление различным оборудованием и операционными системами, а также туннелирование TCP соединений (например, для передачи файлов). При передаче по SSH информация шифруется, что обеспечивает высокий уровень защиты. Протокол SSH работает по модели «клиент-сервер». Он обеспечивает зашифрованный канал между ними и предотвращает несанкционированный доступ к данным. К частыми сценариями его использования можно отнести работу с удаленными машинами и передачу файлов на устройства. По умолчанию для подключения и передаче файлов по SSH используется TCP порт 22. В процессе настройки SSH на Факел администратор может указать предпочтительный для него порт. На данный момент существует две основные версии протокола SSH: SSH-1 и SSH-2.

Работа SSH-протокола базируется на нескольких этапах:

- Открытие транспортного канала аутентификации;
- Аутентификация;
- Подключение.

Алгоритм настройки SSH для удаленного доступа

1) Войти в консоль маршрутизатора. Используйте консольный доступ, чтобы войти в командную строку маршрутизатора Факел.
2) Перейти в режим конфигурации. Введите команду configure, чтобы перейти в режим конфигурации.
3) Настроить параметры SSH. Введите следующие команды для настройки параметров SSH:
    - Порт, на котором будет работать SSH (по умолчанию 22)
        - `set service ssh port <number>`
    - IP-адрес, на котором будет прослушиваться SSH (по умолчанию все интерфейсы)
        - `set service ssh listen-address <address>`
    - Пользователь, которому разрешен удаленный доступ по SSH

```
set service ssh access-control allow user <name>
```

4) Настроить метод аутентификации для подключения по SSH. Введите следующую команду для настройки метода аутентификации:
    - Метод аутентификацию для пользователя по паролю

```
set system login user <name> authentication
plaintext-password <text>
```

5) Сохранить изменения. Выйдите из режима конфигурации, используя команду ***commit*** и ***save***.

6) Проверить подключение. Попробуйте подключиться к маршрутизатору по SSH, используя указанный порт и адрес. Убедитесь, что подключение работает корректно.

> **Примечание**
>
> *Это основные шаги для настройки SSH на маршрутизаторе Факел. Обратите внимание, что некоторые параметры могут отличаться в зависимости от конфигурации вашего маршрутизатора.*

Пример настройки удаленного доступа

В качестве примера рассмотрим настройку удаленного доступа до маршрутизатора на базе ПО **Факел** через SSH по TCP порту 2022. В качестве метода аутентификации будет настроен метод аутентификации по ключу.

Исходные данные для настройки VRRP группы:

- *fakel* – учетная запись пользователя, для которого будет настроена аутентификация по паролю
- *2202* – порт, на котором будет работать SSH
- *192.168.100.2* – IP-адрес, на котором будет прослушиваться SSH
- *<public-key-value>* – открытый ключ учетной записи пользователя *fakel*
- *ssh-rsa* – тип открытого ключа для учетной записи пользователя *fakel*
- *admin-arm* – идентификатор открытого ключа для учетной записи пользователя *fakel*

Список команд для настройки удаленного доступа:

```
set service ssh port 2022
```

```
set service ssh listen-address 192.168.100.2
set service ssh access-control allow user fakel
set system login user fakel authentication public-keys admin-arm type ssh-rsa
set system login user itr authentication public-keys admin-arm key <public-key-value>
```

Основные настройки удаленного доступа

```
set service ssh port <port>
```

Задает порт *<port>*, на котором будет работать SSH. По умолчанию SSH работает на порту 22.

```
set service ssh listen-address <address>
```

Устанавливает адрес *<address>* на котором будет прослушиваться SSH. Можно задать несколько адресов.

```
set service ssh ciphers <cipher>
```

Определяет какие алгоритмы шифрования *<cipher>*, будут использоваться для SSH-соединения. Можно указать несколько алгоритмов шифрования.

Поддерживаемые алгоритмы шифрования: 3des-cbc, aes128-cbc, aes192-cbc, aes256-cbc, aes128-ctr, aes192-ctr, aes256-ctr, arcfour128, arcfour256, arcfour, blowfish-cbc, cast128-cbc.

```
set service ssh disable-password-authentication
```

Отключает возможность использовать метод аутентификации на основе пароля. Пользователи могут пройти аутентификацию только по ключу.

```
set service ssh disable-host-validation
```

Отключает проверку хоста через обратный DNS запрос. Отключение проверки хоста ускоряет время входа в систему, когда проверка через обратный запрос недоступна.

```
set service ssh macs <mac>
```

Определяет доступные MAC алгоритмы *<mac>*. Алгоритм MAC используется в протоколе SSH версии 2 для защиты целостности данных. Может быть определено несколько алгоритмов.

Поддерживаемые MAC алгоритмы: hmac-md5, hmac-md5-96, hmac-ripemd160, hmac-sha1, hmac-sha1-96, hmac-sha2-256, hmac-sha2-512, umac-64@openssh.com, umac-128@openssh.com, hmac-md5-etm@openssh.com, hmac-md5-96-etm@openssh.com, hmac-ripemd160-etm@openssh.com, hmac-sha1-etm@openssh.com, hmac-sha1-96-etm@openssh.com, hmac-sha2-256-etm@openssh.com, hmac-sha2-512-etm@openssh.com, umac-64-etm@openssh.com, umac-128-etm@openssh.com

- `set service ssh access-control allow user <name>`

Задает пользователя <*name*>, которому разрешен удаленный доступ по SSH.

- `set service ssh access-control allow group <name>`

Задает группу пользователей <*name*>, которым разрешен удаленный доступ по SSH.

- `set service ssh access-control deny user <name>`

Задает пользователя <*name*>, которому запрещен удаленный доступ по SSH.

- `set service ssh access-control deny group <name>`

Задает группу пользователей <*name*>, которым запрещен удаленный доступ по SSH.

- `set service ssh client-keepalive-interval <interval>`

Устанавливает интервал передачи keepalive <*interval*> пакетов от сервера к клиенту.

- `set service ssh key-exchange <kex>`

Определяет какие алгоритмы обмена ключами KEX <*kex*> можно использовать для аутентификации через SSH.

Поддерживаемые KEX алгоритмы: diffie-hellman-group1-sha1, diffie-hellman-group14-sha1, diffie-hellman-group14-sha256, diffie-hellman-group16-sha512, diffie-hellman-group18-sha512, diffie-hellman-group-exchange-sha1, diffie-hellman-group-exchange-sha256, ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521, curve25519-sha256, curve25519-sha256@libssh.org.

- `set service ssh loglevel <quiet | fatal | error | info | verbose>`

Устанавливает уровень журналирования для sshd. По умолчанию установлен уровень info.

- `set service ssh vrf <name>`

Определяет VRF экземпляр <*name*>, в котором будет работать SSH.

- `restart ssh`

Перезапускает процесс демона SSH, текущая сессия не затрагивается, перезапускается только фоновый демон.

- `generate ssh server-key`

Генерирует новые ключи SSH для маршрутизатора и перезапустите сервер SSH.

Протокол ARP

Общая информация о протоколе ARP

Протокол ARP является коммуникационным протоколом, который используется для обнаружения адресов канального уровня, таких как MAC адрес, ассоциированных с адресами сетевого уровня, такими как IP адрес. Такая ассоциация является критически важной для обеспечения функционирования стека протоколов IP. Протокол ARP был впервые определен в спецификации RFC 826, получившей в итоге статус стандарта STD 37.

В сетях, построенных на основе протокола IPv6, функциональность протокола ARP обеспечивается другим протоколом - NDP.

Для работы с записями ARP таблицы необходимо использовать команды, описание которых представлено в разделах ниже.

Список команд

- `set protocols static arp <address> hwaddr <mac>`

Создает статическую запись в ARP таблице, определяющую постоянную ассоциацию между IP адресом <*address*> и MAC адресом <*mac*>.

- `show protocols static arp`

Выводит на экран все записи ARP таблицы по всем интерфейсам.

- `show protocols static arp interface <iface_name>`

Выводит на экран записи ARP таблицы для конкретного интерфейса <*iface_name*>.

Широковещательная ретрансляция UDP

Общая информация о широковещательной ретрансляции UDP

Некоторые производители используют широковещательные рассылки для идентификации своего оборудования в пределах одного сегмента локальной сети. При разделении сети на несколько сегментов (VLAN) теряется возможность идентификации оборудования.

Для того, чтобы решить проблему обмена широковещательными UDP сообщения между различными сегментами локальной сети используется механизм широковещательной ретрансляция UDP (UDP broadcast relay).

Каждый UDP-порт, который будет заниматься обменом широковещательными UDP сообщениями, требует одного уникального идентификатора. В настоящее время операционная система Факел поддерживаем 99 идентификаторов.

Пример настройки широковещательной ретрансляции UDP

Пример настройки пересылки широковещательных пакетов, полученных на UDP-порт 1900 сетевых интерфейсов eth3, eth4 или eth5, на все остальные сетевые интерфейсы.

Список команд для настройки широковещательной ретрансляции UDP:

```
set service broadcast-relay id 1 description 'SONOS'
set service broadcast-relay id 1 interface 'eth3'
set service broadcast-relay id 1 interface 'eth4'
set service broadcast-relay id 1 interface 'eth5'
set service broadcast-relay id 1 port '1900'
```

Основные настройки широковещательной ретрансляции UDP

```
set service broadcast-relay id <n> description <description>
```

Добавляет описание *<description>* для каждого уникального идентификатора *<n>* сервиса широковещательной ретрансляции. Описание удобно использовать, когда в системе настроено нескольких различных портов или приложений.

```
set service broadcast-relay id <n> interface <interface>
```

Определяет интерфейс, который будет использоваться для приема и ретрансляции отдельных широковещательных пакетов.

- `set service broadcast-relay id <n> address <ipv4-address>`

Устанавливает IP-адрес источника *<ipv4-address>* пересылаемых пакетов, в противном случае используется исходный адрес отправителя.

- `set service broadcast-relay id <n> port <port>`

Задает номер UDP-порта, который будет использоваться вашим приложением.

- `set service broadcast-relay id <n> disable`

Отключает определенный экземпляр *<n>* широковещательной ретрансляции без удаления из конфигурации устройства.

- `set service broadcast-relay disable`

Отключает сервис широковещательной ретрансляции, не удаляя его из текущей конфигурации.

> **Примечание**
>
> *Вы можете запустить службу ретрансляции широковещательных пакетов UDP на нескольких маршрутизаторах, подключенных к одной подсети. Широковещательного шторма пакетов при ретрансляции UDP не происходит.*

Отслеживание соединений – Conntrack Sync

Общая информация об отслеживании соединений

Одной из важных функций, построенных на основе фреймворка Netfilter, является отслеживание соединений (conntrack-sync). Отслеживание соединений позволяет ядру отслеживать все логические сетевые соединения или сеансы и, таким образом, соотносить все пакеты, которые могут составлять это соединение. NAT опирается на эту информацию, чтобы переводить все связанные пакеты одинаковым образом, а iptables может использовать эту информацию для работы в качестве межсетевого экрана с контролем состояния.

Однако состояние соединения совершенно не зависит от состояния верхнего уровня, например, от состояния TCP или SCTP. Отчасти это объясняется тем, что при простой пересылке пакетов, т. е. без локальной доставки, механизм TCP может быть вообще не задействован. Эвристика для таких протоколов часто основана на заданном значении тайм-аута бездействия, по истечении которого соединение Netfilter разрывается.

Каждое соединение Netfilter однозначно идентифицируется кортежем (протокол третьего уровня, адрес источника, адрес назначения, протокол четвертого уровня, ключ четвертого уровня). Ключ уровня-4 зависит от транспортного протокола; для TCP/UDP это номера портов, для туннелей это может быть их туннельный идентификатор, но в

остальных случаях он равен нулю, как если бы он не был частью кортежа. Для того чтобы во всех случаях можно было проверить TCP-порт, пакеты будут в обязательном порядке подвергаться дефрагментации.

Для синхронизации трафика можно использовать как Многоадресную (Multicast), так и Одноадресную (Unicast) рассылки. В большинстве приведенных ниже примеров используется Multicast, но можно указать и unicast, используя ключ рее после конкретного интерфейса, как в следующем примере:

```
set service conntrack-sync interface eth0 peer 192.168.0.250
```

Пример настройки Conntrack Sync

Следующий пример представляет собой простую настройку службы отслеживания состояния между двумя маршрутизаторами.

Список команд для настройки службы отслеживания состояния на 1-ом и 2-ом маршрутизаторах:

```
set high-availablilty vrrp group internal virtual-address
set high-availability vrrp sync-group syncgrp member 'internal'
set service conntrack-sync accept-protocol 'tcp'
set service conntrack-sync accept-protocol 'udp'
set service conntrack-sync accept-protocol 'icmp'
set service conntrack-sync failover-mechanism vrrp sync-group
'syncgrp'
set service conntrack-sync interface 'eth0'
set service conntrack-sync mcast-group '225.0.0.50'
```

На основном маршрутизаторе информация о состоянии активных соединений представлена во внутреннем кэше таблицы conntrack-sync. Такое же количество активных соединений должно присутствовать во внешнем кэше таблицы conntrack-sync резервного маршрутизатора. Для получения информации о количестве активных соединений используется команда show conntrack-sync statistics.

Проверка состояния активных соединений на основном маршрутизаторе:

```
fakel@fakel:~$ show conntrack-sync statistics

Main Table Statistics:

cache internal:

current active connections:              10
```

```
connections created:                    8517    failed:            0
connections updated:                     127    failed:            0
connections destroyed:                  8507    failed:            0
cache external:
current active connections:                0
connections created:                       0    failed:            0
connections updated:                       0    failed:            0
connections destroyed:                     0    failed:            0
traffic processed:
                  0 Bytes                          0 Pckts
multicast traffic (active device=eth0):
              868780 Bytes sent               224136 Bytes recv
               20595 Pckts sent                14034 Pckts recv
                   0 Error send                    0 Error recv
message tracking:
                   0 Malformed msgs                    0 Lost msgs
```

Проверка состояния активных соединений на резервном маршрутизаторе:

```
fakel@fakel:~$ show conntrack-sync statistics
Main Table Statistics:
cache internal:
current active connections:                0
connections created:                       0    failed:            0
connections updated:                       0    failed:            0
connections destroyed:                     0    failed:            0
cache external:
current active connections:               10
connections created:                     888    failed:            0
connections updated:                     134    failed:            0
connections destroyed:                   878    failed:            0
```

```
traffic processed:
                    0 Bytes                            0 Pckts
multicast traffic (active device=eth0):
               234184 Bytes sent                907504 Bytes recv
                14663 Pckts sent                 21495 Pckts recv
                    0 Error send                     0 Error recv
message tracking:
                    0 Malformed msgs                     0 Lost msgs
```

Основные настройки Conntrack Sync

- `set service conntrack-sync accept-protocol <proto>`

Определяет протоколы ***<proto>***, для которых будут синхронизироваться локальные записи службы отслеживания состояния.

Поддерживаемые протоколы: tcp, sctp, dccp, udp, icmp and ipv6-icmp.

- `set service conntrack-sync event-listen-queue-size <size>`

Задает максимальный размера очереди для локальных событий службы отслеживания состояния, который может быть достигнут. Значение размера очереди для прослушивания локальных событий службы отслеживания состояния задается в Мегабайтах.

- `set service conntrack-sync expect-sync <all|ftp|h323|nfs|sip|sqlnet>`

Определяет протокол, для которого необходимо синхронизировать записи ожиданий.

- `set service conntrack-sync failover-mechanism vrrp sync-group <group>`

Активирует механизм отказоустойчивости, используемый для службы отслеживания состояния. Поддерживается только протокол VRRP.

- `set service conntrack-sync ignore-address <x.x.x.x>`

Задает IP-адрес или подсеть *<x.x.x.x>*, для которых локальные записи службы отслеживания состояния не будут синхронизироваться.

- `set service conntrack-sync interface <name>`

Определяет интерфейс <*name*>, используемый для синхронизации записей службы отслеживания состояния.

- `set service conntrack-sync interface <name> port <port>`

Задает номер порта <*port*> для интерфейса <*name*>, используемого для синхронизации записей службы отслеживания состояния.

- `set service conntrack-sync listen-address <ipv4address>`

Задает локальный IPv4-адрес <*ipv4address*> прослушивания входящих соединений для службы отслеживания состояния.

- `set service conntrack-sync mcast-group <x.x.x.x>`

Задает адрес для группы Многоадресной трансляции <*x.x.x.x*>, используемой для синхронизации записей службы отслеживания состояния. По умолчанию используется адрес *225.0.0.50*.

- `set service conntrack-sync interface <name> peer <address>`

Задает адрес узла <*address*> отправки Одноадресных UDP сообщений для службы отслеживания состояния, если не используется конфигурация Многоадресной трансляции, описанная выше.

- `set service conntrack-sync sync-queue-size <size>`

Задает размер очереди <*size*> для синхронизации записей системы отслеживания состояния. Значение задается в Мегабайтах.

- `set service conntrack-sync disable-external-cache`

Отключает внешний кэш и напрямую передает состояния потока в систему отслеживания соединений в ядро резервного межсетевого экрана.

Мониторинг состояния Conntrack Sync

- `show conntrack table ipv4`

Выводит таблицу системы отслеживания состояния для протокола IPv4-.

Примечание

> *Если в таблице отсутствуют записи об активных соединениях и выдается предупреждение, значит, система отслеживания состояния не включена. Чтобы включить систему отслеживания состояния, достаточно создать правило трансляции NAT или правило межсетевого экрана.*

- `show conntrack-sync cache external`

Выводит записи внешнего кэша синхронизации соединений системы отслеживания состояния

- `show conntrack-sync cache internal`

Выводит записи внутреннего кэша синхронизации соединений системы отслеживания состояния

- `show conntrack-sync statistics`

Выводит актуальную статистику системы отслеживания соединений.

- `show conntrack-sync status`

Выводит информацию об актуальном состоянии системы отслеживания соединений.

Консольный сервер

Общая информация о консольном сервере

Маршрутизатор, под управлением ПО **Факел** может выступать в качестве устройства внеполосного управления, обеспечивающего удаленный доступ по протоколу SSH к непосредственно подключенным последовательным интерфейсам.

В качестве последовательных интерфейсов могут выступать как интерфейсы, напрямую подключенные к процессору или чипсету (в Linux это чаще всего называется ttyS-интерфейсом), так и любые другие преобразователи USB в последовательные интерфейсы (микросхемы Prolific PL2303 или FTDI FT232/FT4232).

Основные настройки консольного сервера

Между компьютерами чаще всего используется конфигурация «8N1»: восемь битов символов, с одним стартовым битом, одним стоповым битом и без бита четности. Таким образом, для передачи одного символа используется 10 бод, и деление скорости передачи сигнала на десять дает общую скорость передачи в символах в секунду. Это также является настройкой по умолчанию, если ни одна из этих опций не задана.

- `connect console <device>`

Локальное подключение к последовательному порту <*device*>.

> **Подсказка**
>
> *К одному и тому же последовательному порту могут подключаться несколько пользователей, но запись в консольный порт разрешена только одному.*

- `set service console-server device <device> data-bits [7 | 8]`

Задает значение битов данных *[7 | 8]* для передачи информации через последовательный порт <*devise*>. Если этот параметр не задан, то по умолчанию будет установлено значение 8.

- `set service console-server device <device> description <string>`

Добавляет описание <*string*> для последовательного порта <*device*>, которое помогает идентифицировать подключенное периферийное устройство.

- `set service console-server device <device> parity [even | odd | none]`

Устанавливает параметр четности *[even | odd | none]* для последовательного порта <*device*>. Если этот параметр не задан, то в качестве значения по умолчанию он будет установлен параметр *none*.

- `set service console-server device <device> stop-bits [1 | 2]`

Задает параметр стопового бита *[1 | 2]* для последовательного порта <*device*>. Если этот параметр не задан, то по умолчанию будет установлено значение *1*.

- `set service console-server device <device> speed [ 300 | 1200 | 2400 | 4800 | 9600 | 19200 | 38400 | 57600 | 115200 ]`

Определяет скорость передачи данных *[300 | 1200 | 2400 | 4800 | 9600 | 19200 | 38400 | 57600 | 115200]* для последовательного порта <*device*>.

> **Примечание**
>
> *Преобразователи USB в последовательный порт выполняют большую часть своей работы в программном обеспечении, поэтому следует внимательно относиться к выбранной скорости передачи данных, так как иногда они не могут справиться с ожидаемой скоростью.*

Настройка удаленного доступа через консольный сервер

Пользователь может напрямую подключиться через SSH к настроенному последовательному порту.

- `set service console-server device <device> ssh port <port>`

Открывает TCP порт *<port>* последовательного порта *<device>* для подключения по SSH. После успешной аутентификации пользователь будет переброшен непосредственно на подключенное через последовательный порт устройство.

> **Подсказка**
>
> *К одному и тому же последовательному устройству могут подключаться несколько пользователей, но запись в консольный порт разрешена только одному.*

Мониторинг состояния консольного сервера

- `show console-server ports`

Выводит информацию о настроенных последовательных портах и конфигурацию соответствующих этим портам интерфейсов.

- `show console-server user`

Выводит информацию о подключенных к последовательным портам пользователях.

- `show log console-server`

Выводит на экран журнал событий консольного сервера.

Ретранслятор DHCP

Общая информация о ретрансляторе DHCP

Устройство на базе операционной системы Факел можно настроить на работу в качестве агента ретрансляции DHCP. После настройки агента ретрансляции DHCP маршрутизатор начинает пересылать DHCP запросы на внешний DHCP сервер. Агент ретрансляции DHCP работает с адресами IPv4 и IPv6.

На всех интерфейсах, используемые для ретрансляции DHCP, должны быть настроены IP адреса.

Пример настройки ретранслятора DHCP для протокола IPv4

Пример настройки агента ретрансляции DHCP для протокола IPv4 со следующими параметрами:

- Прослушивание запросов DHCP настроено на интерфейсе eth1.
- DHCP-сервер расположен по IPv4-адресу 10.0.1.4, который доступен через интерфейс eth2.
- Маршрутизатор получает запросы DHCP клиентов на интерфейс eth1 и передает их на сервер 10.0.1.4 через интерфейс eth2.

Список команд для настройки агента ретранслятора DHCP:

```
set service dhcp-relay interface eth1
set service dhcp-relay interface eth2
set service dhcp-relay server 10.0.1.4
set service dhcp-relay relay-options relay-agents-packets
discard
```

Сформированная конфигурация агента ретранслятора DHCP будет выглядеть следующим образом:

```
fakel@fakel# show service dhcp-relay
    interface eth1
    interface eth2
    server 10.0.1.4
    relay-options {
       relay-agents-packets discard
    }
```

Пример настройки ретранслятора DHCP для протокола IPv6

Пример настройки агента ретрансляции DHCP для протокола IPv6 со следующими параметрами:

- Запросы DHCPv6 принимаются маршрутизатором на прослушивающем интерфейсе eth1.
- Запросы DHCPv6 направляются через интерфейс eth2.

- Внешний DHCPv6-сервер находится по адресу 2001:db8::4

Список команд для настройки агента ретранслятора DHCP:

```
set service dhcpv6-relay listen-interface eth1
set service dhcpv6-relay upstream-interface eth2
set service dhcpv6-relay upstream-interface eth2 address
2001:db8::4
```

Сформированная конфигурация агента ретранслятора DHCP будет выглядеть следующим образом:

```
fakel@fakel# show service dhcpv6-relay
    listen-interface eth1 {
    }
    upstream-interface eth2 {
        address 2001:db8::4
    }
```

Настройки ретранслятора DHCP для протокола IPv4

```
set service dhcp-relay interface <interface>
```

Определяет интерфейсы *<interface>*, участвующие в процессе ретрансляции DHCP, включая интерфейс для связи с DHCP-сервером.

```
set service dhcp-relay server <server>
```

Задает IP адрес DHCP *<server>*, который будет обрабатывать ретранслированные пакеты.

```
restart dhcp relay-agent
```

Перезапускает службу ретрансляции DHCP.

```
set service dhcp-relay relay-options hop-count <count>
```

Задает максимальное количество переходов *<count>*, после которого пакеты будут отброшены. Значение задается в диапазоне *0-255*. Значение по умолчанию *10*.

```
set service dhcp-relay relay-options max-size <size>
```

Задает максимальный размер DHCP-пакетов *<size>*, включающих информацию об агенте ретрансляции. Если размер DHCP-пакета превышает это значение, он будет передан без добавления информации об агенте ретрансляции. Значение задается в диапазоне *64-1400*. Значение по умолчанию *576*.

```
set service dhcp-relay relay-options relay-agents-packets
<append | discard | forward | replace>
```

Существует четыре типа политики перенаправления DHCP-пакетов:

- **append**: Агенту ретрансляции разрешается добавлять свою собственную информацию о ретрансляции в полученный DHCP-пакет, не обращая внимания на уже имеющуюся в пакете информацию о ретрансляции.
- **discard**: Полученные пакеты, уже содержащие информацию о ретрансляции, будут отброшены.
- **forward**: Все пакеты пересылаются, уже имеющаяся ретрансляционная информация игнорируется.
- **replace**: Ретранслируемая информация, уже присутствующая в пакете, удаляется и заменяется собственным набором ретранслируемой информации маршрутизатора.

Настройки ретранслятора DHCP для протокола IPv6

```
set service dhcpv6-relay listen-interface <interface>
```

Устанавливает интерфейс *<interface>* в качестве прослушивающего интерфейса для ретрансляции DHCPv6. В качестве прослушивающего интерфейса можно указать несколько интерфейсов.

```
set service dhcpv6-relay upstream-interface <interface> address
<server>
```

Определяет интерфейс *<interface>*, который будет принимать ответы от DHCP сервера *<server>* и других агентов ретрансляции.

```
restart dhcpv6 relay-agent
```

Перезапускает службу ретрансляции DHCPv6.

```
set service dhcpv6-relay max-hop-count <count>
```

Задает значение максимального количества переходов до отбрасывания пакетов *<count>*. Значение по умолчанию: *10*

```
set service dhcpv6-relay use-interface-id-option
```

Если этот параметр установлен, то агент ретрансляции будет вставлять идентификатор интерфейса. Эта опция устанавливается автоматически, если используется более одного прослушивающего интерфейса.

DHCP сервер

Протокол DHCP позволяет производить автоматическую настройку сети на компьютерах и других устройствах. DHCP это стандартный протокол в сети с архитектурой «клиент-сервер», который динамически назначает IP-адреса и другую необходимую информацию конфигурации для новых устройств в сети. Операционная система Факел может выступать в качестве DHCP-сервера для назначения IPv4 и IPv6 адресов устройствам в локальной сети.

DHCP сервер IPv4

Служба DHCP может обслуживать несколько общих сетей, причем каждая общая сеть имеет одну или несколько подсетей. Каждая подсеть должна быть настроена на интерфейсе устройства, который выступает в качестве DHCP сервера. Для определения пула динамических адресов внутри подсети задается специальный диапазон адресов. Можно определить несколько диапазонов адресов. Статические привязки используются для назначения статических адресов клиентам на основе их MAC-адресов.

Необработанные параметры

Необработанные параметры могут быть переданы в *shared-network-name, subnet u static-mapping:*

```
set service dhcp-server shared-network-name <name> shared-network-parameters
set service dhcp-server shared-network-name <name> subnet <subnet> subnet-parameters
set service dhcp-server shared-network-name <name> subnet <subnet> static-mapping <description> static-mapping-parameters
```

Эти параметры напрямую передаются в файл *dhcpd.conf* под тем узлом конфигурации, в котором они определены. Они не проверяются, поэтому ошибка в исходных параметрах не будет отловлена скриптами операционной системы и приведет к невозможности запуска службы *dhcpd*. Необходимо всегда проверять корректность внесенных параметров перед фиксацией конфигурации устройства.

Пример настройки DHCP сервера IPv4

Пример настройки устройства на базе операционной системы Факел, которое будет выступать в качестве DHCP сервера локальной сети. DHCP сервер будет передавать устройствам внутри локальной сети информацию о назначенном им IP адресе, а также адресах DNS сервера и шлюза по умолчанию.

Для настройки службы DHCP будут использоваться следующие параметры:

- Для шлюза по умолчанию и DNS сервера будет использоваться адрес *192.168.0.1/24*;
- Диапазон адресов *192.168.0.2/24 - 192.168.0.8/24* будет зарезервирован для назначения статических адресов;
- Клиентам DHCP будут назначены IP-адреса в диапазоне *192.168.0.9 - 192.168.0.254* с доменным именем *internal-network*;
- Срок действия аренды DHCP составляет один день (86400 секунд);
- Только узлы из локальной сети *192.168.0.0/24* могут использовать DNS сервер.

Список команд для настройки DHCP сервера:

```
set service dhcp-server shared-network-name LAN subnet 192.168.0.0/24 default-router '192.168.0.1'
set service dhcp-server shared-network-name LAN subnet 192.168.0.0/24 name-server '192.168.0.1'
set service dhcp-server shared-network-name LAN subnet 192.168.0.0/24 domain-name 'fakel.net'
set service dhcp-server shared-network-name LAN subnet 192.168.0.0/24 lease '86400'
set service dhcp-server shared-network-name LAN subnet 192.168.0.0/24 range 0 start 192.168.0.9
set service dhcp-server shared-network-name LAN subnet 192.168.0.0/24 range 0 stop '192.168.0.254'
set service dns forwarding cache-size '0'
set service dns forwarding listen-address '192.168.0.1'
set service dns forwarding allow-from '192.168.0.0/24'
```

Пример настройки отказоустойчивого кластера DHCP

Параметры для настройки отказоустойчивого кластера DHCP:

- Отказоустойчивый сервер DHCP будет настроен для подсети 192.0.2.0/24

- Для шлюза по умолчанию и DNS сервера будет настроен адрес 192.0.2.254
- Для адреса основного участника отказоустойчивого кластера DHCP будет настроен адрес 192.168.189.252
- Для адреса резервного участника отказоустойчивого кластера DHCP будет настроен адрес 192.168.189.253
- DHCP сервер будет выдавать клиентам адреса из диапазона 192.168.189.10 - 192.168.189.250

Список команд для общих настроек отказоустойчивого кластера DHCP:

```
set service dhcp-server shared-network-name NET-VYOS subnet 192.0.2.0/24 default-router '192.0.2.254'
set service dhcp-server shared-network-name NET-VYOS subnet 192.0.2.0/24 name-server '192.0.2.254'
set service dhcp-server shared-network-name NET-VYOS subnet 192.0.2.0/24 domain-name 'vyos.net'
set service dhcp-server shared-network-name NET-VYOS subnet 192.0.2.0/24 range 0 start '192.0.2.10'
set service dhcp-server shared-network-name NET-VYOS subnet 192.0.2.0/24 range 0 stop '192.0.2.250'
set service dhcp-server shared-network-name NET-VYOS subnet 192.0.2.0/24 enable-failover
```

Список команд для настройки основного участника отказоустойчивого кластера DHCP:

```
set service dhcp-server failover source-address '192.168.189.252'
set service dhcp-server failover name 'NET-VYOS'
set service dhcp-server failover remote '192.168.189.253'
set service dhcp-server failover status 'primary'
```

Список команд для настройки резервного участника отказоустойчивого кластера DHCP:

```
set service dhcp-server failover source-address '192.168.189.253'
set service dhcp-server failover name 'NET-VYOS'
```

```
set service dhcp-server failover remote '192.168.189.252'
set service dhcp-server failover status 'secondary'
```

Основные настройки DHCP сервера IPv4

```
set service dhcp-server hostfile-update
```

Обновляет файл */etc/hosts*, добавляя актуальные DNS записи для каждого клиента.

Запись будет иметь формат: <*shared-network-name*>_<*hostname*>.<*domain-name*>

```
set service dhcp-server host-decl-name
```

При записи в файл */etc/hosts* исключает значение <*shared-network-name*>_ из DNS записи клиента, используя только имя хоста и имя домена: <*hostname*>.<*domain-name*>

```
set service dhcp-server shared-network-name <name> domain-name
<domain-name>
```

Устанавливает доменное имя <*domain-name*> для хостов, которые будут получать адреса из подсети <name. Доменное имя будет добавлено к имени хоста клиента для формирования полностью определенного доменного имени (FQDN).

```
set service dhcp-server shared-network-name <name> domain-
search <domain-name>
```

Задает дополнительные доменные имена, которые образуют список доменов, используемых при поиске хостов в локальной сети по доменному имени. Если эта опция не указана, то по умолчанию используется единственный домен, указанный в параметре *domain-name*.

```
set service dhcp-server shared-network-name <name> name-server
<address>
```

Устанавливает адрес DNS сервера, который будут получать адреса из подсети <*name*>. Поддерживается возможность указать несколько адресов для использования различных DNS серверов.

```
set service dhcp-server shared-network-name <name> ping-check
```

Активирует механизм проверки ICMP для подсети <*subnet*> из которой DHCP сервер будет выдавать адреса.

Примечание

> *Если свободных адресов для назначения клиентам нет, но есть IP-адреса, в использовании которых отказано, DHCP-сервер будет пытаться вернуть и использовать такие IP-адрес независимо от значения параметра abandon-lease-time.*

```
set service dhcp-server listen-address <address>
```

Устанавливает адрес <*address*> для прослушивания DHCP сервером входящих запросов от клиентов.

Настройка индивидуальной подсети для DHCP сервера IPv4

```
set service dhcp-server shared-network-name <name>
authoritative
```

При активации данного параметра устройство будет являться единственным DHCP сервером для данной сети <*name*>.

```
set service dhcp-server shared-network-name <name> subnet
<subnet> default-router <address>
```

Устанавливает адрес шлюза по умолчанию <*address*> для хостов, которые будут получать адреса из подсети <*subnet*>.

```
set service dhcp-server shared-network-name <name> subnet
<subnet> name-server <address>
```

Устанавливает адрес DNS сервера <*address*> для хостов, которые будут получать адреса из подсети <*subnet*>. Поддерживается возможность указать несколько адресов для использования различных DNS серверов.

```
set service dhcp-server shared-network-name <name> subnet
<subnet> lease <time>
```

Определяет время аренды IP адреса <*time*>, получаемого по DHCP, для хоста из подсети <*subnet*>. Значение, используемое по умолчанию - *86400* секунд, что соответствует одному дню.

```
set service dhcp-server shared-network-name <name> subnet
<subnet> range <n> start <address>
```

Задает начальный адрес <*address*> из диапазона адресов для DHCP сервера с идентификатором диапазона <*n*>. Адреса для аренды будут выделяться из пула адресов <*subnet*>.

```
set service dhcp-server shared-network-name <name> subnet
<subnet> range <n> stop <address>
```

Задает конечный адрес <*address*> из диапазона адресов для DHCP сервера с идентификатором диапазона <*n*>. Адреса для аренды будут выделяться из пула адресов <*subnet*>.

```
set service dhcp-server shared-network-name <name> subnet
<subnet> exclude <address>
```

Задает адрес <*address*>, который будет исключен для выдачи DHCP сервером из подсети <*subnet*>. Поддерживается возможность указать несколько адресов для исключения.

```
set service dhcp-server shared-network-name <name> subnet
<subnet> domain-name <domain-name>
```

Устанавливает доменное имя <*domain-name*> для хостов, которые будут получать адреса из подсети <*name*>. Доменное имя будет добавлено к имени хоста клиента для формирования полностью определенного доменного имени (FQDN).

```
set service dhcp-server shared-network-name <name> subnet
<subnet> domain-search <domain-name>
```

Задает дополнительные доменные имена <*domain-name*> , которые образуют список доменов, используемых при поиске хостов в локальной сети по доменному имени. Если эта опция не указана, то по умолчанию используется единственный домен, указанный в параметре domain-name.

```
set service dhcp-server shared-network-name <name> subnet
<subnet> ping-check
```

Активирует механизм проверки ICMP для подсети <*subnet*> из которой DHCP сервер будет выдавать адреса.

```
set service dhcp-server shared-network-name <name> subnet
<subnet> enable-failover
```

Активирует механизм отказоустойчивости службы DHCP для пула адресов <*subnet*>.

Настройка механизма отказоустойчивости DHCP сервера IPv4

```
set service dhcp-server failover source-address <address>
```

Устанавливает локальный адрес <*address*>, используемый при обмене данными с резервным DHCP сервером на котором настроен механизм отказоустойчивости.

```
set service dhcp-server failover remote <address>
```

Устанавливает адрес резервного DHCP сервера <*address*>, используемый при обмене данными с основным DHCP сервером на котором настроен механизм отказоустойчивости.

- `set service dhcp-server failover name <name>`

Задает общее имя <*name*> кластера отказоустойчивости DHCP сервера.

> **Примечание**
>
> *Общее имя <name> должно быть идентичным у всех участников кластера отказоустойчивости DHCP сервера.*

- `set service dhcp-server failover status <primary | secondary>`

Задает роль DHCP сервера <*primary* | *secondary*> в кластере отказоустойчивости.

> **Примечание**
>
> *Для того чтобы основной и резервный DHCP сервера могли синхронизировать свои таблицы арендованных адресов, они должны иметь возможность обращаться друг к другу по TCP порту 647.*

> **Подсказка**
>
> *Обмен таблицами арендованных адресов между участниками кластера отказоустойчивости DHCP сервера не шифруется и не аутентифицируется. Поскольку большинство DHCP серверов находятся в защищенной локальной сети организации, это будет избыточным. Если обмен информацией между участниками кластера отказоустойчивости DHCP сервера проходит через незащищенные каналы связи, рекомендуется использовать защищенный VPN туннель, чтобы гарантировать, что передача данных не будет скомпрометирована.*

Настройка статической привязки для DHCP сервера IPv4

Можно задать статическое назначение DHCP для каждого узла. Для этого потребуется MAC-адрес станции и желаемый IP-адрес. Адрес должен находиться внутри определения подсети, но может выходить за пределы оператора диапазона.

- `set service dhcp-server shared-network-name <name> subnet <subnet> static-mapping <description> mac-address <address>`

Создает статическую привязку DHCP <*description*>, которая будет действовать для хоста, идентифицируемого по его MAC-адресу <*address*>.

- `set service dhcp-server shared-network-name <name> subnet <subnet> static-mapping <description> ip-address <address>`

Задает адрес узла для статической привязки DHCP <*description*>. Адрес должен находиться внутри заданной подсети <*subnet*>, но может выходить за пределы динамического диапазона, созданного с помощью команды ***set service dhcp-server shared-network-name <name> subnet <subnet> range <n>***. Если адрес не указан, то используется адрес из динамического пула.

Мониторинг и эксплуатация DHCP сервера IPv4

- `show log dhcp server`

Выводит на экран журнал событий службы DHCP сервера

- `show log dhcp client`

Выводит на экран журнал событий всех процессов DHCP клиента

- `show log dhcp client interface <interface>`

Выводит на экран журнал событий процесса DHCP клиента для указанного интерфейса <interface>.

- `restart dhcp server`

Перезагружает службу DHCP сервера.

- `show dhcp server statistics`

Выводит на экран статистику работы службы DHCP сервера.

- `show dhcp server statistics pool <pool>`

Выводит на экран статистику работы службы DHCP сервера для указанного пула <pool>.

- `show dhcp server leases`

Выводит на экран информацию обо всех арендованных адресах, выданных DHCP сервером.

> **Подсказка**
>
> *При использовании команды show dhcp server leases информация о статической привязке адреса к определенному хосту не отображается. Чтобы получить информацию обо всех арендованных адресах, используйте команду show dhcp server leases state all.*

- `show dhcp server leases pool <pool>`

Выводит на экран информацию обо всех арендованных адресах, выданных DHCP сервером для указанного пула *<pool>*.

- `show dhcp server leases sort <key>`

Сортирует вывод информации обо всех арендованных адресах по указанному ключу *<key>*. Возможные варианты ключей: ip, hardware_address, state, start, end, remaining, pool, hostname (по умолчанию = ip)

- `show dhcp server leases state <state>`

Выводит информацию обо всех арендованных адресах с указанным состоянием *<state>*. Возможные состояния: all, active, free, expired, released, abandoned, reset, backup (по умолчанию = active)

DHCP сервер IPv6

DHCPv6 - версия протокола DHCP для работы с IPv6. Этот протокол назначает как IPv6 адреса, так и другие параметры настройки сети, такие, как адрес DNS или доменное имя. DHCPv6 может назначать IPv6 адреса через ретранслятор. DHCPv6 сервер также может обеспечить сервис DHCPv6 без состояния отслеживания SLAAC, при котором клиенту могут быть назначены параметры конфигурации, такие как адрес DNS-сервера и доменное имя без назначения IPv6-адреса.

Пример настройки пула адресов для DHCP сервера IPv6

В следующем примере описан один из сценариев настройки операционной системы Факел в качестве сервера DHCPv6.

Параметры для настройки сервера DHCPv6:

- Публикуемая сеть с именем *NET1* обслуживает подсеть *2001:db8::/64*.
- Подсеть *2001:db8::/64* подключена к интерфейсу *eth1*
- Адрес DNS сервера *2001:db8::ffff*
- Пул адресов будет состоять из диапазона *2001:db8::100 - 2001:db8::199*
- Значение для времени аренды адресов будет оставлено по умолчанию и составит 24 часа

Список команд для настройки сервера DHCPv6:

```
set service dhcpv6-server shared-network-name 'NET1' subnet
2001:db8::/64 address-range start 2001:db8::100 stop
2001:db8::199
set service dhcpv6-server shared-network-name 'NET1' subnet
2001:db8::/64 name-server 2001:db8::ffff
```

Конфигурация операционной системы будет выглядеть следующим образом:

```
fakel@fakel:~$ show service dhcpv6-server
    shared-network-name NET1 {
        subnet 2001:db8::/64 {
            address-range {
                start 2001:db8::100 {
                    stop 2001:db8::199
                }
            }
            name-server 2001:db8::ffff
        }
    }
```

Пример настройки статической привязки для DHCP сервера IPv6

В следующем примере описан сценарий настройки для сопоставления конкретных IPv6-адресов с конкретными хостами, через создание статической привязки.

Параметры для настройки сервера DHCPv6:

- Для сопоставления адреса с узлом используется IPv6 адрес *2001:db8::101*
- Для сопоставления адреса с узлом используется IPv6 префикс *2001:db8:0:101::/64*
- Для узла будет назначено имя *client1*

> **Подсказка**
>
> *Идентификатор представляет собой DUID устройства: разделенный двоеточием шестнадцатеричный список. Если устройство уже имеет динамическую аренду у*

> *DHCPv6 сервера, его DUID можно найти с помощью команды show service dhcpv6 server leases. DUID начинается с 5-го октета (после 4-го двоеточия) IAID_DUID.*

Список команд для настройки сервера DHCPv6:

```
set service dhcpv6-server shared-network-name 'NET1' subnet 2001:db8::/64 static-mapping client1 ipv6-address 2001:db8::101
set service dhcpv6-server shared-network-name 'NET1' subnet 2001:db8::/64 static-mapping client1 ipv6-prefix 2001:db8:0:101::/64
set service dhcpv6-server shared-network-name 'NET1' subnet 2001:db8::/64 static-mapping client1 identifier 00:01:00:01:12:34:56:78:aa:bb:cc:dd:ee:ff
```

Конфигурация операционной системы будет выглядеть следующим образом:

```
fakel@fakel:~$ show service dhcp-server shared-network-name NET1
   shared-network-name NET1 {
       subnet 2001:db8::/64 {
          name-server 2001:db8:111::111
          address-range {
              start 2001:db8::100 {
                 stop 2001:db8::199 {
              }
          }
          static-mapping client1 {
             ipv6-address 2001:db8::101
             identifier 00:01:00:01:12:34:56:78:aa:bb:cc:dd:ee:ff
          }
       }
    }
```

Основные настройки DHCP сервера IPv6

```
set service dhcpv6-server preference <preference value>
```

Задает приоритет *<preference value>* для выбора DHCP сервера. Приоритет задается в случае если клиенты, получающие сообщения об аренде адреса от нескольких DHCP серверов. Клиент выбирают DHCP сервер с наибольшим значением приоритета. Значение задается в диапазоне 0 - 255.

```
set service dhcpv6-server shared-network-name <name> subnet
<prefix> lease-time {default | maximum | minimum}
```

Задает время аренды *{default | maximum | minimum}* DHCPv6 сервера. По умолчанию время аренды DHCPv6 сервера составляет 24 часа. Это значение можно изменить, указав время по умолчанию, максимальное время и минимальное время. Все значения должны быть указаны в секундах.

```
set service dhcpv6-server shared-network-name <name> subnet
<prefix> nis-domain <domain-name>
```

Задает домен сетевой информационной системы NIS *<domain-name>* для клиентов DHCPv6.

```
set service dhcpv6-server shared-network-name <name> subnet
<prefix> nisplus-domain <domain-name>
```

Задает домен сетевой информационной системы NIS+ *<domain-name>* для клиентов DHCPv6.

```
set service dhcpv6-server shared-network-name <name> subnet
<prefix> nis-server <address>
```

Устанавливает адрес сервера сетевой информационной системы NIS *<address>* для клиентов DHCPv6.

```
set service dhcpv6-server shared-network-name <name> subnet
<prefix> nisplus-server <address>
```

Устанавливает адрес сервера сетевой информационной системы NIS+ *<address>* для клиентов DHCPv6.

```
set service dhcpv6-server shared-network-name <name> subnet
<prefix> sip-server <address | fqdn>
```

Устанавливает адрес SIP сервера *<address | fqdn>* для клиентов DHCPv6.

```
set service dhcpv6-server shared-network-name <name> subnet
<prefix> sntp-server-address <address>
```

Устанавливает адрес SNTP сервера <*address* | *fqdn*> для клиентов DHCPv6.

Настройка индивидуальных префиксов для клиентов DHCP сервера IPv6

```
set service dhcpv6-server shared-network-name <name> subnet
<prefix> prefix-delegation start <address> prefix-length
<length>
```

Устанавливает размер префиксов <*length*> для клиентов в подсети <*prefix*> при их запросе на делегирование префиксов.

```
set service dhcpv6-server shared-network-name <name> subnet
<prefix> prefix-delegation start <start-address> stop <stop-
address>
```

Устанавливает начальный <*start-address*> и конечный <*stop-address*> адреса для клиентов в подсети <prefix> при их запросе на делегирование префиксов.

Мониторинг состояния DHCP сервера IPv6

```
show log dhcpv6 server
```

Выводит на экран журнал событий службы DHCPv6 сервера

```
show log dhcpv6 client
```

Выводит на экран журнал событий всех процессов DHCPv6 клиента

```
show log dhcpv6 client interface <interface>
```

Выводит на экран журнал событий процесса DHCPv6 клиента для указанного интерфейса <*interface*>.

```
restart dhcpv6 server
```

Перезагружает службу DHCP сервера.

```
show dhcpv6 server status
```

Выводит на экран информацию о статусе работы DHCPv6 сервера.

```
show dhcpv6 server leases
```

Выводит на экран информацию обо всех арендованных адресах, выданных DHCPv6 сервером.

Подсказка

При использовании команды show dhcpv6 server leases информация о статической привязке адреса к определенному хосту не отображается. Чтобы получить информацию обо всех арендованных адресах, используйте команду show dhcpv6 server leases state all.

- `show dhcpv6 server leases pool <pool>`

Выводит на экран информацию обо всех арендованных адресах, выданных DHCPv6 сервером для указанного пула *<pool>*.

- `show dhcpv6 server leases sort <key>`

Сортирует вывод информации обо всех арендованных адресах по указанному ключу *<key>*.

Возможные варианты ключей: ip, hardware_address, state, start, end, remaining, pool, hostname (по умолчанию = ip)

- `show dhcpv6 server leases state <state>`

Выводит информацию обо всех арендованных адресах с указанным состоянием *<state>.*

Возможные состояния: all, active, free, expired, released, abandoned, reset, backup (по умолчанию = active)

Перенаправляющий DNS сервер

Общая информация о перенаправляющем DNS сервере

Перенаправляющий DNS сервер — перенаправляет полученные рекурсивные запросы вышестоящему кэширующему серверу в виде рекурсивных запросов. Используется преимущественно для снижения нагрузки на кэширующий DNS сервер. Перенаправляющий (Forwarding) DNS-сервер имеет следующие свойства:

- Способность обрабатывать рекурсивные запросы без выполнения самой рекурсии;
- Предоставлять локальный кэш в ближайшем сетевом расположении;
- Увеличивает гибкость в определении локального доменного пространства.

В операционной системе Факел перенаправляющий DNS сервер не требует наличия вышестоящего DNS сервера. Он может выступать как в качестве полноценного рекурсивного DNS сервера, так и пересылать запросы на настраиваемые вышестоящие DNS серверы.

Пример настройки Split DNS

Пример настройки Split DNS для домена example.com.

Для настройки разделенного DNS будут использоваться следующие параметры:

- Все DNS запросы для домена example.com должны перенаправляться на DNS сервер по адресу *192.0.2.254* и *2001:db8:cafe::1*
- Все остальные DNS запросы будут перенаправлены на другой набор DNS серверов по адресам *192.0.2.1*, *192.0.2.2*, *2001:db8::1:ffff* и *2001:db8::2:ffff*
- Перенаправляющий DNS сервер будет прослушивать запросы только по адресам внутреннего интерфейса *eth1* - *192.168.1.254* для IPv4 и *2001:db8::ffff* для IPv6.
- Перенаправляющий DNS сервер будет принимать запросы на поиск только из внутренних подсетей - *192.168.1.0/24* и *2001:db8::/64*
- Перенаправляющий DNS сервер будет передавать обратные запросы на поиск для зон *10.in-addr.arpa*, *168.192.in-addr.arpa*, *16-31.172.in-addr.arpa* на вышестоящий сервер

Список команд для настройки Split DNS:

```
set service dns forwarding domain example.com server 192.0.2.254
set service dns forwarding domain example.com server 2001:db8:cafe::1
set service dns forwarding name-server 192.0.2.1
set service dns forwarding name-server 192.0.2.2
set service dns forwarding name-server 2001:db8::1:ffff
set service dns forwarding name-server 2001:db8::2:ffff
set service dns forwarding listen-address 192.168.1.254
set service dns forwarding listen-address 2001:db8::ffff
set service dns forwarding allow-from 192.168.1.0/24
set service dns forwarding allow-from 2001:db8::/64
set service dns forwarding no-serve-rfc1918
```

Основные настройки перенаправляющего DNS сервера

```
set service dns forwarding system
```

Перенаправляет входящие DNS запросы на DNS сервера, которые настроенные в операционной системы.

```
set service dns forwarding dhcp <interface>
```

Определяет интерфейс *<interface>*, на который DHCP клиенты направляют DNS запросы.

```
set service dns forwarding name-server <address>
```

Задает адрес DNS сервера *<address>*, на который будут передаваться все DNS запросы. Поддерживается возможность указать несколько адресов для использования различных DNS серверов.

```
set service dns forwarding domain <domain-name> server
<address>
```

Перенаправляет полученные запросы для определенного домена *<domain-name>* на заданный DNS сервер *<address>*. Поддерживается возможность указать несколько адресов для использования различных DNS серверов. Этот параметр можно использовать для настройки разделенного DNS.

```
set service dns forwarding domain <domain-name> addnta
```

Активирует механизм NTA для определенного домена *<domain-name>*. Этот параметр должен быть установлен, если домен не поддерживает DNSSEC.

```
set service dns forwarding domain <domain-name> recursion-
desired
```

Устанавливает RD бит для запросов к вышестоящему DNS серверу.

```
set service dns forwarding allow-from <network>
```

Задает адрес сети *<network>*, в которой узлам разрешено использовать перенаправляющий DNS сервер.

```
set service dns forwarding dnssec <off | process-no-validate |
process | log-fail | validate>
```

Устанавливает уровень для параметра DNSSEC *<off | process-no-validate | process | log-fail | validate>.*

Перенаправляющий DNS сервер имеет 5 различных уровней обработки DNSSEC:

- **off** - В этом режиме обработка DNSSEC не производится. Перенаправляющий DNS сервер не будет устанавливать бит DNSSEC OK (DO) в исходящих запросах и будет игнорировать биты DO и AD в запросах.

- **process-no-validate** - В этом режиме перенаправляющий DNS сервер будет устанавливать DO-бит в исходящих запросах и предоставлять RR-наборы, связанные с DNSSEC (NSEC, RRSIG), клиентам, которые их запрашивают (с помощью DO-бита в запросе), за исключением зон, предоставляемых с помощью настройки зон для авторизации. В этом режиме он не будет выполнять никаких проверок, даже по запросу клиента.
- **process** - Если для параметра dnssec установлено значение process, то поведение перенаправляющего DNS сервера будет аналогично поведению в режиме process-no-validate. Однако перенаправляющий DNS сервер попытается проверить данные, если в запросе установлен хотя бы один из битов DO или AD. В этом случае он установит в ответе бит AD, если данные проверены успешно, или пошлет SERVFAIL, если проверка окажется ошибочной.
- **log-fail** - В этом режиме перенаправляющий DNS сервер будет пытаться проверить все данные, получаемые с доверенных серверов, независимо от запросов клиента по DNSSEC, и записывать результат проверки в журнал. Этот режим можно использовать для определения дополнительной нагрузки и количества возможных фиктивных ответов перед включением полноценной проверки. Ответы на запросы клиента аналогичны процессу.
- **validate** - Наивысший режим обработки DNSSEC. В этом режиме все запросы будут проверяться и в случае обнаружения не легитимных данных будет отправлен ответ SERVFAIL, независимо от запроса клиента.

Примечание

Популярный инструмент Dig для Unix/Linux устанавливает AD-бит в запросе. Это может привести к неожиданным результатам запроса при тестировании. В этом случае нужно указать параметр +noad в командной строке dig.

CD бит корректно выполняется для process и validate. Для log-fail сбои также будут регистрироваться.

```
set service dns forwarding ignore-hosts-file
```

Отключает поиска по имени узла из файла */etc/hosts* для перенаправляющего DNS сервера.

```
set service dns forwarding cache-size <0-2147483647>
```

Задает максимальное значение для количества записей DNS кэша *<0-2147483647>*. По умолчанию для максимального количества записей установлено значение *10000*.

```
set service dns forwarding negative-ttl <0-7200>
```

Задает максимальное время кэширования *<0-7200>* для отрицательных записей. По умолчанию для максимального времени кэширования установлено значение *3600* секунд.

Примечание

Запрос, на который нет подтвержденного ответа, кэшируется, чтобы впоследствии быстро исключить DNS запись, не создавая большой нагрузки на удаленный сервер.

- `set service dns forwarding listen-address <address>`

Устанавливает локальные адреса IPv4 или IPv6 <address>, к которым будет привязан перенаправляющий DNS сервер. Этого адреса DNS сервер будет использовать для прослушивания входящих соединений.

- `set service dns forwarding source-address <address>`

Устанавливает локальные адреса IPv4 или IPv6 *<address>*, используемые в качестве адреса источника для отправки запросов. С этого адреса DNS сервер будет отправлять перенаправленные на него исходящие DNS запросы.

- `set service dns forwarding no-serve-rfc1918`

Активирует параметр, который делает перенаправляющий DNS сервер неизвестным для зон: *10.in-addr.arpa*, *168.192.in-addr.arpa*, *16-31.172.in-addr.arpa*, что позволяет использовать вышестоящие DNS-серверы для обратного поиска этих зон.

Мониторинг и эксплуатация перенаправляющего DNS сервера

- `reset dns forwarding <all | domain>`

Сбрасывает локальную базу данных кэша переадресации DNS. Можно сбросить кэш для всех записей или только для записей, относящихся к определенному домену.

- `restart dns forwarding`

Перезапускает процесс перенаправляющего DNS сервера. При этом также аннулируется локальный кэш переадресации DNS.

- `show dns forwarding`

Выводит на экран статистику работы перенаправляющего DNS сервера.

Динамический DNS

Общая информация о динамическом DNS

Операционная система Факел умеет обновлять удаленную DNS запись, когда интерфейс получает новый IP адрес. Для этого в состав операционной системы Факел входит клиент *ddclient*.

Клиент *ddclient* использует два метода обновления DNS записей. Первый из них, в соответствии с *RFC 2136*, посылает обновления непосредственно демону DNS. Во втором случае используется сторонний сервис, например DynDNS.com или любой другой подобный сайт. Этот метод использует HTTP запросы для передачи нового IP адреса. В операционной системе Факел можно настроить оба способа.

Пример настройки динамического DNS

Пример настройки динамической смены DNS записи.

Для настройки динамической смены DNS выполните следующие настройки:

- Зарегистрируйте DNS запись *example.fakel.io* на DNS сервере *ns1.fakel.io*.
- Используйте файл ключей аутентификации по адресу */config/auth/my.key*.
- Установить TTL равным *300* секундам

```
fakel@fakel# show service dns dynamic
 interface eth0.7 {
     rfc2136 Fakel-DNS {
         key /config/auth/my.key
         record example.fakel.io
         server ns1.fakel.io
         ttl 300
         zone fakel.io
     }
 }
```

В результате будет получена следующая запись конфигурации ddclient:

```
#
```

```
# ddclient configuration for interface "eth0.7":

#

use=if, if=eth0.7


# RFC2136 dynamic DNS configuration for example.vyos.io.vyos.io

server=ns1.vyos.io

protocol=nsupdate

password=/config/auth/my.key

ttl=300

zone=vyos.io

example.vyos.io
```

Примечание

Можно также поддерживать в актуальном состоянии различные зоны DNS. Для этого достаточно создать новый узел конфигурации: set service dns dynamic interface <interface> rfc2136 <other-service-name>.

Основные настройки динамического DNS

- `set service dns dynamic interface <interface> rfc2136 <service-name>`

Создает новую конфигурацию обновления DNS для RFC 2136, которая будет обновлять IP-адрес, назначенный <*interface*> на сервисе, настроенном в разделе <*service-name*>.

- `set service dns dynamic interface <interface> rfc2136 <service-name> key <keyfile>`

Определяет файл <*keyfile*>, содержащий секретный ключ RNDC, совместно используемый с удаленным DNS сервером. Это опция является обязательной.

- `set service dns dynamic interface <interface> rfc2136 <service-name> server <server>`

Задает интерфейс, имя сервиса и адрес DNS сервера, используемые при обновлении данного динамического назначения.

- `set service dns dynamic interface <interface> rfc2136 <service-name> zone <zone>`

Задает обновляемую DNS <*zone*>. Это опция является обязательной.

- `set service dns dynamic interface <interface> rfc2136 <service-name> record <record>`

Задает DNS запись <*record*>, которая должна обновляться. Этот параметр может быть задан несколько раз. Это опция является обязательной.

- `set service dns dynamic interface <interface> rfc2136 <service-name> ttl <ttl>`

Задает значение TTL <*ttl*> для записи ресурса. Значение по умолчанию равно 600 секундам.

Настройка сервисов на основе HTTP

- `set service dns dynamic interface <interface> service <service> host-name <hostname>`

Определяет динамическое имя хоста <*hostname*>, идентифицированным по определенному сервису <*service*>, при изменении IP адреса на интерфейсе <*interface*>.

- `set service dns dynamic interface <interface> service <service> login <username>`

Задает имя пользователя <*username*>, используемое при аутентификации запроса на обновление для службы определенной в поле <*service*>.

- `set service dns dynamic interface <interface> service <service> password <password>`

Задает пароль <*password*>, используемый при аутентификации запроса на обновление для службы определенной в поле <*service*>.

- `set service dns dynamic interface <interface> service <service> protocol <protocol>`

Определяет протокол <*protocol*>, используемый при аутентификации запроса на обновление для службы определенной в поле <*service*>.

- `set service dns dynamic interface <interface> service <service> server <server>`

Определяет сервер <*server*>, используемый при аутентификации запроса на обновление для службы определенной в поле <*service*>.

```
set service dns dynamic interface <interface> service <service>
zone <zone>
```

Задает зону DNS <*zone*> для обновления. Это параметр доступен только для CloudFlare.

Настройка для работы с NAT

По умолчанию *ddclient* обновляет динамическую dns-запись, используя IP-адрес, непосредственно подключенный к интерфейсу. Если ваше устройство находится за NAT, то запись будет обновлена и будет указывать на ваш внутренний IP адрес.

Чтобы *ddclient* определял внешние IP адреса необходимо выполнить следующие настройки:

```
set service dns dynamic interface <interface> use-web url <url>
```

Задает значение параметра <*url*> для определения вашего IP адреса. *ddclient* загрузит <url> и попытается извлечь ваш IP адрес из ответа.

```
set service dns dynamic interface <interface> use-web skip
<pattern>
```

Задает значение параметра <*pattern*>. Клиент *ddclient* будет пропускать все адреса, расположенные до строки, заданной в поле <*pattern*>.

Системные механизмы

Системный DNS

В этом разделе описывается настройка DNS в системе, а именно:

- Серверы имен DNS;
- Порядок поиска доменов.

Для того чтобы система могла использовать и заполнять неквалифицированные имена хостов, можно определить список, который будет использоваться для поиска доменов.

Пример настройки системного DNS

В данном примере используется несколько серверов OpenNIC, два IPv4 адреса и два IPv6 адреса.

Список команд для системного DNS:

```
set system name-server 176.9.37.132
set system name-server 195.10.195.195
set system name-server 2a01:4f8:161:3441::1
set system name-server 2a00:f826:8:2::1
set system domain-search domain fakel.io
set system domain-search domain fakel.net
set system domain-search domain fakel.network
```

Система настроена на попытку завершения домена в следующем порядке: *fakel.io* (первый), *fakel.net* (второй) и *fakel.network* (последний).

Основные настройки системного DNS

```
set system name-server <address>
```

С помощью этой команды можно указать DNS сервер для системы, который будет использоваться для поиска DNS. Можно добавить несколько DNS серверов, настраивая по одному. Поддерживаются адреса как IPv4, так и IPv6.

```
set system domain-search domain <domain>
```

С помощью этой команды можно определить домены по одному, чтобы система использовала их для завершения неквалифицированных имен хостов. Максимум: 6 записей.

Примечание

Доменные имена могут включать буквы, цифры, дефисы и точки, максимальная длина которых составляет 253 символа.

Протокол NTP

Общая информация о протоколе NTP

NTP (Network Time Protocol) - сетевой протокол для синхронизации часов между компьютерными системами в сетях передачи данных с пакетной коммутацией и переменной задержкой. NTP, работающий с 1985 года, является одним из старейших протоколов Интернета, используемых в настоящее время.

NTP предназначен для синхронизации всех компьютеров-участников с точностью до нескольких миллисекунд от UTC. Для выбора серверов точного времени используется алгоритм пересечения, представляющий собой модифицированную версию алгоритма Марзулло, и предназначенный для уменьшения влияния переменных сетевых задержек. Обычно NTP может поддерживать время с точностью до десятков миллисекунд в общедоступном Интернете, а в локальных сетях при идеальных условиях может достигать точности более одной миллисекунды. Асимметричные маршруты и перегруженность сети могут приводить к ошибкам в 100 мс и более.

Протокол обычно описывается в терминах модели «клиент-сервер», но может также легко использоваться в одноранговых отношениях, когда оба участника рассматривают друг друга в качестве потенциального источника времени. Реализации передают и получают временные метки с помощью UDP-порта с номером 123.

NTP предупреждает о предстоящей корректировке високосной секунды, но информация о локальных часовых поясах или переходе на летнее время не передается.

В настоящее время используется протокол версии 4 (NTPv4), который является предлагаемым стандартом и документирован в RFC 5905. Он обратно совместим с версией 3, описанной в RFC 1305.

Основные настройки протокола NTP

- `set system ntp server <address>`

Настройте один или несколько серверов для синхронизации. Имя сервера может быть как IP адресом, так и FQDN.

По умолчанию установлено 3 NTP-сервера, которые можно изменить:

- *0.pool.ntp.org*
- *1.pool.ntp.org*

- *2.pool.ntp.org*

```
system ntp server <address> <noselect | pool | preempt |
prefer>
```

Настройте один или несколько атрибутов для данного NTP сервера:

- **noselect** помечает сервер как неиспользуемый, за исключением целей отображения. Сервер отбрасывается алгоритмом выбора.
- **pool** мобилизует постоянную ассоциацию клиентского режима с рядом удаленных серверов.
- **preempt** вытесняемая ассоциация является расходуемой.
- **prefer** помечает сервер как предпочтительный. При прочих равных условиях именно этот хост будет выбран для синхронизации среди множества корректно работающих хостов.

```
system ntp listen-address <address>
```

Процесс NTP будет прослушивать только указанный IP-адрес. Вы должны указать *<address>* и, опционально, разрешенных клиентов. Можно настроить несколько адресов прослушивания.

```
system ntp allow-clients address <address>
```

Список сетей или клиентских адресов, которым разрешено связываться с данным NTP сервером. Можно настроить несколько сетей.

```
system ntp vrf <name>
```

Задает NTP сервер для экземпляра VRF.

Информация об устройстве

В этом разделе описывается информация о хостах системы и способы их настройки, рассматриваются следующие темы:

- Имя устройства;
- Домен;
- IP-адрес;
- Псевдонимы.

Имя устройства

Имя устройства — это метка, присваиваемая сетевому устройству в сети и используемая для отличия одного устройства от другого в определенных сетях или в Интернете. С другой стороны, это будет имя, которое отображается в приглашении командной строки.

Доменное имя

Доменное имя — это метка, присвоенная компьютерной сети, и поэтому оно уникально. Операционная система Факел добавляет доменное имя в качестве суффикса к любому неквалифицированному имени. Например, если вы задали доменное имя example.com, а отправляете ICMP запрос на неквалифицированное имя crux, то операционная система Факел квалифицирует это имя как crux.example.com.

Статическое сопоставление имени устройства

В операционной системе Факел поддерживается возможность сопоставить IP адрес с именем устройства для локального разрешения имен. Это эквивалент записей файла */etc/hosts*.

> **Примечание**
>
> *Не редактируйте вручную файл /etc/hosts. Этот файл будет автоматически регенерироваться при загрузке на основе настроек, указанных в данном разделе, а значит, все внесенные вручную правки будут потеряны. Вместо этого настройте статическое сопоставление хостов следующим образом.*

Список команд для настройки информации об устройстве

- `set system host-name <hostname>`

Устанавливает имя устройства *<hostname>*. Имя устройства может содержать до 63 символов. Имя устройства должно начинаться и заканчиваться буквой или цифрой и иметь в качестве внутренних символов только буквы, цифры или дефис.

По умолчанию используется имя хоста *fakel*.

- `system domain-name <domain>`

Устанавливает доменное имя системы. Доменное имя должно начинаться и заканчиваться буквой или цифрой и иметь в качестве внутренних символов только буквы, цифры или дефис.

- `set system static-host-mapping host-name <hostname> inet <address>`

Создает статическое сопоставление имени устройства, которое всегда будет разрешать имя <*hostname*> в IP адрес <*address*>.

```
set system static-host-mapping host-name <hostname> alias
<alias>
```

Создает псевдоним <*alias*> для настроенного статического отображения имени устройства <*hostname*>. Таким образом, адрес, настроенный как ***set system static-host-mapping host-name <hostname> inet <address>***, может быть доступен через несколько имен. Для каждого имени хоста можно указать несколько псевдонимов.

Управление пользователями

Общая информация об управлении пользователями

Учетная запись пользователя ПО **Факел** по умолчанию *fakel*, а также вновь создаваемые учетные записи пользователей имеют все возможности для настройки системы. Все учетные записи имеют права *sudo* и, следовательно, могут работать в системе в качестве привилегированного пользователя.

Поддерживаются как локально администрируемые, так и удаленно администрируемые учетные записи RADIUS.

В крупных инсталляциях нецелесообразно настраивать каждого пользователя отдельно на каждой системе. ПО **Факел** поддерживает использование серверов RADIUS для аутентификации пользователей.

Аутентификация пользователей на основе ключа

Настоятельно рекомендуется использовать аутентификацию по ключу SSH. По умолчанию существует только один пользователь *fakel*, которому можно назначить любое количество ключей. Создать ssh-ключ можно с помощью команды ***ssh-keygen*** на локальной машине, которая (по умолчанию) сохранит его в файле *~/.ssh/id_rsa.pub*.

Каждый SSH-ключ состоит из трех частей:

ssh-rsa **|** *AAAAB3NzaC1yc2EAAAABAA...VBD5lKwEWB* **|** *username@host.example.com*

Используются только тип *ssh-rsa* и ключ *AAAB3N...*. Обратите внимание, что длина ключа обычно составляет несколько сотен символов, и его придется копировать и вставлять. Некоторые эмуляторы терминалов могут случайно разбить его на несколько строк. Будьте внимательны при вставке, чтобы он вставлялся только в одну строку. Третья часть является просто идентификатором и служит для справки.

Пример настройки аутентификации пользователя по ключу

В следующем примере и *User1*, и *User2* смогут подключаться к маршрутизатору под управлением ПО **Факел** по SSH под именем пользователя *fakel*, используя свои

собственные ключи. *User1* имеет ограничение на подключение только с одного IP адреса.

Список команд для настройки аутентификации пользователя по ключу:

```
set system login user fakel authentication public-keys 'User1' key "AAAAB3Nz...KwEW"
set system login user fakel authentication public-keys 'User1' type ssh-rsa
set system login user fakel authentication public-keys 'User1' options "from=&amp;quot;192.168.0.100&amp;quot;"
set system login user fakel authentication public-keys 'User2' key "AAAAQ39x...fbV3"
set system login user fakel authentication public-keys 'User2' type ssh-rsa
```

Основные настройки для управления пользователями

```
system login user <name> full-name <string>
```

Создает нового системного пользователя с именем пользователя <*name*> и реальным именем, заданным в поле <*string*>.

```
system login user <name> authentication plaintext-password <password>
```

Устанавливает пароль пользователя <*name*> в открытом виде для данной системы. Пароль в открытом виде будет автоматически переведен в защищенный хэшированный пароль и нигде не будет сохранен в открытом виде.

```
system login user <name> authentication encrypted-password <password>
```

Устанавливает зашифрованный пароль для заданного имени пользователя <*name*>. Это удобно для переноса хэшированного пароля из системы в систему.

Настройки аутентификации на основе ключей

```
system login user <username> authentication public-keys <identifier> key <key>
```

Назначает связку открытых ключей SSH <*key*>, идентифицированную по ключу <*identifier*>, локальному пользователю <*username*>.

- `system login user <username> authentication public-keys <identifier> type <type>`

Каждая часть открытого ключа SSH, на которую ссылается <*identifier*>, требует настройки <*type*> используемого открытого ключа. Этот тип может быть любым из:

- *ecdsa-sha2-nistp256*
- *ecdsa-sha2-nistp384*
- *ecdsa-sha2-nistp521*
- *ssh-dss*
- *ssh-ed25519*
- *ssh-rsa*

Примечание

Вы можете назначить несколько ключей одному и тому же пользователю, используя уникальный идентификатор для каждого SSH-ключа.

- `system login user <username> authentication public-keys <identifier> options <options>`

Задает параметры для данного открытого ключа <*identifier*>.

Настройка авторизации через RADIUS

- `set system login radius server <address> key <secret>`

Устанавливает <*address*> пользователя RADIUS-сервера с секретом pre-shared-secret, заданным в поле <*secret*>. Можно указать несколько серверов.

- `set system login radius server <address> port <port>`

Определяет порт <*port*>, через который можно связаться с сервером RADIUS <*address*>. По умолчанию это значение равно *1812*.

- `set system login radius server <address> timeout <timeout>`

Установите <*timeout*> в секундах при запросе к серверу RADIUS <*address*>.

- `set system login radius server <address> disable`

Временно отключает определенный сервер RADIUS <*address*>.

- `set system login radius source-address <address>`

Серверы RADIUS могут быть усилены путем разрешения подключения только определенных IP адресов. Для этого можно настроить адрес источника каждого запроса RADIUS. Если этот параметр не задан, то входящие соединения с сервером RADIUS будут использовать ближайший адрес интерфейса, указывающий на сервер, что приводит к ошибкам в сетях OSPF, например, при отказе канала связи и использовании резервного маршрута.

Примечание

Если вы хотите, чтобы пользователи-администраторы проходили аутентификацию через RADIUS, необходимо отправить атрибут Cisco-AV-Pair shell:priv-lvl=15. Без этого атрибута вы получите только обычных, непривилегированных пользователей системы.

Настройка баннера входа в систему

Вы можете установить баннер с сообщением после или перед входом в систему для отображения определенной информации для данной системы.

- `set system login banner pre-login <message>`

Устанавливает сообщение <*message*>, которое будет показано при подключении по SSH и перед входом пользователя в систему.

- `set system login banner post-login <message>`

Устанавливает сообщение <*message*>, которое будет показано после входа пользователя в систему.

Примечание

Чтобы создать новую строку в сообщении для входа в систему, необходимо экранировать символ новой строки с помощью \\\n.

Чтобы создать новую строку в сообщении для входа в систему, необходимо экранировать символ новой строки с помощью \\\n.

Эксплуатация

ПО **Факел** предоставляет администратору широкий спектр команд, выполняемых в эксплуатационном режиме, для получения информации о работе системы.

Управление лицензиями ПО Факел

Система управления лицензиями в ПО Факел обеспечивает возможность регистрации экземпляра программного обеспечения для целей учета и активации дополнительных возможностей.

Для активации ПО Факел используются лицензионные ключи, представляющие собой комбинацию букв и цифр, хранящихся в виде текстовых строк (лицензионных строк). Одна упомянутая строка может содержать один лицензионный ключ. Один ключ может активировать один экземпляр ПО Факел. Лицензионные ключи поставляются посредством электронных средств связи, таких как электронная почта.

Установка лицензии

Установка лицензии проходит в режиме конфигурирования ПО Факел.

- `set license add 'license_string'`

Команда позволяет установить лицензию для приобретенного экземпляра ПО Факел.

> **Примечание**
>
> *Если введена некорректная строка лицензии, то на экран будет выведено сообщение License is invalid and can not be imported*

Просмотр лицензий

Просмотр лицензий ПО Факел выполняется в режиме администрирования.

- `show license`

Команда позволяет вывести на экран текущие лицензии приобретенного экземпляра ПО Факел.

```
fakel@fakel:~$ show license
|      LicId       |     CliId     | LicType  | Status  | DaysLeft |
|------------------|---------------|----------|---------|----------   |
| 0                | 0             | trial    | active  | 29          |
Total number of licenses: 1
```

Удаление лицензий

Удаление лицензии проходит в режиме конфигурирования ПО Факел.

- `set license del 'license_id'`

Команда позволяет удалить установленную лицензию для приобретенного экземпляра ПО Факел.

Интерфейсы USB

Ранее последовательные интерфейсы были определены как устройства ttySx и ttyUSBx, где x - номер экземпляра последовательного интерфейса. Было установлено, что между загрузками распределение на шине последовательных USB интерфейсов может отличаться в зависимости от того, какой драйвер был загружен операционной системой первым. При наличии последовательных интерфейсов не только для подключения последовательной консоли, но и для поддержки функциональности интерфейсов WWAN данная особенность приводила к возникновению проблем в использовании устройств.

Для того, чтобы решить проблемы с определением устройств в системе, а также с учетом того, что до половины всех относительно недорогих конвертеров из USB в последовательный интерфейс не имеют встроенного на производстве номера, такие конвертеры определяются в системе как непосредственно подключенные к корневому концентратору USB и к шине соответственно. Такое поведение характерно для многих дистрибутивов операционных систем Linux последних версий.

- `show hardware usb`

Данная команда позволяет вывести на экран иерархически упорядоченную информацию обо всех подключенных USB устройствах.

> **Примечание**
>
> *Если устройство было отключено и подключено повторно, то ему будут присвоены новые идентификаторы Port, Dev и If.*

Пример работы команды show hardware usb:

```
fakel@fakel:~$ show hardware usb
/:  Bus 03.Port 1: Dev 1, Class=root_hub, Driver=ehci-pci/2p, 480M
   |__ Port 1: Dev 2, If 0, Class=Hub, Driver=hub/4p, 480M
      |__ Port 3: Dev 4, If 0, Class=Vendor Specific Class, Driver=qcserial, 480M
      |__ Port 3: Dev 4, If 2, Class=Vendor Specific Class, Driver=qcserial, 480M
      |__ Port 3: Dev 4, If 3, Class=Vendor Specific Class, Driver=qcserial, 480M
      |__ Port 3: Dev 4, If 8, Class=Vendor Specific Class, Driver=qmi_wwan, 480M
/:  Bus 02.Port 1: Dev 1, Class=root_hub, Driver=xhci_hcd/2p, 5000M
```

```
/:  Bus 01.Port 1: Dev 1, Class=root_hub, Driver=xhci_hcd/2p, 480M
   |__ Port 1: Dev 2, If 0, Class=Vendor Specific Class, Driver=pl2303, 12M
   |__ Port 2: Dev 3, If 0, Class=Hub, Driver=hub/4p, 480M
      |__ Port 4: Dev 5, If 2, Class=Vendor Specific Class, Driver=ftdi_sio, 480M
      |__ Port 4: Dev 5, If 0, Class=Vendor Specific Class, Driver=ftdi_sio, 480M
      |__ Port 4: Dev 5, If 3, Class=Vendor Specific Class, Driver=ftdi_sio, 480M
      |__ Port 4: Dev 5, If 1, Class=Vendor Specific Class, Driver=ftdi_sio, 480M
      |__ Port 3: Dev 4, If 0, Class=Hub, Driver=hub/4p, 480M
         |__ Port 3: Dev 6, If 0, Class=Hub, Driver=hub/4p, 480M
            |__ Port 4: Dev 8, If 2, Class=Vendor Specific Class, Driver=ftdi_sio, 480M
            |__ Port 4: Dev 8, If 0, Class=Vendor Specific Class, Driver=ftdi_sio, 480M
            |__ Port 4: Dev 8, If 3, Class=Vendor Specific Class, Driver=ftdi_sio, 480M
            |__ Port 4: Dev 8, If 1, Class=Vendor Specific Class, Driver=ftdi_sio, 480M
         |__ Port 4: Dev 7, If 3, Class=Vendor Specific Class, Driver=ftdi_sio, 480M
         |__ Port 4: Dev 7, If 1, Class=Vendor Specific Class, Driver=ftdi_sio, 480M
         |__ Port 4: Dev 7, If 2, Class=Vendor Specific Class, Driver=ftdi_sio, 480M
         |__ Port 4: Dev 7, If 0, Class=Vendor Specific Class, Driver=ftdi_sio, 480M
```

- `show hardware usb serial`

Данная команда позволяет вывести на экран список всех подключенных последовательных USB устройств. Имя устройства (например, usb0b2.4p1.0) может быть использовано при обращении к последовательной консоли.

Пример работы команды show hardware usb serial:

```
fakel@fakel$ show hardware usb serial
Device           Model               Vendor
------           ------              ------
usb0b1.3p1.0     MC7710              Sierra Wireless, Inc.
usb0b1.3p1.2     MC7710              Sierra Wireless, Inc.
usb0b1.3p1.3     MC7710              Sierra Wireless, Inc.
usb0b1p1.0       USB-Serial_Controller_D Prolific Technology, Inc.
usb0b2.3.3.4p1.0 Quad_RS232-HS       Future Technology Devices International, Ltd
usb0b2.3.3.4p1.1 Quad_RS232-HS       Future Technology Devices International, Ltd
usb0b2.3.3.4p1.2 Quad_RS232-HS       Future Technology Devices International, Ltd
usb0b2.3.3.4p1.3 Quad_RS232-HS       Future Technology Devices International, Ltd
usb0b2.3.4p1.0   Quad_RS232-HS       Future Technology Devices International, Ltd
usb0b2.3.4p1.1   Quad_RS232-HS       Future Technology Devices International, Ltd
usb0b2.3.4p1.2   Quad_RS232-HS       Future Technology Devices International, Ltd
usb0b2.3.4p1.3   Quad_RS232-HS       Future Technology Devices International, Ltd
usb0b2.4p1.0     Quad_RS232-HS       Future Technology Devices International, Ltd
usb0b2.4p1.1     Quad_RS232-HS       Future Technology Devices International, Ltd
usb0b2.4p1.2     Quad_RS232-HS       Future Technology Devices International, Ltd
usb0b2.4p1.3     Quad_RS232-HS       Future Technology Devices International, Ltd
```

Версия

- `show version`

Данная команда позволяет вывести на экран информацию о версии и номере сборки ПО **Факел**. Эта информация также включает кодовое обозначение релиза.

Пример работы команды show version:

```
fakel@fakel:~$ show version

Version:          Fakel 1.2.3-45678
Release Train:    zubr

Built by:         IT ROUTE
Built on:         Mon 01 Jan 2023 09:00 UTC
Build UUID:       8d9796d2-511e-4dea-be4f-cd4516c404f3
Build Commit ID:  2abc286ccff594

Architecture:     x86_64
Boot via:         installed image
System type:      VMware guest

Hardware vendor:  VMware, Inc.
Hardware model:   VMware Virtual Platform
Hardware S/N:     VMware-42 33 79 fe 73 64 2d 62-d5 62 ab 99 5a 3e d9 6d
Hardware UUID:    fe793352-6473-622d-d562-ab975a3ed96d

Copyright:        IT ROUTE
```

- `show version kernel`

Данная команда позволяет вывести на экран информацию о версии ядра операционной системы Linux, дистрибутив которой лежит в основе операционной системы на базе ПО **Факел**.

Пример работы команды show version kernel:

```
fakel@fakel:~$ show version kernel
5.4.128-amd64-vyos
```

- `show version frr`

Данная команда позволяет вывести на экран информацию о встроенной в операционную систему Факел подсистеме маршрутизации FRR, которая представляет собой архитектурный слой управления потоками данных (Control Plane) и является продуктом дальнейшего развития таких открытых (GNU) программных проектов маршрутизации, как Zebra и Quagga.

Пример работы команды show version frr:

```
fakel@fakel:~$ show version frr
FRRouting 7.5.1-20210625-00-gf07d935a2 (fakel).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

Опции загрузки

ПО **Факел** предоставляет ряд опций командной строки на уровне ядра операционной системы, которые позволяют внести изменения в штатный процесс загрузки системы. Чтобы добавить опцию, необходимо выбрать соответствующий образ в меню загрузчика операционной системы GRUB, нажать клавишу e, отредактировать первую строчку, затем использовать сочетание клавиш Ctrl + X, чтобы выполнить загрузку.

> **! Предупреждение**
>
> *Возможности, описание которых представлено в данном разделе, могут оказать существенное влияние на работу системы, например, их использование может привести к прерыванию работы основных подсистем в составе ПО **Факел**. Необходимо использовать данные возможности с осторожностью и только в случаях, когда это действительно необходимо.*

Произвольный файл конфигурации

ПО **Факел** можно сообщить путь до произвольного файла конфигурации вместо */config/config.boot*. Если указанный файл не существует или система не может его прочитать, происходит откат системы в конфигурации по умолчанию. При чтении файла система не проводит валидацию его структуры, поэтому очень важно обеспечить валидность файла конфигурации самостоятельно.

- `fakel-config=/path/to/file`

Для загрузки файла конфигурации по умолчанию, наличие которого предусмотрено в ПО **Факел** на производстве, необходимо указать следующее значение для данной опции:

- `fakel-config=/opt/fakel/etc/config.boot.default`

Отключение шагов загрузки

Использование данных опций приводит к пропуску определенных шагов при загрузке операционной системы на базе ПО **Факел**. Перед использованием опций рекомендуется детально ознакомиться с процессом загрузки, включая полный перечень шагов, из которых он состоит.

- `no-fakel-migrate`

При использовании данной опции не выполняется миграция конфигурации.

- `no-fakel-firewall`

При использовании данной опции не выполняется инициализация цепочек правил межсетевого экрана. Это приведет к тому, что вывод на экран конфигурации межсетевого экрана будет недоступен.

Восстановление пароля

Используя консоль, перезапустите маршрутизатор под управлением ПО **Факел**. После этого на экран будет выведено меню загрузчика операционной системы GRUB. Выберите необходимый пункт меню и нажмите клавишу «**Enter**». Такой пункт должен содержать в названии текст «*Lost password change*».

После этого операционной системой на базе ПО **Факел** будут запущены утилиты восстановления *stand-alone* и *user-password*, которые запросят подтверждение сброса пароля локального пользователя системы.

```
Do you wish to reset the admin password? (y or n)

y

Which admin account do you want to reset? [fakel]

my_username

Enter my_username password:

Retype my_username password:

System will reboot in 10 seconds...
```

Дисковый массив

Дисковый массив RAID - использует два или более жестких диска для улучшения скоростных показателей операций чтения и записи, размещения больших объемов данных или обеспечения отказоустойчивости. Механизм RAID поддерживает несколько схем организации накопителей, каждая из которых отличается комбинацией

характеристик емкости, надежности и производительности. ПО **Факел** поддерживает схему RAID 1. Схема RAID 1 позволяет использовать два или более жестких диска для зеркалирования областей хранения данных, чтобы обеспечить отказоустойчивость. В схеме RAID 1 каждый сектор одного диска дублируется сектором всех остальных дисков. Схема RAID 1 функционирует даже при наличии одного жесткого диска, то есть система продолжит работу в том числе при выполнении замены диска, если данная операция поддерживается аппаратной платформой. Схема RAID 1 может быть реализована либо с использование специализированного аппаратного обеспечения, либо программного.

ПО **Факел** поддерживает программную реализацию схемы RAID 1 из двух жестких дисков, а также предоставляет следующие возможности:

- обнаружение и уведомление об ошибках в работе жесткого диска;
- обеспечение работоспособности системы при сбоях одного из жестких дисков;
- обеспечение загрузки системы при сбоях одного из жестких дисков;
- замена жесткого диска после сбоев и инициализация повторной сборки дискового массива RAID;
- отслеживание статуса повторной сборки дискового массива RAID .

Варианты установки

Утилита установки операционной системы на базе ПО **Факел** предоставляет несколько вариантов установки на дисковый массив, собранный по схеме RAID 1:

- установка системы для создания дискового массива RAID 1;
- использование системных команд Linux для создания дискового массива RAID 1 перед установкой системы;
- использование предварительно созданного дискового массива RAID 1.

> **Примечание**
>
> *Перед окончательной установкой на платформу операционная система Факел работает в режиме live-установки.*

- `set interfaces ethernet <ethN> ipv6 address eui64 <prefix>`

Один диск и штатная установка

При установке операционной системы на базе ПО **Факел** утилита автоматически определяет наличие двух дисков, не являющихся частью дискового массива. В этом случае утилита установки предложит сконфигурировать дисковый массив RAID 1 и будет ожидать подтверждения:

Would you like to configure RAID 1 mirroring on them?

Если потребность в настройке дискового массива RAID 1 отсутствует, то необходимо ввести команду ***No*** в диалоге, после чего установка пойдет в штатном режиме.

Два и более пустых дисков

При установке операционной системы на базе ПО Факел утилита автоматически определяет наличие двух дисков, не являющихся частью дискового массива. В этом случае утилита установки предложит сконфигурировать дисковый массив RAID-1 и будет ожидать подтверждения:

Would you like to configure RAID 1 mirroring on them?

Для создания дискового массива RAID 1 необходимо ввести команду ***Yes*** в диалоге. Если утилита обнаруживает наличие некоторой файловой системы в разделах, которые должны быть использованы схемой RAID 1, то она запросит подтверждение продолжения процесса создания дискового массива RAID 1:

Continue creating array?

Если есть потребность перезаписать уже имеющуюся в разделах файловую систему, необходимо ввести команду ***Yes***. В этом случае утилита выведет предупреждение, что все данные на обоих жестких дисках будут стерты, и будет ожидать подтверждения данного действия:

Are you sure you want to do this?

Если есть потребность сохранить текущую конфигурацию операционной системы Факел после завершения ее установки, необходимо ввести команду ***Yes***. Если такой потребности нет, то необходимо ввести ***No***, после чего текущая конфигурация будет удалена:

Would you like me to save the data on it before I delete it?

После всех принятых решений установка продолжится в штатном режиме.

Уже имеющийся дисковый массив

При установке операционной системы Факел утилита автоматически определяет наличие сконфигурированного ранее дискового массива RAID 1 и предлагает использовать его:

Would you like to use this one?

Если есть потребность разобрать имеющийся дисковый массив RAID 1, необходимо ввести команду ***No***. После этого утилита установки автоматически определит наличие двух одинаковых жестких дисков и предложит сконфигурировать дисковый массив RAID 1, ожидая подтверждения:

Would you like to configure RAID 1 mirroring on them?

Если есть потребность отказаться от повторной сборки дискового массива RAID 1, необходимо ввести команду ***No***. После этого утилита установки запросит имя раздела, на который необходимо установить систему:

Which partition should I install the root on? [sda1]:

После указания имени раздела утилита установки запросит подтверждения сохранения текущей конфигурации ПО **Факел**:

Would you like me to save the data on it before I delete it?

Если есть потребность сохранить текущую конфигурацию ПО **Факел** после завершения ее установки, необходимо ввести команду ***Yes***. Если такой потребности нет, то необходимо ввести ***No***, после чего текущая конфигурация будет удалена.

После всех принятых решений установка продолжится в штатном режиме.

Обнаружение сбоев и замена диска

ПО **Факел** автоматически обнаруживает сбои диска в составе дискового массива RAID 1 и сообщает об этом посредством вывода сообщений в системной консоли. Удостовериться в сбое можно с помощью команды ***show raid***.

Для замены диска после сбоев в составе дискового массива RAID 1 необходимо выполнить следующие действия:

1. Программно исключить диск из дискового массива с помощью следующей команды:

   - `delete raid <raid-1-device> member <disk-partition>`

   Параметр *raid-1-device* - имя устройства в составе дискового массива RAID 1 (например, md0).

   Параметр *disk-parition* - имя раздела диска (например, sdb2)

2. Физически извлечь диск из системы. Если система не поддерживает возможность горячей замены дисков, потребуется сначала выключить систему.

3. Заменить диск на новый той же емкости или больше.

4. Форматировать новый диск перед добавлением в дисковый массив RAID 1 с помощью следующей команды:

   - `format disk <disk-device1> like <disk-device2>`

   Параметр *disk-device1* - имя нового диска (например, sdb).

   Параметр *disk-device2* - имя оставшегося в дисковом массиве устройства, функционирующего корректно (например, sda).

5. Программно включить новый диск в дисковый массив RAID 1 с помощью следующей команды:

   - `add raid <raid-1-device> member <disk-partition>`

Параметр *raid-1-device* - имя устройства в составе дискового массива RAID 1 (например, md0).

Параметр *disk-parition* - имя раздела диска (например, sdb2).

Эксплуатация

В данном разделе представлено описание возможностей по добавлению разделы диска в дисковый массив RAID 1, инициализации процесса зеркалирования, проверке текущего статуса и отображению подробной информации о дисковом массиве.

- `add raid <raid-1-device> member <disk-partition>`

Данная команда позволяет добавить раздел диска в состав дискового массива RAID 1. Добавление раздела в дисковый массив приведет к запуску процесса зеркалирования, при котором все данные на существующем разделе будут скопированы на новый раздел.

- `format disk <disk-device1> like <disk-device2>`

Данная команда позволяет подготовить диск к включению в предварительно собранный дисковый массив RAID 1 (устройство disk-device2 уже входит в состав дискового массива) посредством его форматирования.

- `show raid <raid-1-device>`

Данная команда позволяет вывести на экран общую информацию о дисковом массиве и его составе, а также информацию о ходе процесса зеркалирования.

Пример работы команды show raid <raid-1-device>:

```
fakel@fakel:~$ show raid md0
/dev/md0:
   Version : 00.90
   Creation Time : Wed Oct 29 09:19:09 2008
   Raid Level : raid1
   Array Size : 1044800 (1020.48 MiB 1069.88 MB)
   Used Dev Size : 1044800 (1020.48 MiB 1069.88 MB)
   Raid Devices : 2
   Total Devices : 2
   Preferred Minor : 0
   Persistence : Superblock is persistent
   Update Time : Wed Oct 29 19:34:23 2008
   State : active, degraded, recovering
```

```
   Active Devices : 1
   Working Devices : 2
   Failed Devices : 0
   Spare Devices : 1
   Rebuild Status : 17% complete
   UUID : 981abd77:9f8c8dd8:fdbf4de4:3436c70f
   Events : 0.103

   Number   Major   Minor   RaidDevice   State
   0        8       1       0            active sync        /dev/sda1
   2        8       17      1            spare rebuilding   /dev/sdb1
```

- `show disk <disk-device>`

Данная команда позволяет вывести на экран информацию о результатах форматирования жесткого диска.

Пример работы команды show disk <disk-device>:

```
fakel@fakel:~$ show disk sda format
Disk /dev/sda: 1073 MB, 1073741824 bytes
85 heads, 9 sectors/track, 2741 cylinders
Units = cylinders of 765 * 512 = 391680 bytes
Disk identifier: 0x000b7179
Device Boot      Start     End      Blocks       Id     System
/dev/sda1        6         2737     1044922+     fd     Linux raid autodet
```

Встроенные контейнеры

ПО **Факел** использует встроенный механизм контейнеризации без фоновых процессов, основанный на решении Podman.

Пример конфигурации

Далее представлен пример настроек на основе официальной документации на систему Zabbix с поправкой на синтаксис команд ПО **Факел**.

```
set container network zabbix-net prefix 172.20.0.0/16
set container network zabbix-net description 'Network for
Zabbix component containers'

set container name mysql-server image mysql:8.0
set container name mysql-server network zabbix-net

set container name mysql-server environment 'MYSQL_DATABASE'
value 'zabbix'
set container name mysql-server environment 'MYSQL_USER' value
'zabbix'
set container name mysql-server environment 'MYSQL_PASSWORD'
value 'zabbix_pwd'
set container name mysql-server environment
'MYSQL_ROOT_PASSWORD' value 'root_pwd'

set container name zabbix-java-gateway image zabbix/zabbix-
java-gateway:alpine-5.2-latest
set container name zabbix-java-gateway network zabbix-net

set container name zabbix-server-mysql image zabbix/zabbix-
server-mysql:alpine-5.2-latest
set container name zabbix-server-mysql network zabbix-net

set container name zabbix-server-mysql environment
'DB_SERVER_HOST' value 'mysql-server'
```

- set container name zabbix-server-mysql environment 'MYSQL_DATABASE' value 'zabbix'
- set container name zabbix-server-mysql environment 'MYSQL_USER' value 'zabbix'
- set container name zabbix-server-mysql environment 'MYSQL_PASSWORD' value 'zabbix_pwd'
- set container name zabbix-server-mysql environment 'MYSQL_ROOT_PASSWORD' value 'root_pwd'
- set container name zabbix-server-mysql environment 'ZBX_JAVAGATEWAY' value 'zabbix-java-gateway'

- set container name zabbix-server-mysql port zabbix source 10051
- set container name zabbix-server-mysql port zabbix destination 10051

- set container name zabbix-web-nginx-mysql image zabbix/zabbix-web-nginx-mysql:alpine-5.2-latest
- set container name zabbix-web-nginx-mysql network zabbix-net

- set container name zabbix-web-nginx-mysql environment 'MYSQL_DATABASE' value 'zabbix'
- set container name zabbix-web-nginx-mysql environment 'ZBX_SERVER_HOST' value 'zabbix-server-mysql'
- set container name zabbix-web-nginx-mysql environment 'DB_SERVER_HOST' value 'mysql-server'
- set container name zabbix-web-nginx-mysql environment 'MYSQL_USER' value 'zabbix'
- set container name zabbix-web-nginx-mysql environment 'MYSQL_PASSWORD' value 'zabbix_pwd'
- set container name zabbix-web-nginx-mysql environment 'MYSQL_ROOT_PASSWORD' value 'root_pwd'

- set container name zabbix-web-nginx-mysql port http source 80
- set container name zabbix-web-nginx-mysql port http destination 8080

Основные настройки для встроенных контейнеров

- `set container name <name> image`

Данная команда позволяет задать имя образа в реестре образов.

Если реестр образов не указан, в качестве него будет использован ресурс Docker.io до тех пор, пока реестр не будет явно указан с помощью команды ***set container registry <name>*** или пока реестр не будет задан как часть имени образа.

- `set container name <name> allow-host-networks`

Данная команда позволяет обеспечить доступ контейнера к сетевому стеку хоста. Сетевой стек контейнера не изолирован от сетевого стека хоста и использует IP адрес хоста.

> **Примечание**
>
> *Обратите внимание, что параметр allow-host-networks не может быть использован совместно с параметром network.*

- `set container name <name> network <networkname>`

Данная команда позволяет подключить к контейнеру пользовательскую сеть. При этом только одна сеть может быть указана в команде и она должна быть предварительно создана.

- `set container name <name> network <networkname> address <address>`

Данная команда позволяет опционально задать статический IPv4 или IPv6 адрес для контейнера. Задаваемый адрес должен быть из подсети, определенной префиксом указанной в команде сети.

> **Примечание**
>
> *Обратите внимание, что адрес первого хоста (X.X.X.1) зарезервирован механизмом и не может быть использован.*

- `set container name <name> description <text>`

Данная команда позволяет задать текстовое описание для созданного контейнера.

- `set container name <name> environment <key> value <value>`

Данная команда позволяет добавить для контейнера пользовательские переменные окружения. С помощью данной команды можно добавить несколько переменных окружения одновременно.

- `set container name <name> port <portname> source <portnumber>`
- `set container name <name> port <portname> destination <portnumber>`
- `set container name <name> port <portname> protocol <tcp | udp>`

Данные команды позволяют опубликовать во внешней сети определенный порт контейнера.

- `set container name <name> volume <volumename> source <path>`
- `set container name <name> volume <volumename> destination <path>`

Данные команды позволяют монтировать к контейнеру определенный раздел.

- `set container name <name> volume <volumename> mode <ro | rw>`

Данная команда позволяет установить режим доступа к разделу (ro - только чтение, rw - чтение и запись), по умолчанию - ro.

- `set container name <name> restart [no | on-failure | always]`

Данная команда позволяет задать условия перезапуска контейнера при выходе. Для выбора доступно одно из следующих условий:

- **no** - не перезапускать контейнер.
- **on-failure** - перезапускать контейнер, если получено ненулевое значение кода выхода (по умолчанию количество попыток перезапуска не ограничено).
- **always** - перезапускать контейнер вне зависимости от значения кода выхода (по умолчанию количество попыток перезапуска не ограничено).

- `set container name <name> memory <MB>`

Данная команда позволяет ограничить объем оперативной памяти (RAM) хоста, выделяемый для контейнера. Значение по умолчанию составляет 512 Мбайт. Используйте значение 0 Мбайт, чтобы полностью снять ограничение.

- `set container name <name> device <devicename> source <path>`

- `set container name <name> device <devicename> destination <path>`

Данные команды позволяют монтировать к контейнеру устройство, являющееся частью аппаратного обеспечения хоста.

- `set container name <name> cap-add <text>`

Данная команда позволяет назначить контейнеру определенные уровни доступа и полномочия. Для назначения доступны следующие варианты:

- **net-admin** - сетевые настройки (сетевые интерфейсы, межсетевой экран, таблицы маршрутизации).
- **net-bind-service** - назначение сокета привилегированным портам (номера портов меньше 1024).
- **net-raw** - создание «сырых» (raw) сокетов.
- **setpcap** - назначение уровней доступа (из привязанных или унаследованных наборов).
- **sys-admin** - администрирование (использование утилит quotactl, mount, sethostname, setdomainame).
- **sys-time** - установка системного времени.

- `set container name <name> disable`

Данная команда позволяет отключить контейнер.

- `set container network <networkname>`

Данная команда позволяет создать для контейнера сеть с заданным именем.

- `set container registry <name>`

Данная команда позволяет добавить реестр образов в список неквалифицированных реестров для поиска подходящих образов. По умолчанию для любого образа, имя которого не содержит явное указанием на реестр, операционная система Факел будет использовать ресурс Docker.io в качестве реестра образов.

Мониторинг и эксплуатация встроенных контейнеров

- `add container image <containername>`

Данная команда позволяет извлечь новый образ для контейнера из реестра.

- `show container`

Данная команда позволяет вывести на экран список всех активных контейнеров.

- `show container image`

Данная команда позволяет вывести на экран список всех образов, размещенных локально на диске устройства.

- `show container log <containername>`

Данная команда позволяет вывести на экран список событий, зарегистрированных в ходе работы выбранного контейнера.

- `show container network`

Данная команда позволяет вывести на экран список доступных для контейнера сетей.

- `restart container <containername>`

Данная команда позволяет перезапустить выбранный контейнер.

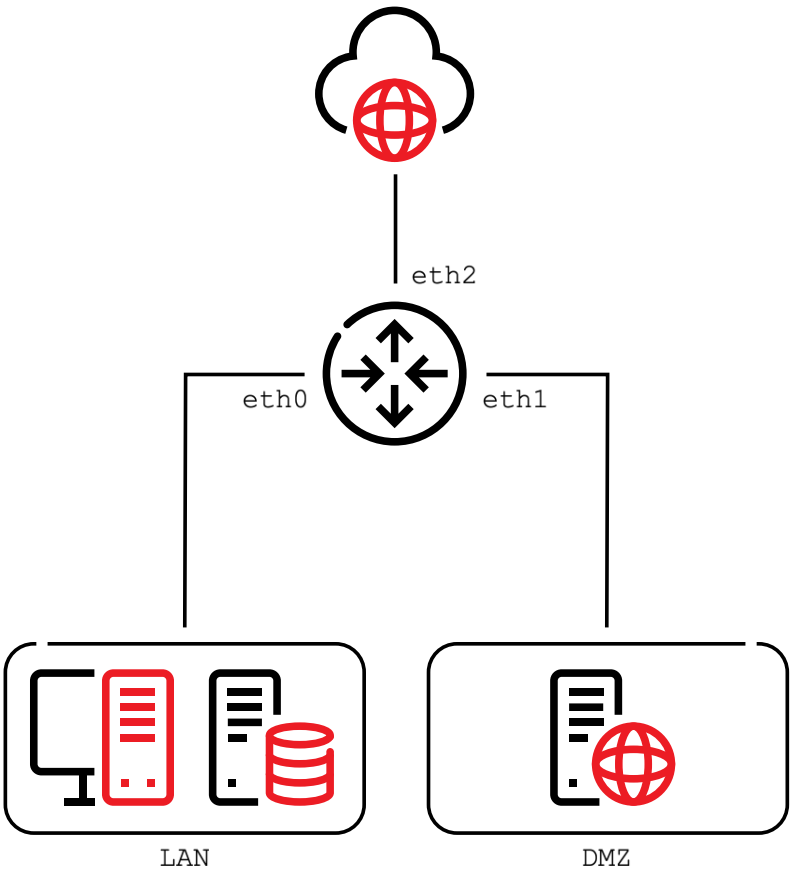
- `update container image <containername>`

Данная команда позволяет обновить образ выбранного контейнера.

Быстрый старт

Введение

Раздел «Быстрый старт» представляет собой краткое руководство по вводу в эксплуатацию устройства на базе ПО **Факел**. Краткое руководство описывает действия, которые необходимо выполнить с новым устройством, для настройки его базового функционала. Настройка устройства будет выполнена в соответствии со схемой, представленной ниже:



Подключение к устройству с установленным ПО

При первом запуске устройства на выбор можно использовать два типа подключения:

- Подключение к интерфейсу командной строки через консольный порт. Консольный порт устройства работает на скорости *115200*;
- Подключение к интерфейсу командной строки по SSH через сетевой порт управления. По умолчанию в качестве сетевого порта управления настроен порт *eth0*.

Для подключения к устройству укажите IP-адрес *192.168.150.1/24*, который настроен на порту *eth0*. После подключения к устройству открывается интерфейс командной строки в режиме администрирования.

Для настройки удаленного доступа к устройству по протоколу SSH выполните следующую команду:

```
set service ssh port 22
```

Использование метода аутентификации по ключам при подключении по SSH увеличит степень защиты удаленного подключения. Для настройки метода аутентификации по ключам выполните следующие команды:

```
set system login user fakel authentication public-keys
new_user@description type ssh-rsa
set system login user fakel authentication public-keys
new_user@description key contents_of_id_rsa.pub
```

Для получения подробной информации о возможности ограничения доступа по протоколу SSH обратитесь к разделу **Удаленный доступ**.

Командная строка

Режимы работы командной строки

Интерфейс командной строки операционной системы по умолчанию запускается в режиме администрирования. Символ «**$**» указывает на то, что интерфейс командной строки работает в режиме администрирования.

Для внесения изменений в настройки операционной системы переведите интерфейс командной строки в режим конфигурирования. Символ «**#**» указывает на то, что интерфейс командной строки работает в режиме конфигурирования. Для перехода в режим конфигурирования, введите команду ***configure***:

```
fakel@fakel:~$ configure
[edit]
fakel@fakel#
```

Для выхода из режима конфигурирования и перехода в режим администрирования введите команду ***exit***:

```
fakel@fakel# exit
exit
fakel@fakel:~$
```

Для получения подробной информации о работе с командной строкой обратитесь к разделу **Командная строка**.

Применение и сохранение конфигурации

После внесения изменений в настройках операционной системы, выполненных в режиме конфигурирования, необходимо применить эти изменения. Чтобы применить внесенные изменения выполните команду ***commit***.

Чтобы сохранить примененные изменения настроек операционной системы, выполните команду ***save***.

Чтобы отменить внесенные изменения настроек операционной системы, выполните команду ***exit discard***. Команда ***exit discard*** может отменить внесенные изменения только в том случае, если не была применена команда ***save***. После применения команды ***exit discard*** интерфейс командной строки переходит в режим администрирования.

Чтобы отменить сохраненные изменения настроек операционной системы, используйте команду ***rollback***. При выполнении команды ***commit*** в текущую версию конфигураций операционной системы, добавляются внесенные изменения и создается новая версия конфигураций операционной системы. Команда ***rollback*** выполняет откат до предыдущих версий конфигураций операционной системы. После выполнения команды ***rollback*** требуется перезагрузка устройства.

Для получения подробной информации о работе с конфигурацией устройства обратитесь к разделу **Режим Конфигурации.**

Учетные записи

При первом подключении к устройству используется учетная запись пользователя с именем *fakel* и паролем *fakel*. После авторизации под учетной записью пользователя *fakel*, смените пароль этой учетной записи.

Для смены пароля учетной записи fakel выполните следующую команду:

```
set system login user fakel authentication plaintext-password text_password
set system login user fakel authentication plaintext-password text_password
```

Чтобы создать нового пользователя *new_user* для работы с устройством выполните следующую команду:

```
set system login user new_user authentication plaintext-password
text_password
```

Для получения подробной информации об управлении учетными записями обратитесь к разделу **Управление пользователями.**

Сетевые интерфейсы и маршрутизация

Роль внешнего WAN интерфейса будет выполнять сетевой интерфейс *eth2*. Сетевые параметры этот интерфейс будет получать динамически по средствам протокола DHCP.

Для настройки IP-адреса на интерфейсе *eth2* выполните следующие команды:

```
set interfaces ethernet eth2 address dhcp
set ethernet eth0 description WAN
```

В роли внутреннего LAN интерфейса выступает сетевой интерфейс *eth0*, а интерфейс *eth1* в роли DMZ интерфейс. Сетевые параметры для этих интерфейсов будут настроены статически.

Для настройки IP-адреса на интерфейсах *eth0* и *eth1* выполните следующие команды:

```
set interfaces ethernet eth1 address 172.16.100.1/24
set interfaces ethernet eth1 description LAN
set interfaces ethernet eth1 address 10.150.0.1/24
set interfaces ethernet eth1 description DMZ
```

После настройки сетевых интерфейсов укажите статический маршрут в сторону шлюза по умолчанию.

Для настройки статического маршрута выполните следующую команду:

```
set protocols static route 0.0.0.0/0 next-hop 87.156.23.1
```

Для получения подробной информации о настройке сетевых интерфейсов обратитесь к разделу **Сетевые интерфейсы**.

Правила фильтрации (Firewall)

Для фильтрации входящего и исходящего трафика настройте правила межсетевого экрана. Фильтрация трафика выполняется в зависимости от направления, по которому этот трафик проходит через сетевой интерфейс:

- **local** - трафик, который приходит непосредственно на адрес сетевого интерфейса
- **in** - трафик, входящий на сетевой интерфейс
- **out** - трафик, исходящий с сетевого интерфейса

Для настройки фильтрации трафика создайте набор правил и примените его к сетевому интерфейсу.

Для настройки правил фильтрации трафика выполните следующие команды:

```
set firewall group network-group LAN-NET description 'LAN network'
set firewall group network-group LAN-NET network '172.16.100.0/24'
set firewall group network-group DMZ-NET description 'DMZ network'
set firewall group network-group DMZ-NET network '10.150.0.0/24'
set firewall receive-redirects 'disable'
set firewall send-redirects 'disable'
set firewall name WAN-IN default-action 'drop'
set firewall name WAN-LOCAL default-action 'drop'
set firewall name WAN-OUT default-action 'drop'
set firewall name WAN-OUT rule 10 action 'accept'
set firewall name WAN-OUT rule 10 source group network-group LAN-NET
set firewall name WAN-OUT rule 10 destination address 0.0.0.0/24
set firewall name WAN-OUT rule 10 description 'Allow LAB-NET to Internet'
set firewall name WAN-OUT rule 20 action 'accept'
set firewall name WAN-OUT rule 20 source group network-group DMZ-NET
set firewall name WAN-OUT rule 20 destination address 0.0.0.0/24
set firewall name WAN-OUT rule 20 destination port 'https, http'
set firewall name WAN-OUT rule 20 protocol 'tcp'
set firewall name WAN-OUT rule 20 description 'Allow DMZ-NET to Internet'
set interfaces ethernet eth2 firewall in name 'WAN-IN'
set interfaces ethernet eth2 firewall local name 'WAN-LOCAL'
```

```
set interfaces ethernet eth2 firewall out name 'WAN-OUT'
```

Для получения подробной информации о настройке правил фильтрации обратитесь к разделу **Межсетевой экран.**

Правила трансляции (NAT)

Чтобы предоставить доступ из внутренних сетей LAN и DMZ к ресурсам из внешней сети WAN, настройте правила трансляции адреса источника SNAT. Правила трансляции адресов будут заменять адрес источника из локальной сети *172.16.100.0/24* или сети DMZ *10.150.0.0/24* на адрес, настроенный на внешнем интерфейсе *eth2* по типу MASQUERADE.

Для настройки правил трансляции адресов выполните следующие команды:

```
set nat source rule 10 description "LAN to WAN translation"
set nat source rule 10 source address '172.16.100.0/24'
set nat source rule 10 outbound-interface 'eth2'
set nat source rule 10 translation address masquerade


set nat source rule 20 description "DMZ to WAN translation"
set nat source rule 20 source address '10.150.0.0/24'
set nat source rule 20 outbound-interface 'eth2'
set nat source rule 20 translation address masquerade
```

Чтобы предоставить доступ из внешней сети WAN к веб серверу из сети DMZ настройте правила трансляции адреса назначения DNAT. Правила трансляции адресов будут заменять адрес внешнего интерфейса *87.156.23.78* на адрес веб сервера *10.150.0.78* при обращении к адресу внешнего интерфейса из сети WAN по протоколу HTTPS.

Для настройки правил трансляции адресов выполните следующие команды:

```
set nat destination rule 10 description "Port Forward: 443 to
10.150.0.78"
set nat destination rule 10 destination address '87.156.23.78'
set nat destination rule 10 destination port '443'
set nat destination rule 10 inbound-interface 'eth2'
set nat destination rule 10 protocol 'tcp'
set nat destination rule 10 translation address '10.150.0.78'
```

```
set nat destination rule 10 translation options address-mapping
'persistent'
```

Для получения подробной информации о настройке правил трансляции обратитесь к разделу **Трансляция адресов.**

DNS, NTP и DHCP сервисы

Устройство может выступать в качестве шлюза по умолчанию, DNS, DHCP и NTP сервера локальной сети.

DNS сервер будет отвечать на запросы от хостов из подсети *172.16.100.0/24*. В качестве адреса DNS сервера будет настроен адрес *172.16.100.1*.

Для настройки функционала DNS сервера выполните следующие команды:

```
set service dns forwarding listen-address '172.16.100.1'
set service dns forwarding allow-from '172.16.100.0/24'
set service dns forwarding cache-size '0'
```

DHCP сервер будет выдавать адреса хостам из подсети *172.16.100.0/24* начиная с адреса *172.16.100.9* и заканчивая адресом *172.16.100.254*. В качестве DNS сервера и шлюза по умолчания для хостов из подсети *172.16.100.0/24* будет указан адрес *172.16.100.1*. Имя домена для хостов из подсети *172.16.100.0/24* будет *fakel.net*.

Для настройки функционала DHCP сервера выполните следующие команды:

```
set service dhcp-server shared-network-name LAN subnet
172.16.100.0/24
set service dhcp-server shared-network-name LAN subnet
172.16.100.0/24 default-router '172.16.100.1'
set service dhcp-server shared-network-name LAN subnet
172.16.100.0/24 name-server '172.16.100.1'
set service dhcp-server shared-network-name LAN subnet
172.16.100.0/24 domain-name 'fakel.net'
set service dhcp-server shared-network-name LAN subnet
172.16.100.0/24 lease '86400'
set service dhcp-server shared-network-name LAN subnet
172.16.100.0/24 range 0 start 172.16.100.9
set service dhcp-server shared-network-name LAN subnet
172.16.100.0/24 range 0 stop '172.16.100.254'
```

NTP сервер будет отвечать на запросы клиентов из подсетей *172.16.100.0/24* и *10.150.0.0/24*.

Для настройки NTP сервера выполните следующие команды:

```
set system ntp allow-clients address '172.16.100.0/24'
set system ntp allow-clients address '10.150.0.0/24'
set system ntp listen-address '172.16.100.1'
set system ntp listen-address '10.150.0.1'
```

Для настройки сервисов NTP и DNS на самом устройстве выполните следующие команды:

```
set system name-server 1.1.1.1
set system ntp server 172.16.100.10
```

После выполнения всех действий, описанных в данной главе, вы получите решение на базе ПО **Факел** в базовой конфигурации и обладающее достаточной степенью защищенности.

Контактная информация

Юридическая информация

Информация о юридическом лице компании:

- Название компании: ООО «ИТ Роут».
- Юр. адрес: 117105, г. Москва, ВН.ТЕР. Г. МУНИЦИПАЛЬНЫЙ ОКРУГ НАГОРНЫЙ, Ш ВАРШАВСКОЕ, Д. 26, СТР. 10, ПОМЕЩ. 1/3, КОМ. 5
- ОГРН: 1237700099141
- ИНН: 9726036525

Контактная информация службы технической поддержки

Связаться со специалистами службы технической поддержки можно одним из следующих способов:

- Сайт: https://fakel.io/support/
- Телефон: +7 499 390-98-21
- e-mail: support@fakel.io