АЛС-24623-10-xx

Configuration Manual

АЛСИТЕК

СОДЕРЖАНИЕ

ГЛАВА 1. БАЗОВЫЕ ОПЕРАЦИИ

1.1. Настройка доступа

Управляемые коммутаторы позволяют изменять свои настройки по COM, Telnet, SSH путем ввода текстовых команд через так называемый интерфейс командной строки (Command Line Interface, CLI). Также широко распространен протокол управления сетевыми устройствами SNMP.

Подключение к коммутатору по COM-порту

Шаг 1. Подключение COM-порта

Подключить консольный кабель к разъему на плате коммутатора, помеченному "control" или "service" (зависит от модели коммутатора).

Рисунок 1. Консольный кабель

Шаг 2. Настройка клиента

Запустить программу Putty со следующими параметрами:

- Connection type — Serial;
- Serial line — зависит от настроек компьютера (в примере использован COM1);
- Speed — 115200.

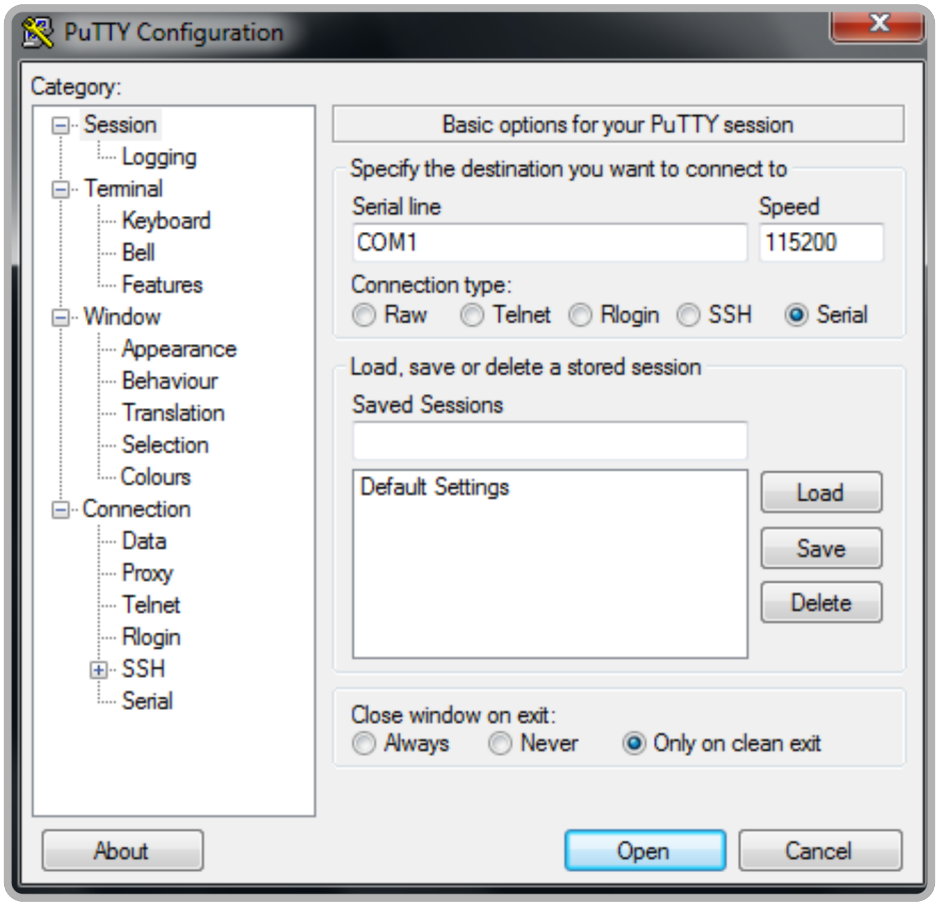

Рисунок 2. Настройка консольного подключения в Putty

Скорость порта в 115200 устанавливается по умолчанию на всех коммутаторах, однако в процессе эксплуатации ее возможно изменить в конфигурации коммутатора.

Для настройки скорости COM-порта используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #line console
(als_sw) (configure) (line) #serial baudrate <speed>
```

Поддерживаемые скорости: 9600, 19200, 38400, 57600, 115200 (по умолчанию). После выполнения данной команды значение скорости будет сохранено на flash коммутатора, но изменения вступят в силу только после перезагрузки коммутатора. При различной скорости подключения у клиента и коммутатора в консоли будут наблюдаться случайные символы, а управление будет невозможно.

Для просмотра текущих настроек COM-порта используется следующая команда:

```
(als_sw) #show serial

  Current baudrate.............................. 115200 Bd
  Next boot baudrate............................ 9600 Bd
  Inactivity timeout............................ 5 min
```

На примере текущая скорость COM установлена в 115200, но при следующей загрузке она будет изменена на 9600.

Шаг 3. Включение коммутатора и процесс загрузки

Включить питание коммутатора. В консоли Putty отобразится процесс загрузки, после чего появится приглашение:

```
User:
```

По умолчанию на коммутаторе есть два пользователя: **admin** (полный доступ) и **guest** (только чтение), с пустым паролем. Для входа и первичной настройки используйте пользователя **admin** с пустым паролем:

```
User:admin
Password:

(als_sw) #
```

Настройка сетевых параметров коммутатора

Помимо COM-порта для управления можно использовать протоколы Telnet, SSH и SNMP. Для подключения к коммутатору по этим протоколам необходимо знать сетевой адрес коммутатора. При поставке коммутатор имеет чистую конфигурацию и следующие параметры:

- уникальный MAC-адрес вида 00:13:AA:XX:XX:XX, назначенный производителем;

- VLAN управления, равный VLAN 1;
- IPv4-адрес 172.17.1.1, маска подсети 255.255.0.0;
- IPv6-адрес, сгенерированный на основании MAC-адреса;
- двух пользователей по умолчанию: **admin** (полный доступ) и **guest** (только чтение), с пустым паролем;
- доступ по протоколу Telnet включен по умолчанию;
- доступ по протоколу SSH (v1/v2) выключен по умолчанию;
- доступ по протоколу SNMP (v1/v2c) включен по умолчанию и имеет два SNMP-community: **private** (полный доступ) и **public** (только чтение);
- доступ по протоколу SNMP (v3) включен по умолчанию и имеет двух пользователей по умолчанию: **admin** (полный доступ) и **guest** (только чтение). Режим подключения SNMPv3: без хеширования пароля и без шифрования соединения (noAuthNoPriv в терминах пакета net-snmp).

Шаг 1. Настройка IPv4-адреса

Для настройки IPv4-адреса коммутатора используется команда:

```
(als_sw) #network parms <ip address> <subnet> [<gateway>]
```

Параметры:

- <ip address> — IP-адрес, например 172.17.1.1;
- <subnet> — маска подсети, например 255.255.0.0;
- [<gateway>] — шлюз по умолчанию (необязательный параметр), например 172.17.1.254.

Пример подключения напрямую изображен на схеме ниже:

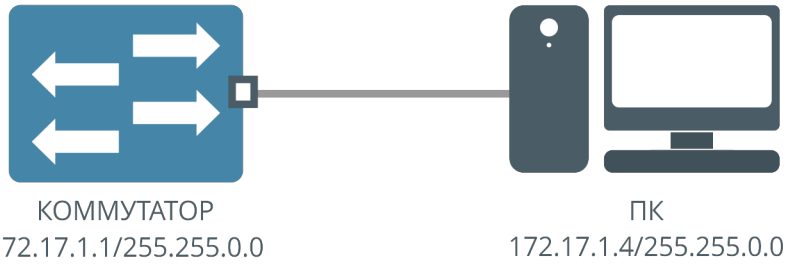

Рисунок 3. Прямое подключение ПК к коммутатору

На схеме указан адрес коммутатора по умолчанию. При таком соединении и настройке доступность коммутатора с ПК можно проверить командой ping:

```
$ ping 172.17.1.1
PING 172.17.1.1 (172.17.1.1) 56(84) bytes of data.
64 bytes from 172.17.1.1: icmp_seq=1 ttl=64 time=4.95 ms
64 bytes from 172.17.1.1: icmp_seq=2 ttl=64 time=2.06 ms
64 bytes from 172.17.1.1: icmp_seq=3 ttl=64 time=2.35 ms
^C
--- 172.17.1.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 2.067/3.124/4.954/1.299 ms
```

Пример подключения ПК к коммутатору через шлюз:

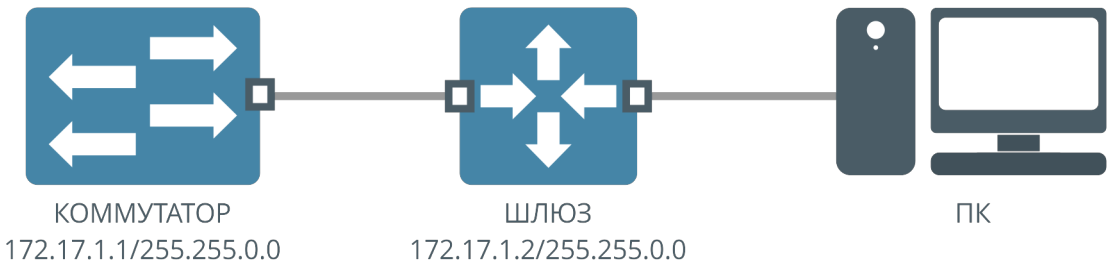

Рисунок 4. Подключение ПК к коммутатору через шлюз

Для подключения ПК к коммутатору через шлюз необходимо настроить адрес шлюза на коммутаторе, а также настроить маршрутизацию в сеть коммутатора со стороны ПК (на шлюзе или на ПК, в зависимости от настроек сети).

Конфигурация коммутатора для примера выше:

```
(als_sw) #network parms 172.17.1.1 255.255.0.0 172.17.1.2
```

Также есть возможность получить адрес по протоколу DHCP, для этого нужно выполнить следующую команду:

```
(als_sw) #network protocol dhcp
```

При данной настройке коммутатор будет получать адрес по протоколу DHCP в управляющем VLAN. При этом статический адрес, настроенный ранее, не будет работать.

При настройке получения адреса по DHCP также может быть необходимо указать DHCP Option 60 для клиента. Для включения отправки опции и задания ее значения используются следующие команды:

```
(als_sw) #configure
(als_sw) (configure) #dhcp client vendor-id-option
(als_sw) (configure) #dhcp client vendor-id-option-string "client47"
```

После включения этой опции в клиентских пакетах DHCP будет присутствовать опция DHCP 60, значение по умолчанию для этой опции содержит идентификатор производителя и модели коммутатора. При задании значения опции в конфигурации она будет отправлять точно так, как установлена настройкой. По значению данной опции DHCP-сервер может выполнить дополнительные проверки и выдать адрес из определенного пула.

Для отключения получения адреса по DHCP и возвращению к статическим параметрам сети используется следующая команда:

```
(als_sw) #network protocol none
```

Шаг 2. Настройка VLAN управления

По умолчанию коммутатор управляется во VLAN 1. Чтобы задать другой VLAN управления, используется команда:

```
(als_sw) #network mgmt_vlan <vlan>
```

Параметры:

- <vlan> — номер VLAN управления в диапазоне от 1 до 4095.

Шаг 3. Настройка IPv6-адреса

Коммутатору можно назначить статический IPv6-адрес с помощью команды:

```
(als_sw) #network ipv6 address <ipv6>/<prefix>
```

Параметры:

- <ipv6>/<prefix> — IPv6 адрес и префикс, например 2001::100/64.

Кроме того, возможна автоматическая настройка адреса по протоколу SLAAC, согласно RFC 2462. По умолчанию автоматическое назначение адреса отключено, для его включения используется команда:

```
(als_sw) #network ipv6 address autoconfig
```

Также есть возможность получить IPv6 адрес по DHCPv6, для этого нужно выбрать соответствующую опцию в настройке адреса:

```
(als_sw) #network ipv6 address dhcp
```

Включение получения адреса по DHCPv6 подразумевает включение SLAAC, по которому клиент получит рекомендацию продолжить получение дополнительных настроек по DHCPv6.

Шаг 4. Настройка IPv6-шлюза

Настройка IPv6-шлюза не является обязательной. Она осуществляется следующей командой:

```
(als_sw) #network ipv6 gateway <ipv6>
```

Параметры:

- <ipv6> — IPv6-адрес шлюза.

Шаг 5. Просмотр сетевых параметров

Просмотр сетевых параметров осуществляется следующей командой:

```
(als_sw) #show network

Primary Interface:
  IPv4 Address..................................... 172.17.1.10
  Subnet Mask...................................... 255.255.0.0
  DHCPv4 Mode...................................... Disabled
  IPv6 Autoconfig Mode............................. Disabled
  DHCPv6 Mode...................................... Disabled
  IPv6 Link-Local Address.......................... fe80::213:aaff:fe1e:5/64
  IPv6 Static Address.............................. 2001:470:1f15:2db:20e:dff:
                                                    fec4:20b5/64
  MAC Address...................................... 00:13:aa:1e:00:05
  Management VLAN ID............................... 10

Additional Interface:
  IPv4 Address..................................... 0.0.0.0
  Subnet Mask...................................... 0.0.0.0
  DHCPv4 Mode...................................... Disabled
  IPv6 Autoconfig Mode............................. Disabled
  DHCPv6 Mode...................................... Disabled
  MAC Address...................................... 00:13:aa:1e:00:05
  Management VLAN ID............................... 4094

Default gateway:
  IPv4 Default Gateway Address..................... 172.17.1.100
  IPv6 Default Gateway Address..................... None
```

Поля:

- IP Address — IPv4-адрес коммутатора (значение по умолчанию 172.17.1.1);
- Subnet Mask — сетевая маска коммутатора (значение по умолчанию 255.255.0.0);
- DHCPv4 Mode — состояние клиента DHCPv4 (по умолчанию отключен);
- IPv6 AutoConfig Mode — режим автоконфигурирования IPv6 (SLAAC, по умолчанию отключен);
- DHCPv6 Mode — состояние клиента DHCPv6 (по умолчанию отключен);
- IPv6 Link-Local Address — IPv6-адрес коммутатора в link-local scope. Назначается автоматически на основе MAC-адреса;
- IPv6 Static Address — назначенный в конфигурации IPv6-адрес;
- IPv6 Dynamic Address — полученный по SLAAC или DHCPv6 динамический адрес. Отсутствует, если не включен SLAAC и/или DHCPv6;
- MAC Address — MAC-адрес коммутатора. Назначается производителем;
- Management VLAN ID — VLAN управления коммутатором, значение по умолчанию VLAN 1, то есть доступ к коммутатору не требует установки тега на управляющем трафике;
- IPv4 Default Gateway Address — адрес IPv4-шлюза по умолчанию (по умолчанию отсутствует);
- IPv6 Default Gateway Address — адрес IPv6-шлюза по умолчанию (по умолчанию отсутствует).

Настройка Out-of-band управления

Некоторые модели коммутаторов имеют на передней панели порт Ethernet 10/100 Mbit/s для Out-of-band управления. Интерфейс не связан с коммутатором и не может быть использован для передачи трафика вместе с остальными интерфейсами коммутатора.

Шаг 1. Включение интерфейса

Для включения интерфейса необходимо выполнить команду:

```
(als_sw) #serviceport protocol none
```

Шаг 2. Настройка IPv4-адреса

Для настройки IPv4-адреса на интерфейсе используется команда:

```
(als_sw) #serviceport ip <ip address> <subnet> [<gateway>]
```

Параметры:

- <ip address> — IP-адрес, например 172.18.1.1;
- <subnet> — маска подсети, например 255.255.0.0;
- [<gateway>] — шлюз по умолчанию (необязательный параметр), например 172.18.1.254.

IP-адрес интерфейса для Out-of-band управления не может быть из подсети, используемой для основного или дополнительного сетевого интерфейса коммутатора.

Шаг 3. Просмотр сетевых параметров

Просмотр сетевых параметров осуществляется следующей командой:

```
(als_sw) #show serviceport

  Interface Status.............................. Up
  IP Address.................................... 172.18.1.1
  Subnet Mask................................... 255.255.0.0
  Default Gateway............................... 0.0.0.0
  Configured IPv4 Protocol...................... None
  Burned In MAC Address......................... 00:13:aa:1b:09:6e
```

Поля:

- Interface Status — текущее состояние интерфейса;
- IP Address — IPv4-адрес коммутатора;
- Subnet Mask — сетевая маска коммутатора;
- Default Gateway — шлюз (значение по умолчанию 0.0.0.0);
- Configured IPv4 Protocol — используемый протокол настройки IP-адреса;
- MAC Address — MAC-адрес коммутатора.

Подключение по протоколу Telnet

Шаг 1. Настройка сети

Необходимо настроить сеть на коммутаторе описанным выше способом. Сетевой адрес коммутатора должен быть доступен с ПК.

Шаг 2. Использование клиента

Запустить программу Putty со следующими параметрами:

- Connection type — Telnet;
- Host name (or IP address) — IP-адрес коммутатора;
- Port — 23.

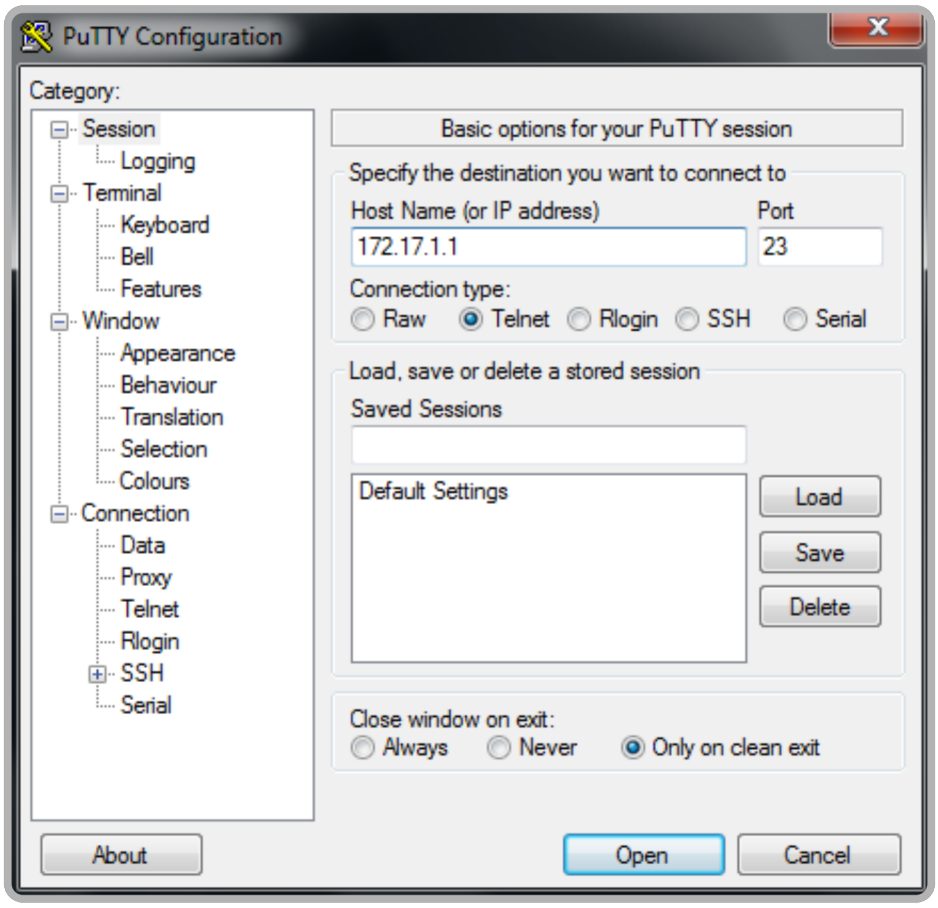

Рисунок 5. Настройка Telnet-соединения в Putty

Подключение по протоколу SSH

Шаг 1. Настройка сети

Необходимо настроить сеть на коммутаторе описанным выше способом. Сетевой адрес коммутатора должен быть доступен с ПК.

Шаг 2. Генерация ключей SSH (при первом использовании)

Перед подключением по протоколу SSH необходимо настроить коммутатор, разрешив подключение. Кроме того, необходимо создать ключи шифрования на коммутаторе. Создание ключей требуется только один раз, впоследствии ключи сохраняются на flash-памяти коммутатора и не меняются. Есть возможность заменить ключи, повторно вызвав команду генерации.

Команда генерации ключа RSA (используется для подключения по протоколу SSHv1):

```
(als_sw) #configure

(als_sw) (configure) #crypto key generate rsa

INFO: Single-user command was called in current or another control session, man
agement access is blocked for the duration of the single-user command

RSA keys generation. It may take several minutes to complete, please wait..

INFO: Single-user command was completed in current or another control session,
management access is restored

RSA keys have been generated successfully
```

Команда генерации ключа DSA (используется для подключения по протоколу SSHv2):

```
(als_sw) (configure) #crypto key generate dsa

INFO: Single-user command was called in current or another control session, man
agement access is blocked for the duration of the single-user command

DSA keys generation. It may take several minutes to complete, please wait..

INFO: Single-user command was completed in current or another control session,
management access is restored

DSA keys have been generated successfully
```

Команда генерации ключа RSA1 (используется для подключения по протоколу SSHv2 в новых версиях библиотеки OpenSSL и подобных):

```
(als_sw) (configure) #crypto key generate rsa1

INFO: Single-user command was called in current or another control session, man
agement access is blocked for the duration of the single-user command

RSA keys generation. It may take several minutes to complete, please wait..

INFO: Single-user command was completed in current or another control session,
management access is restored

RSA keys have been generated successfully
```

Для нормальной работы доступа по SSH нужно сгенерировать все ключи. Если ключи уже были сгенерированы ранее, коммутатор потребует подтверждение на замену ключей. При первой генерации подтверждения не потребуется. Генерация ключей занимает несколько минут.

Шаг 3. Включение доступа по SSH

Команда включения доступа по SSH:

```
(als_sw) #ip ssh server enable
```

После этого доступ к коммутатору можно производить по протоколу SSH обеих версий.

Шаг 4. Использование клиента

Для доступа к коммутатору по протоколу SSH можно использовать любой клиент, в качестве примера для Windows можно привести Putty, в качестве примера для Linux — OpenSSH или любой другой. Поддерживаются версии протокола SSHv1 и SSHv2.

Подключение по протоколу HTTP

Шаг 1. Настройка сети

Необходимо настроить сеть на коммутаторе описанным выше способом. Сетевой адрес коммутатора должен быть доступен с ПК.

Шаг 2. Включение доступа по HTTP

По умолчанию доступ по протоколу HTTP на коммутатор отключен. Для его включения используется команда:

```
(als_sw) #ip web server enable
```

После выполнения этой команды можно воспользоваться любым браузером и зайти на коммутатор по его адресу, набрав адрес в адресной строке браузера.

Внешний вид окна браузера при входе на коммутатор изображен на рисунке ниже:

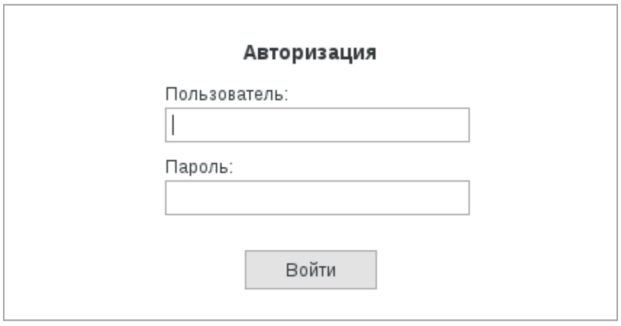

Рисунок 6. Форма входа

Шаг 3. Вход по HTTP

В поле "Пользователь" необходимо ввести имя пользователя на коммутаторе, а в поле "Пароль" — пароль этого пользователя. Внешний вид браузера может быть следующим:

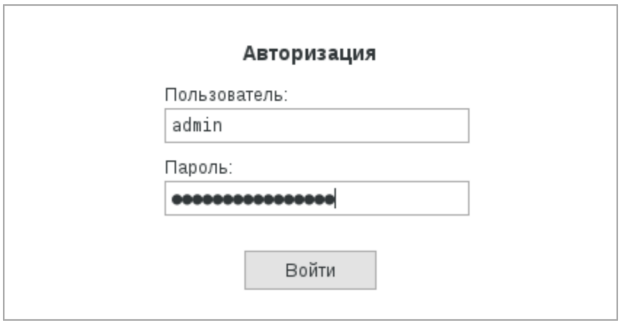

Рисунок 7. Вход пользователя

После успешного входа будет показана вкладка "Обзор". Содержание вкладки сильно зависит от модели коммутатора и его конфигурации. Внешний вид вкладки представлен на рисунке ниже:

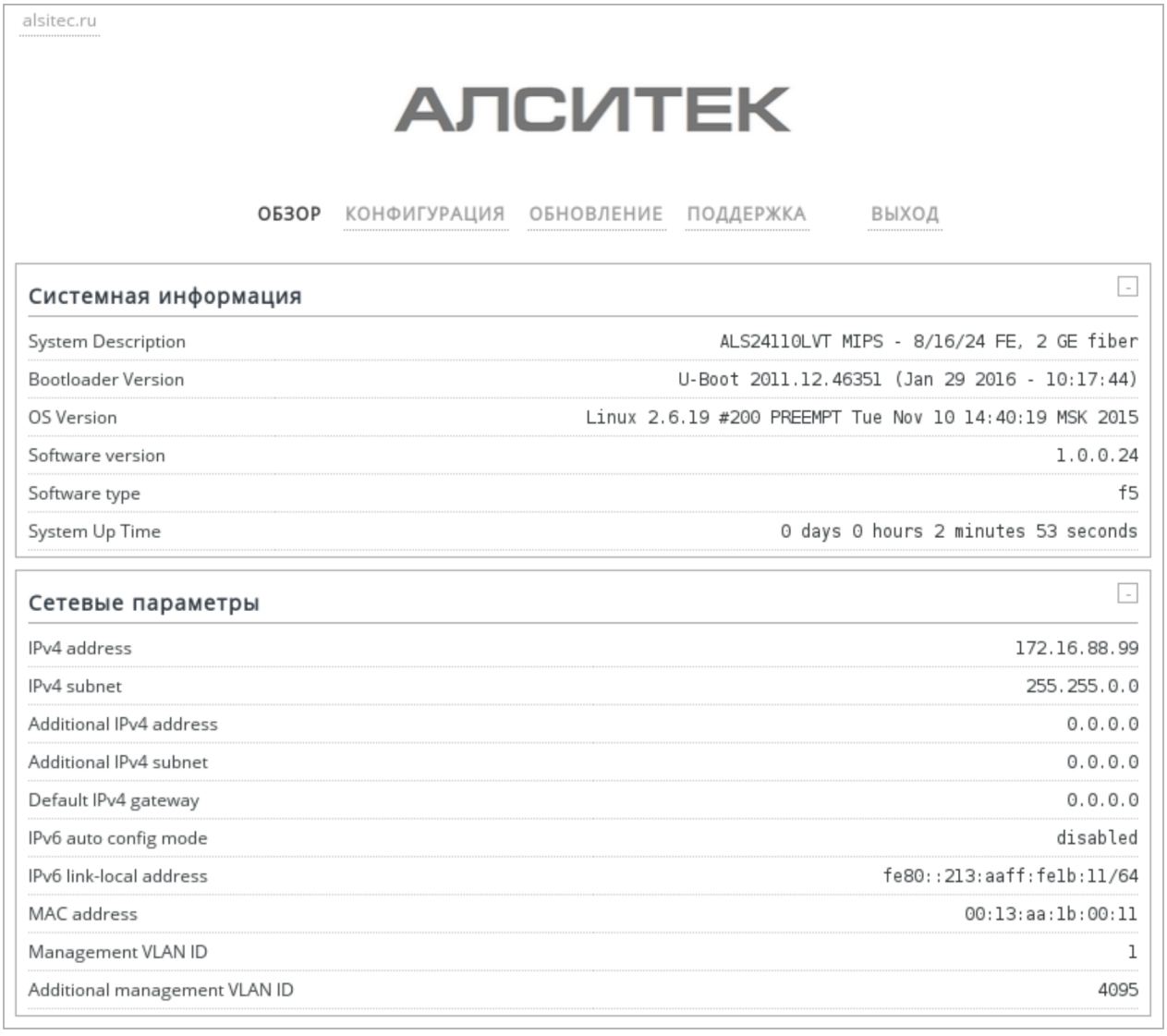

Рисунок 8. Вкладка "Обзор"

Используя пункты верхнего меню, можно переключаться между вкладками. Вкладка "Конфигурация" содержит две ссылки для скачивания в виде текстового файла стартовой конфигурации "startup-config.txt" и текущей конфигурации "running-config.txt". Внешний вид вкладки представлен на рисунке ниже:

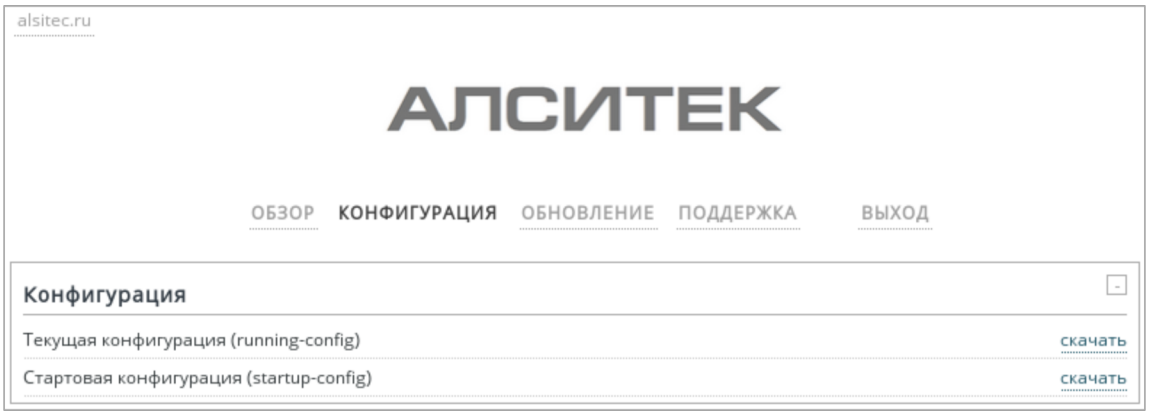

Рисунок 9. Вкладка "Конфигурация"

Вкладка "Обновление" позволяет обновить ПО коммутатора. Внешний вид вкладки представлен на рисунке ниже:

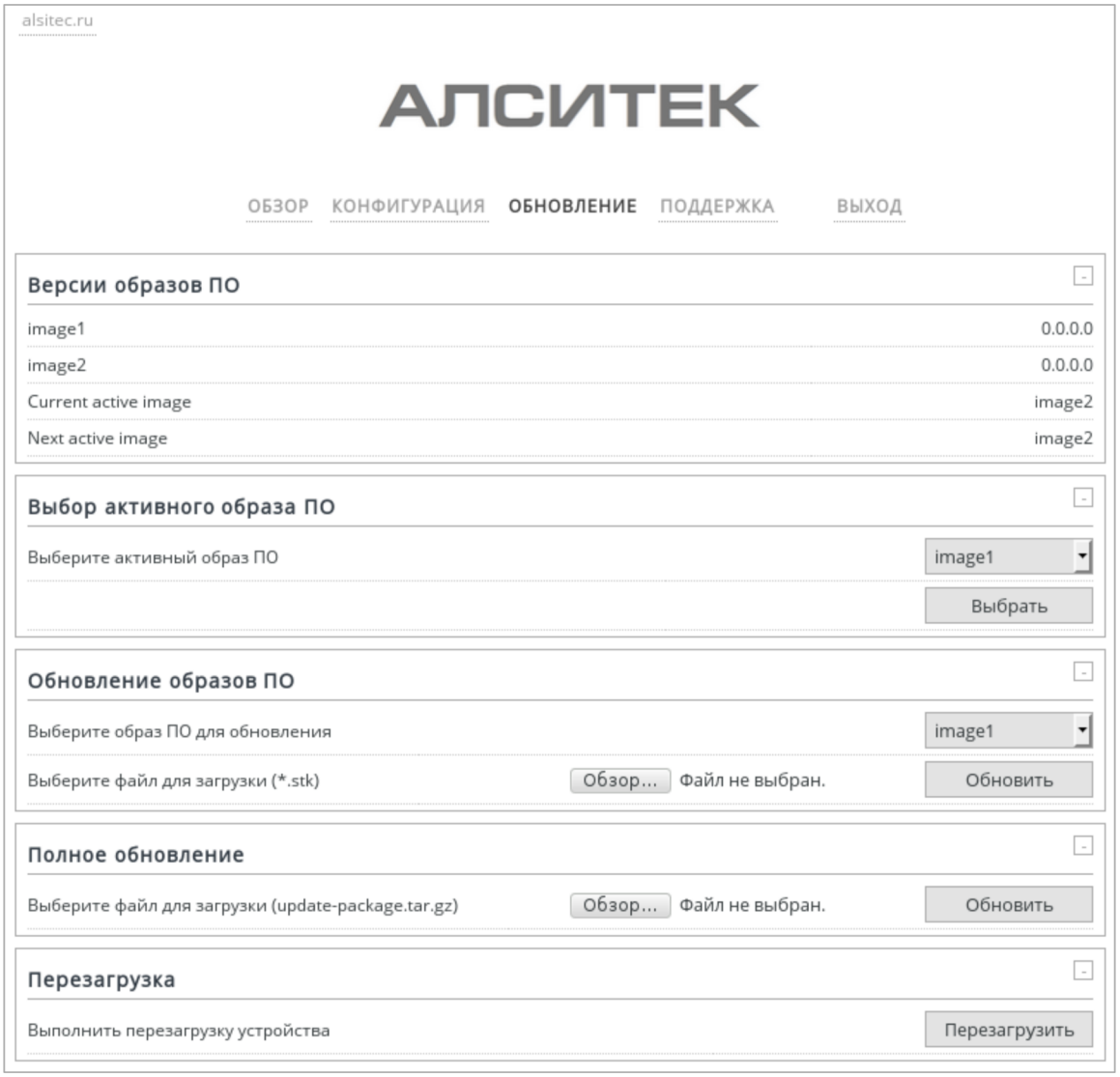

Рисунок 10. Вкладка "Обновление"

Пункт меню "Поддержка" откроет в новом окне браузера сайт технической поддержки, который поможет решить вопросы, возникающие в ходе настройки и эксплуатации оборудования и ПО.

Пункт меню "Выход" закроет текущую сессию пользователя, после чего пользователь будет перенаправлен на страницу входа.

Шаг 4. Настройка таймаута сессии HTTP (опционально)

По умолчанию таймаут сессии HTTP составляет 5 минут. Если пользователь вошел на коммутатор, и с момента его последней активности прошло более 5 минут, то сессия считается завершенной по таймауту, и при попытке сделать что-либо этот пользователь будет перенаправлен на страницу входа. В некоторых случаях этого значения может быть недостаточно.

Для изменения таймаута сессии HTTP используется команда:

```
(als_sw) #ip web server session timeout <minutes>
```

Параметры:

- <minutes> — значение таймаута сессии HTTP в минутах. Допустимые значения от 1 до 160.

1.2. Дополнительные настройки

Настройка дополнительного интерфейса

Для управления коммутатором есть возможность настроить второй (дополнительный) интерфейс.

Для настройки дополнительного IPv4-адреса используется команда:

```
(als_sw) #network addparms <ip address> <subnet> [<gateway>]
```

Параметры:

- <ip address> — IP-адрес, например 192.168.1.1;
- <subnet> — маска подсети, например 255.255.255.0;
- [<gateway>] — шлюз по умолчанию (необязательный параметр), например 192.168.1.254.

Для настройки VLAN управления для дополнительного интерфейса используется команда:

```
(als_sw) #network addmgmt_vlan <vlan>
```

Параметры:

- <vlan> — номер VLAN управления в диапазоне от 1 до 4095.

По умолчанию дополнительный интерфейс выключен и не имеет адреса. При его настройке, а также при настройке адреса основного интерфейса, существуют следующие ограничения:

- IPv4-адреса основного и дополнительного интерфейсов должны быть в разных подсетях;
- VLAN управления для основного и дополнительного интерфейса не должны совпадать;
- Адрес шлюза может быть назначен только один — либо для основного, либо для дополнительного интерфейса. Назначить второй шлюз по умолчанию нельзя.

Дополнительный интерфейс начинает работать сразу же после назначения адреса. Доступ к коммутатору по дополнительному интерфейсу ничем не отличается от доступа по основному интерфейсу. Службы Telnet, SSH, SNMP и прочие работают по нему точно так же.

Пример настройки доступа к коммутатору по двум интерфейсам изображен ниже:

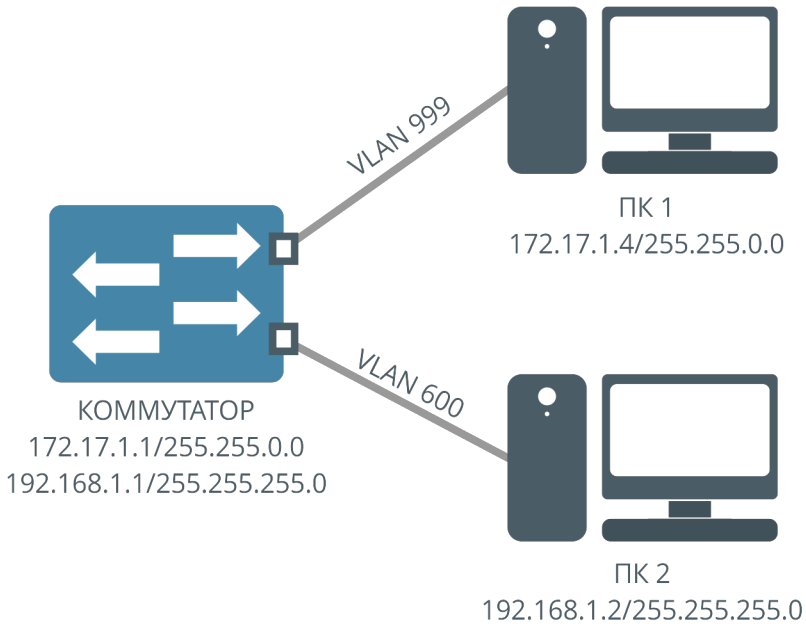

Рисунок 11. Пример настройки доступа к коммутатору по двум интерфейсам

Конфигурация для примера выше:

```
(als_sw) #network parms 172.17.1.1 255.255.0.0
(als_sw) #network mgmt_vlan 999
(als_sw) #network addparms 192.168.1.1 255.255.255.0
(als_sw) #network addmgmt_vlan 600
```

Для того, чтобы коммутатор в примере был доступен с ПК 1 и ПК 2, необходимо также настроить VLAN на интерфейсах подключения (например назначить Port-based VLAN 999 и 600 на соответствующих интерфейсах коммутатора).

Настройка меток 802.1p и DSCP для интерфейсов управления

В некоторых случаях необходимо установить определенное значение метки 802.1p или DSCP для трафика управления коммутатором.

Настройка метки 802.1p для интерфейсов управления

Для установки значения метки 802.1p для исходящего с коммутатора трафика управления на основном интерфейсе управления используется команда:

```
(als_sw) #network parms cos <cos>
```

Параметры:

- <cos> — значение приоритета 802.1, может принимать значения от 0 до 7.

Для установки значения метки 802.1p на дополнительном интерфейсе управления используется аналогичная команда:

```
(als_sw) #network addparms cos <cos>
```

Параметры:

- <cos> — значение приоритета 802.1, может принимать значения от 0 до 7.

Настройка метки DSCP для интерфейсов управления

Для установки метки DSCP для основного интерфейса управления используется команда:

```
(als_sw) #network parms dscp <dscp>
```

Параметры:

- <dscp> — значение метки DSCP, может принимать значения от 0 до 63. Также разрешено использовать строковые константы, перечисленные в таблице ниже.

Для установки метки DSCP для дополнительного интерфейса управления используется команда:

```
(als_sw) #network addparms dscp <dscp>
```

Параметры:

- <dscp> — значение метки DSCP, может принимать значения от 0 до 63. Также разрешено использовать строковые константы, перечисленные в таблице ниже.

Для установки значения DSCP помимо числового значения можно использовать константы:

Символьная константа	Числовое значение
af11	10
af12	12
af13	14
af21	18
af22	20

af23	22
af31	26
af32	28
af33	30
af41	34
af42	36
af43	38
cs0	0
cs1	8
cs2	16
cs3	24
cs4	32
cs5	40
cs6	48
cs7	56

Настройка маршрутизации для интерфейсов управления

При настройке IPv4-адреса может возникнуть необходимость в задании специальных правил маршрутизации для определенных хостов или подсетей.

В этом случае используется команда создания правила статической маршрутизации:

```
(als_sw) #network route <ip> <netmask> gw <gateway> [primary|additional]
```

Параметры:

- <ip — адрес подсети или хоста назначения;
- <netmask> — сетевая маска подсети назначения. В случае хоста укажите маску 255.255.255.255;
- <gateway> — адрес шлюза, через который необходимо направлять IP-трафик;
- [primary|additional] — необязательный параметр, с помощью которого можно явно указать целевой интерфейс для правила маршрутизации.

При создании правил маршрутизации важно помнить, что адрес <gateway> должен быть в одной подсети хотя бы с одним из адресов коммутатора, иначе коммутатор не сможет получить к нему доступ, и, соответственно, не сможет переслать IP-трафик.

Удалить правило маршрутизации можно командой:

```
(als_sw) #no network route <network> <netmask>
```

При удалении достаточно указать только адрес подсети или хоста назначения и маску.

Пример настройки маршрутизации с коммутатора в две различные сети (сеть A и сеть B на рисунке) через два различных шлюза:

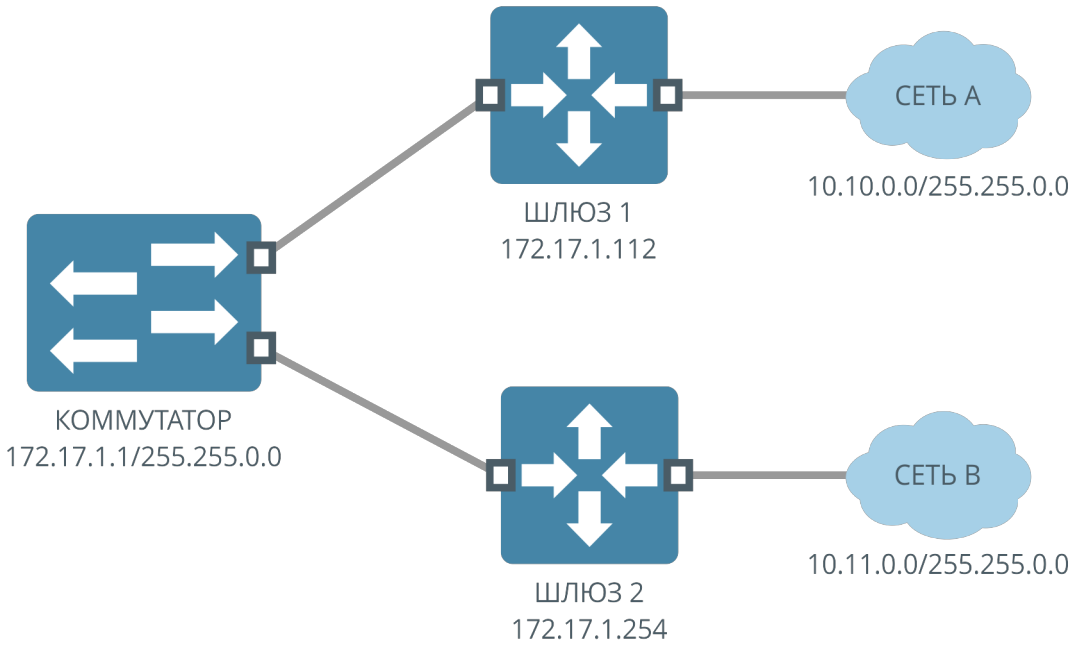

Рисунок 12. Пример настройки маршрутизации с коммутатора в две различные сети

Конфигурация для примера выше:

```
(als_sw) #network parms 172.17.1.1 255.255.0.0
(als_sw) #network route 10.10.0.0 255.255.0.0 gw 172.17.1.112
(als_sw) #network route 10.11.0.0 255.255.0.0 gw 172.17.1.254
```

Просмотреть маршруты можно следующей командой:

```
(als_sw) #show network routes

  Destination/Prefix                Gateway                           Interface

  --------------------------------  --------------------------------  ---------
-
  172.17.0.0/16                     0.0.0.0                           primary
```

В выводе перечислены все фактические маршруты, в том числе и маршруты, полученные через DHCP, а не только те, что были добавлены в конфигурации.

Также можно настроить статические маршруты и для IPv6 с помощью следующей команды:

```
(als_sw) #network route ipv6 <ipv6-dest>/<prefix> gw <ipv6-gateway> [primary|ad
ditional]
```

Параметры:

- <ipv6-dest> — IPv6-адрес назначения;
- <prefix> — префикс назначения
- <ipv6-gateway> — IPv6-адрес шлюза для этого маршрута;
- [primary|additional] — маршрут создается для основного или дополнительного интерфейса. По умолчанию для основного.

Просмотреть маршруты IPv6 можно следующей командой:

```
(als_sw) #show network routes ipv6

  Destination/Prefix                Gateway                           Interface

  --------------------------------  --------------------------------  ---------
-
  fe80::/64                         ::                                primary
  ff00::/8                          ::                                primary
```

В выводе перечислены все маршруты, в том числе полученные DHCPv6, а не только те, что были настроены в конфигурации. Кроме того, механизм ICMPv6 Neighbor Discovery при общении в сегменте по Link Local адресам при получении анонса адреса соседа создает временный маршрут к этому соседу, который также будет отображаться в этой таблице.

Настройка даты и времени

SNTP (англ. Simple Network Time Protocol) — упрощенный протокол синхронизации времени. Является упрощенной реализацией протокола NTP. Используется во встраиваемых системах и устройствах, не требующих высокой точности установки времени.

В протоколе SNTP используется одинаковый с протоколом NTP формат представления времени.

Шаг 1. Настройка серверов

Добавить необходимые SNTP-серверы командой:

```
(als_sw) #configure
(als_sw) (configure) #sntp server <ip> [<prio> [<version> [<port>]]]
(als_sw) #exit
```

Параметры:

- <ip> — IP-адрес SNTP-сервера;
- [<prio>] — приоритет синхронизации с сервером в диапазоне от 1 — высший до 3 — низший (необязательный параметр);
- [<version>] — версия SNTP-сервера (необязательный параметр, значение по умолчанию 4);
- [<port>] — порт сервера в диапазоне от 1 до 65535 (необязательный параметр, значение по умолчанию 123).

Максимальное количество поддерживаемых серверов — 3. Если серверов несколько, они будут использоваться согласно приоритету, то есть сначала идет обращение к серверу с наибольшим приоритетом, если он не отвечает в течение 2 секунд — то к следующему по приоритету и так далее.

Для удаления сервера применяется команда:

```
(als_sw) #configure
(als_sw) (configure) #no sntp server <ip>
(als_sw) #exit
```

Параметры:

- <ip> — IP-адрес SNTP-сервера, указанный при создании.

Шаг 2. Включение SNTP-клиента

Включить SNTP-клиент командой:

```
(als_sw) #configure
(als_sw) (configure) #sntp client mode unicast
(als_sw) #exit
```

После выполнения данной команды SNTP-клиент включается и начинает синхронизировать время с настроенными SNTP-серверами.

Шаг 3. Настройка временной зоны

Настроить временную зону относительно UTC можно командой:

```
(als_sw) #configure
(als_sw) (configure) #clock timezone <hours> [minutes <minutes> [zone <string>]
]
(als_sw) #exit
```

Параметры:

- <hours> — часы в диапазоне от -12 до 12;
- <minutes> — минуты в диапазоне от 0 до 59, для промежуточных зон;
- <string> — аббревиатура временной зоны, например "MSK".

Для просмотра времени на коммутаторе используется следующая команда:

```
(als_sw) #show clock

16:45:40 MSK(UTC+3:00) May 1 2015
```

Шаг 4. Настройка автоматического перевода на летнее время

Перевод на летнее время и обратно не предусмотрен в стандартном представлении времени UTC. Для автоматического перехода на летнее время в определенную дату и время необходимо включить перевод на коммутаторе. По умолчанию переход на летнее время выключен.

Для включения перевода на летнее время и назначения интервала дат летнего времени используется команда:

```
(als_sw) #configure
(als_sw) (configure) #clock summer-time recurring <first|last> <day> <month> <h
h>:<mm> <first|last> <day> <month> <hh>:<mm>
(als_sw) #exit
```

Параметры:

- <first|last> — первая или последняя неделя в указанном месяце;
- <day> — день недели в виде сокращенного названия, а именно: Mon, Tue, Wed, Thu, Fri, Sat, Sun. Регистр букв не имеет значения, названия SUN и sun равнозначны;
- <month> — месяц, в формате сокращенного названия, а именно: Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec. Регистр букв не имеет значения, названия MAR и mar равнозначны;
- <hh>:<mm> — час и минуты перехода на летнее время и обратно, <hh> должен быть числом от 0 до 23, <mm> должен быть числом от 0 до 59.

Эта команда принимает две отметки времени в году, когда нужно перейти на летнее время (первый блок параметров) и когда нужно вернуться к нормальному времени (второй блок параметров).

Пример включения перевода на летнее время со стандартными для Российской Федерации параметрами:

```
(als_sw) #configure
(als_sw) (configure) #clock summer-time recurring last Sun Mar 2:00 last Sun Oc
t 2:00
(als_sw) #exit
```

Введенная команда включает переход на летнее время в последнее воскресенье марта в 2 часа ночи (*last Sun Mar 2:00*), и задает окончание летнего времени в последнее воскресенье октября в 2 часа ночи (*last Sun Oct 2:00*).

Перевод на летнее время работает совместно с синхронизацией времени и временной зоной, между ними нет конфликта, так как синхронизация времени получает время в UTC, настройка временной зоны сдвигает время UTC на определенную величину в часах и минутах, а летнее время добавляет 1 час в определенный период.

Отключение настроенного ранее перехода на летнее время производится командой:

```
(als_sw) #configure
(als_sw) (configure) #no clock summer-time
(als_sw) #exit
```

ГЛАВА 2. КОНЦЕПЦИИ КОНФИГУРИРОВАНИЯ

2.1. Интерфейс командной строки

Интерфейс командной строки (англ. Command Line Interface, CLI) — разновидность текстового интерфейса между человеком и оборудованием, в котором инструкции оборудованию даются путем ввода с клавиатуры текстовых команд.

Синтаксис команд

Команда — это одно или несколько слов с параметрами. Параметры могут быть опциональными, при этом некоторые команды могут не иметь параметров.

Пример команды:

```
(als_sw) #network parms <ip> <subnet> [<gateway>]
```

- network parms — команда;
- <ip> и <subnet> — обязательные параметры;
- [<gateway>] — необязательный (опциональный) параметр.

В документации используются следующие обозначения:

- command — текстовая команда;
- <param> — параметр, который должен ввести пользователь;
- [command] — необязательная текстовая команда, которую можно опустить;
- [<param>] — необязательный параметр, который может ввести пользователь;
- <command1|command2> — два (или более) вариантов, которые нужно выбрать и ввести пользователю.

Автодополнение делает работу с CLI более удобной и быстрой. Данная функция позволяет дополнить набираемую команду нажатием на клавишу [Tab], если она может быть однозначно определена по уже введенному фрагменту. Если введенный фрагмент соответствует нескольким командам, ввод нужно будет продолжить вручную.

Например:

```
(als_sw) #sh[Tab]
(als_sw) #show

(als_sw) #show n[Tab]
(als_sw) #show network
```

Сокращенные формы команд могут применяться во время набора простой команды. Данная функция позволяет выполнить команду, введя нескольких первых символов, однозначно определяющих данную команду. В противном случае ввод команды или параметра нужно будет продолжить.

Например:

```
(als_sw) #con
(als_sw) (configure) #ex
(als_sw) #
```

Некоторые команды могут использоваться совместно со служебным словом **no**. Такой синтаксис применяется для достижения обратного эффекта по отношению к следующей за ним команде.

Например, следующая команда выключает первый интерфейс:

```
(als_sw) #con
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #shutdown
```

Отменяя ее действие, команда **no shutdown** включает первый интерфейс:

```
(als_sw) (configure) (interface 0/1) #no shutdown
```

Помощь

Помощь по доступным командам, их синтаксису и параметрам выводится при введении знака вопроса (?). Знак вопроса в командной строке не отображается.

Пример получения полного списка команд:

```
(als_sw) >[?]

enable                    Enter into user privilege mode / Set the password for

                          the enable privilege level
logout                    Terminate current user session
ping                      Send ICMP echo packets
quit                      Terminate current user session without saving
                          configuration
show                      Show device settings
traceroute                Trace network route using UDP protocol and ICMP
                          response
```

Пример справки по командам, начинающимся с латинской буквы "s":

```
(als_sw) #s[?]

set                       Set device parameters
show                      Show device settings
sshcon                    Configure SSH connection parameters
```

Пример справки по параметрам команды:

```
(als_sw) #network parms [?]

<ipaddress>               Enter IP address
```

Сочетания клавиш

Для упрощения работы в командной строке на коммутаторах АЛСиТЕК есть несколько горячих клавиш:

- клавиши стрелок <Left>, <Right> — переводят курсор влево и вправо относительно текущего положения. Если курсор сдвинуть нельзя, не имеют эффекта;
- клавиши стрелок <Up>, <Down> — заменяют текущую введенную строку (если она есть) на предыдущую введенную команду (история). При этом нажатие клавиши <Up> приводит к движению назад по истории команд, а нажатие клавиши <Down> — вперед. Всего в истории сохраняется последние 10 успешно введенных команд плюс текущая введенная строка;
- клавиши <Home>, <End> — переход в начало и конец вводимой в данный момент строки;
- клавиши <Delete>, <Backspace> — удаление символа из вводимой строки на текущей позиции курсора (Delete) и на предыдущей (Backspace);
- сочетание <Ctrl>-<Z> — переход в корневой контекст настройки из любого контекста. Повторное нажатие не имеет эффекта;
- сочетание <Ctrl>-<C> — может прерывать выполнение некоторых команд (ping, traceroute и подобные).

2.2. Пользователи и привилегии

На коммутаторе существует два уровня доступа для пользователей: Read Only и Read/Write.

Доступ Read Only

Режим непривилегированного пользователя. Пользователь с уровнем доступа Read Only имеет права только на просмотр текущего состояния коммутатора, менять настройки он не может. Приглашение пользователя с уровнем доступа Read Only выглядит следующим образом:

```
User:guest
Password:

(als_sw) >
```

В примере использован гостевой вход по умолчанию. Непривилегированный пользователь может повысить свои права.

Доступ Read/Write

Режим привилегированного пользователя. Пользователь с уровнем доступа Read/Write имеет права на просмотр состояния коммутатора, а так же на изменение конфигурации коммутатора и выполнение любых команд. Приглашение пользователя Read/Write выглядит следующим образом:

```
User:admin
Password:

(als_sw) #
```

В примере использован вход пользователя Read/Write по умолчанию. Привилегированный пользователь может понизить свои права.

2.3. Конфигурационные контексты

Для удобства настройки группы команд объединяются в контексты. По приглашению командной строки всегда можно определить, в каком контексте она находится. Группировка идет по логическому принципу. Например, команды настройки интерфейсов — в контексте интерфейсов, команды общей настройки — в контексте глобального конфигурирования. Управление VLAN осуществляется в контексте VLAN.

Некоторые службы имеют собственные контексты. В качестве примеров можно привести настройку списков доступа ACL и настройку параметров доступа к коммутатору (Console, Telnet, SSH).

Выход из любого контекста производится командой "exit". После ввода этой команды командная строка перейдет в предыдущий контекст, если он есть, вплоть до корневого контекста.

Корневой контекст

Вид контекста (привилегированный пользователь):

```
(als_sw) #
```

Вид контекста (непривилегированный пользователь):

```
(als_sw) >
```

В этом контексте выполняются базовые настройки коммутатора:

- настройка сетевых параметров коммутатора;
- очистка конфигурации;
- операции копирования (образов ПО, конфигурации, резервное копирование);
- просмотр текущего состояния командами, начинающимися со слова "show".

Из этого контекста осуществляется переход в другие контексты.

Контекст настройки VLAN

Контекст содержит команды по настройке обрабатываемых коммутатором VLAN. Вход в контекст:

```
(als_sw) #vlan database
(als_sw) (Vlan) #
```

Контекст глобальной настройки

Контекст содержит команды по настройке основных параметров коммутатора. Переход в контекст осуществляется командой:

```
(als_sw) #configure
(als_sw) (configure) #
```

Из этого контекста осуществляется переход в контексты интерфейсов и системы жизнеобеспечения.

Контекст настройки интерфейсов

Контекст содержит команды по настройке интерфейсов. Переход в контекст осуществляется из контекста глобальной настройки:

```
(als_sw) (configure) #interface <slot>/<port>
(als_sw) (configure) (interface <slot>/<port>) #
```

Контекст настройки интерфейсов может объединять несколько интерфейсов и/или их диапазоны:

```
(als_sw) (configure) #interface 0/1,0/4-0/8,0/11
(als_sw) (configure) (interface 0/1,0/4-0/8,0/11) #
```

Варианты ввода:

- несколько интерфейсов, идущих подряд (интервал): "interface 0/1-0/10";
- несколько интерфейсов, перечисленных через запятую (перечисление): "interface 0/1,0/2,0/8,0/16,0/24";
- комбинированный вариант: "interface 0/1,0/2-0/5,0/8,0/24-0/28".

Если в текущем контексте настройки интерфейсов при входе было указано несколько интерфейсов, как в примерах выше, то выполняемые в этом контексте команды будут применены на все интерфейсы из набора в порядке их следования. Это касается любых команд, выполняемых в контексте настройки интерфейсов. Команда, выполненная на наборе интерфейсов, по действию идентична команде, выполненной на каждом интерфейсе поочередно.

2.4. Команды копирования

Образы программного обеспечения

На энергонезависимой flash-памяти коммутаторов АЛСиТЕК размещается два образа программного обеспечения (image1 и image2). При загрузке выбирается один из этих образов, который считается активным. При обновлении ПО желательно использовать неактивный образ, чтобы в случае сбоя или потери питания во время прошивки коммутатор загрузился с исправного образа и был доступен.

Текущее состояние образов и их версии можно посмотреть командой:

```
(als_sw) #show bootvar

Active image        : image2
Next active image   : image2

Images versions:
image1 version      : 1.0.0.22
image2 version      : 1.0.0.22
```

На листинге видно, что текущий загруженный образ **image2** (*Active image*), при следующей загрузке будет также выбран образ **image2** (*Next active image*). Также показаны версии обоих образов, в данном случае это 1.0.0.22.

Для изменения образа, с которого будет производится загрузка, используется команда:

```
(als_sw) #boot system <image1|image2>
```

Продолжая предыдущий пример, выберем образ **image1** и просмотрим внесенные изменения:

```
(als_sw) #boot system image1

Activating image1...

(als_sw) #show bootvar

Active image        : image2
Next active image   : image1

Images versions:
image1 version      : 1.0.0.22
image2 version      : 1.0.0.22
```

Теперь после перезагрузки коммутатора для загрузки будет выбран образ **image1**.

Работа с конфигурацией

Коммутаторы АЛСиТЕК работают со следующими конфигурациями:

- startup-config — конфигурация, расположенная на энергонезависимой flash-памяти коммутатора. Эта конфигурация применяется при его запуске;
- running-config — конфигурация коммутатора на текущий момент. При старте устройства она загружается из startup-config, но в процессе работы конфигурация может быть изменена;
- backup-config — конфигурация, расположенная на энергонезависимой flash-памяти коммутатора. Может быть полезна для сохранения альтернативной конфигурации.

При изменении конфигурации коммутатора изменения заносятся в running-config и сразу применяются. Изменения в running-config не затрагивают startup-config и backup-config, и работают до перезагрузки коммутатора.

Для просмотра конфигураций используются соответствующие команды:

```
(als_sw) #show running-config

...
```

```
(als_sw) #show startup-config

...
```

```
(als_sw) #show backup-config

...
```

Для сохранения текущей конфигурации (running-config) на flash в startup-config используется команда:

```
(als_sw) #write memory
This operation may take a few minutes.
Management interfaces will not be available during this time.
Are you sure you want to save? (y/n): y
Config file 'startup-config' created successfully
Configuration Saved!
```

После выполнения этой команды текущая конфигурация будет записана на flash в startup-config и они станут одинаковыми.

Для очистки текущей конфигурации используется команда:

```
(als_sw) #clear config
Are you sure you want to clear the configuration? (y/n): y
Clearing configuration. Please wait for login prompt.

User:
```

Эта команда полностью очищает текущую конфигурацию, после чего все пользователи, подключенные к коммутатору по COM, Telnet или SSH, будут принудительно отключены. Это касается всех сессий управления, в том числе и той, с которой была выполнена команда очистки конфигурации. Обратите внимание — адрес коммутатора и VLAN управления будут сброшены в значения по умолчанию, что может привести к потере управления устройством. Очистка текущей конфигурации **не затрагивает** конфигурации, сохраненные на flash-памяти (startup-config и backup-config).

На некоторых моделях коммутаторов есть также аппаратная кнопка сброса конфигурации. При удержании этой кнопки в нажатом состоянии более 5 секунд текущая конфигурация коммутатора очищается. Для удобства использования при этом включается прерывистая светодиодная сигнализация на интерфейсах коммутатора (с частотой 2 раза в секунду).

Конфигурации, сохраненные на flash-памяти (startup-config и backup-config), при этом не затрагиваются — если перезагрузить коммутатор сразу после очистки конфигурации аппаратной кнопкой, конфигурация будет восстановлена из flash-памяти (startup-config).

Перезагрузка коммутатора

Перезагрузка коммутатора производится командой:

```
(als_sw) #reload
The system has unsaved changes.
Would you like to save them now? (y/n): y
Config file 'startup-config' created successfully.
Configuration Saved!
System will now restart!
```

После ввода этой команды коммутатор проверит состояние running-config и startup-config. Если они отличаются, об этом будет сообщено пользователю, и будет выведено предложение сохранить текущую конфигурацию на flash. Если ответить утвердительно, текущая конфигурация будет сохранена на flash в startup-config, после чего коммутатор перезагрузится.

В случае отрицательного ответа коммутатор уточнит, нужна ли перезагрузка без сохранения конфигурации:

```
(als_sw) #reload
The system has unsaved changes.
Would you like to save them now? (y/n): n
Configuration Not Saved!
Are you sure you would like to reset the system? (y/n): n
```

В примере выше был дан отрицательный ответ и перезагрузка была отменена.

Отложенная перезагрузка

В некоторых случаях при настройке коммутатора может возникнуть необходимость подстраховаться. К примеру, удаленно применяется экспериментальный файл конфигурации, при применении которого может быть потеряно управление коммутатором. Чтобы избежать потери управления и отменить все внесенные изменения, можно использовать отложенную перезагрузку.

Для включения отложенной перезагрузки используется команда:

```
(als_sw) #reload withdelay <seconds>
```

Параметры:

- <seconds> — через сколько секунд должна быть произведена перезагрузка, число от 1 до 86400.

Отсчет времени начинается с момента ввода команды. Отложенная перезагрузка является безусловной, она не сохранит текущую конфигурацию. При этом есть возможность отменить отложенную перезагрузку командой:

```
(als_sw) #no reload withdelay
```

Команду отмены перезагрузки можно использовать в случае успешного применения конфигурации из примера выше, когда управление коммутатором не пропало.

Команды копирования

Команды копирования осуществляют обновление и сохранение конфигураций и программного обеспечения коммутатора. Во время работы любой из команд копирования выполнение всех остальных команд блокируется до завершения копирования. Все пользователи оповещаются об этом специальным сообщением:

```
INFO: Single-user command was called in current or another control session, man
agement access is blocked for the duration of the single-user command
```

После завершения команды копирования все пользователи оповещаются об этом сообщением:

```
INFO: Single-user command was completed in current or another control session,
management access is restored
```

Любая команда копирования требует подтверждения для выполнения:

```
Management access will be blocked for the duration of the transfer
Are you sure you want to start? (y/n):
```

Успешное копирование сопровождается сообщением:

```
File transfer operation completed successfully.
```

Неудавшееся копирование сопровождается сообщением:

```
ERROR: File transfer failed!
```

Копирование осуществляется по протоколам TFTP и FTP. Синтаксис команд копирования для обоих протоколов практически одинаковый.

Единственным отличием является поддержка протоколом FTP использования имени пользователя и пароля. В этом случае перед IP-адресом FTP-сервера добавляется логин (имя пользователя), после чего коммутатор запрашивает пароль:

```
(als_sw) #copy ftp://user@192.168.1.1/config.cfg nvram:startup-config

Mode........................................... FTP
Set Server IP.................................. 172.16.67.39
Username....................................... user
Path........................................... ./
Filename....................................... config.cfg
Data Type...................................... Text Configuration

Enter password:
```

Если для работы с сервером FTP имя пользователя не используется, синтаксис команд полностью совпадает с синтаксисом аналогичных команд для работы с сервером TFTP:

```
(als_sw) #copy ftp://192.168.1.1/config.cfg nvram:startup-config
```

Во всех дальнейших примерах используется протокол TFTP.

Копирование на сервер

На сервер с коммутатора можно скопировать конфигурацию и образы ПО.

Копирование конфигурации

Копирование стартовой конфигурации:

```
(als_sw) #copy nvram:startup-config tftp://192.168.1.1/config.cfg

Mode........................................... TFTP
Set Server IP.................................. 192.168.1.1
Path........................................... ./
Filename....................................... config.cfg
```

```
Data Type..................................... Text Configuration

Management access will be blocked for the duration of the transfer
Are you sure you want to start? (y/n): y

INFO: Single-user command was called in current or another control session, man
agement access is blocked for the duration of the single-user command

TFTP Text Configuration transfer starting...
File transfer operation completed successfully.

INFO: Single-user command was completed in current or another control session,
management access is restored
```

Копирование резервной конфигурации:

```
(als_sw) #copy nvram:backup-config tftp://192.168.1.1/config.cfg

Mode........................................... TFTP
Set Server IP.................................. 192.168.1.1
Path........................................... ./
Filename....................................... config.cfg

Management access will be blocked for the duration of the transfer
Are you sure you want to start? (y/n): y

INFO: Single-user command was called in current or another control session, man
agement access is blocked for the duration of the single-user command

TFTP Text Configuration transfer starting...
File transfer operation completed successfully.

INFO: Single-user command was completed in current or another control session,
management access is restored
```

Копирование текущей конфигурации:

```
(als_sw) #copy running-config tftp://192.168.1.1/config.cfg

Mode........................................... TFTP
Set Server IP.................................. 192.168.1.1
Path........................................... ./
Filename....................................... config.cfg
Data Type..................................... Text Configuration

Management access will be blocked for the duration of the transfer
Are you sure you want to start? (y/n): y

INFO: Single-user command was called in current or another control session, man
agement access is blocked for the duration of the single-user command

TFTP Text Configuration transfer starting...
```

```
File transfer operation completed successfully.

INFO: Single-user command was completed in current or another control session,
management access is restored
```

Копирование программного обеспечения

```
(als_sw) #copy image1 tftp://192.168.1.1/image1.img

Mode..................................... TFTP
Set Server IP............................ 192.168.1.1
Path..................................... ./
Filename................................. image1
Data Type................................ Code
Destination Filename..................... image1.img

Management access will be blocked for the duration of the transfer
Are you sure you want to start? (y/n): y

INFO: Single-user command was called in current or another control session, man
agement access is blocked for the duration of the single-user command

TFTP Code transfer starting...
File transfer operation completed successfully.

INFO: Single-user command was completed in current or another control session,
management access is restored
```

Копирование на сервер с использованием лексем

При передаче файлов с коммутатора на сервер и обратно можно использовать лексемы в строке URL, например время копирования и имя коммутатора. Это может быть удобно при использовании функции автоматического копирования конфигурации при ее сохранении.

Доступные лексемы:

- $a — IP-адрес коммутатора, например "172.17.1.1";
- $h — имя хоста коммутатора, например "als_sw";
- $t — текущая дата и время коммутатора в формате YYYYMMDDhhmmss, например "20150604101537" (4 июня 2015 года, 10:15:37).

Ниже представлен пример копирования стартовой конфигурации на TFTP-сервер в директорию "/backup/" с использованием лексем в имени файла:

```
(als_sw) #copy nvram:startup-config tftp://172.17.1.7/backup/config_$a_$h_$t.tx
t

Mode..................................... TFTP
Set Server IP.............................. 172.17.1.7
Path..................................... backup/
Filename................................. config_172.17.1.1_als_sw_201506
04101537.txt
Data Type................................ Text Configuration

Management access will be blocked for the duration of the transfer
Are you sure you want to start? (y/n): y

INFO: Single-user command was called in current or another control session, man
agement access is blocked for the duration of the single-user command

TFTP Text Configuration transfer starting...

File transfer operation completed successfully.

INFO: Single-user command was completed in current or another control session,
management access is restored
```

Примечание: если файл не существует на TFTP-сервере, то для его создания необходима соответствующая настройка TFTP-сервера (создание файлов должно быть разрешено).

Настройка автоматического копирования на сервер

Включить автоматическое сохранение конфигурации можно командой:

```
(als_sw) #copy nvram:startup-config tftp://172.17.1.7/backup/config_$a_$t.bkp o
n-save
```

Обратите внимание, что в момент вызова этой команды само копирование не происходит. Команда сохраняется в конфигурации. После включения автоматического копирования при сохранении изменений в конфигурации новая версия будет скопирована по указанному URL:

```
(als_sw) #write memory
This operation may take a few minutes.
Management interfaces will not be available during this time.
Are you sure you want to save? (y/n): y
Config file 'startup-config' created successfully
Configuration Saved!
Configuration successfully saved to tftp://172.17.1.7/backup/config_172.17.1.1_
20150604111207.bkp
```

Если при сохранении конфигурации произошла ошибка, например TFTP-сервер недоступен в течение 10 секунд, будет выведена ошибка следующего вида:

```
(als_sw) #write memory
This operation may take a few minutes.
Management interfaces will not be available during this time.
Are you sure you want to save? (y/n): y
Config file 'startup-config' created successfully
Configuration Saved!
ERROR: Failed to copy nvram:startup-config to tftp://172.17.1.7/backup/config_1
72.17.1.1_20150604111234.bkp
```

Копирование с сервера

Копирование текстового приглашения (CLI Banner)

Коммутатор позволяет установить текстовое приглашение, которое будет выводиться при каждом подключении к устройству по интерфейсу COM, Telnet или SSH. В этом сообщении может содержаться произвольный текст, заданный пользователем. Загрузка текстового файла на устройство осуществляется из привилегированного режима по протоколу TFTP или FTP.

Требования к текстовому файлу:

- допустимо использование только печатных символов ASCII;
- размер исходного файла: не более 2048 байт;
- исходное количество строк в файле: не более 20.

В случае невыполнения хотя бы одного из этих условий текстовый файл не будет принят, а его загрузка окончится неудачей.

Загрузить файл текстового приглашения на устройство можно командой:

```
(als_sw) #copy tftp://172.17.1.2/banner.txt nvram:clibanner
```

После успешной передачи CLI Banner сразу же устанавливается, и при входе на устройство вы увидите его содержимое в консоли:

```
$ telnet 172.17.1.1
Trying 172.17.1.1...
Connected to 172.17.1.1.
Escape character is '^]'.
--------------------
  Hello, World!!!
--------------------

User:
```

Удалить Cli Banner можно командой:

```
(als_sw) #clear clibanner
```

Эта команда безвозвратно удаляет загруженный файл Cli Banner.

Копирование конфигурации

Стартовую конфигурацию коммутатора можно загрузить целиком с TFTP или FTP сервера с помощью команды:

```
(als_sw) #copy tftp://192.168.1.1/config.cfg nvram:startup-config

Mode..................................... TFTP
Set Server IP............................ 192.168.1.1
Path..................................... ./
Filename................................. config.cfg
Data Type................................ Text Configuration

Management access will be blocked for the duration of the transfer
Are you sure you want to start? (y/n): y

INFO: Single-user command was called in current or another control session, man
agement access is blocked for the duration of the single-user command

TFTP Text Configuration transfer starting...
File transfer operation completed successfully.

INFO: Single-user command was completed in current or another control session,
management access is restored
```

После выполнения этой команды стартовая конфигурация будет загружена на коммутатор с сервера, сохранена на flash, после чего применена. При применении конфигурации сессия управления будет завершена.

Есть возможность выполнить копирование с сервера конфигурации и использовать ее в качестве резервной. Указанное действие выполняется с помощью команды:

```
(als_sw) #copy tftp://192.168.1.1/config.cfg nvram:backup-config

Mode..................................... TFTP
Set Server IP............................ 192.168.1.1
Path..................................... ./
Filename................................. config.cfg

Management access will be blocked for the duration of the transfer
Are you sure you want to start? (y/n): y

INFO: Single-user command was called in current or another control session, man
agement access is blocked for the duration of the single-user command

TFTP Text Configuration transfer starting...
File transfer operation completed successfully.
```

```
INFO: Single-user command was completed in current or another control session,
management access is restored
```

Выполнить копирование конфигурации с сервера и применить ее в качестве текущей можно с помощью команды:

```
(als_sw) #copy tftp://192.168.1.1/config.cfg running-config

Mode........................................... TFTP
Set Server IP.................................. 192.168.1.1
Path........................................... ./
Filename....................................... config.cfg
Data Type...................................... Text Configuration

Management access will be blocked for the duration of the transfer
Are you sure you want to start? (y/n): y

INFO: Single-user command was called in current or another control session, man
agement access is blocked for the duration of the single-user command

TFTP Text Configuration transfer starting...
File transfer operation completed successfully.
Applying configuration, please wait ...

User:
```

После исполнения указанной команды скопированная с сервера конфигурация заменит текущую конфигурацию коммутатора, но не будет сохранена на flash. Все сессии управления будут разорваны и пользователи получат приглашение авторизации.

Копирование программного обеспечения коммутатора

Образы ПО image1 и image2 можно загрузить на коммутатор с сервера TFTP или FTP с помощью команды:

```
(als_sw) #copy tftp://192.168.1.1/phoenix-rtk-mips-1.0.0.21-f.stk image1

Mode..................................... TFTP
Set Server IP............................ 192.168.1.1
Path..................................... ./
Filename................................. phoenix-rtk-mips-1.0.0.21-f.stk

Data Type................................ Code
Destination Filename..................... image1

Management access will be blocked for the duration of the transfer
Are you sure you want to start? (y/n): y

INFO: Single-user command was called in current or another control session, man
agement access is blocked for the duration of the single-user command

TFTP Code transfer starting...
File transfer operation completed successfully.

INFO: Single-user command was completed in current or another control session,
management access is restored
```

Копирование на коммутаторе

Непосредственно на коммутаторе можно произвести копирование одного образа программного обеспечения в другой:

```
(als_sw) #copy image1 image2

Management access will be blocked for the duration of the transfer
Are you sure you want to start? (y/n): y

INFO: Single-user command was called in current or another control session, man
agement access is blocked for the duration of the single-user co
mmand

Copying image1 to image2...
.....................

File transfer operation completed successfully.

INFO: Single-user command was completed in current or another control session,
management access is restored
```

Также непосредственно на коммутаторе можно производить копирование стартовой конфигурации (startup-config) в резервную (backup-config) или наоборот. Для примера приведена команда сохранения стартовой конфигурации в резервную:

```
(als_sw) #copy nvram:startup-config nvram:backup-config
Management access will be blocked for the duration of the transfer
Are you sure you want to start? (y/n): y
INFO: Single-user command was called in current or another control session, man
agement access is blocked for the duration of the single-user command

Text Configuration transfer starting...
File transfer operation completed successfully.

INFO: Single-user command was completed in current or another control session,
management access is restored
```

После выполнения этой команды стартовая конфигурация будет записана в backup-config и они станут одинаковыми.

Можно непосредственно сохранить текущую конфигурацию (running-config) в резервную (backup-config), а также произвести восстановление резервной конфигурации. Для примера приведена команда сохранения текущей конфигурации:

```
(als_sw) #copy running-config nvram:backup-config
Management access will be blocked for the duration of the saving operation
Are you sure you want to save current configuration to backup file? (y/n): y
Backup configuration file 'backup-config' created successfully
```

При этом стартовая конфигурация (startup-config) в энергонезависимой flash-памяти коммутатора не изменяется.

Слияние конфигураций

Иногда бывает удобно сохранить общую часть конфигурации, а затем добавлять ее к специальным настройкам. Для этого на коммутаторе присутствует возможность слияния резервной конфигурации (backup-config) с текущей (running-config), или стартовой конфигурации (startup-config) с текущей. Слияние производится построчным применением команд из резервной (или стартовой) конфигурации к текущей. Соответственно, отсутствующие настройки появятся в текущей конфигурации, а взаимоисключающие — придут в соответствие с настройками в резервной (или стартовой) конфигурации. Для примера приведена команда слияния стартовой конфигурации с текущей:

```
(als_sw) #copy nvram:startup-config running-config merge
Management access will be blocked for the duration of merge and preserve all th
e commands from both configurations
Are you sure you want to merge startup configuration to current configuration?
(y/n): y
Startup configuration merged successfully
```

ГЛАВА 3. АУТЕНТИФИКАЦИЯ

3.1. Введение

Для защиты коммутатора от несанкционированного доступа и изменения настроек используется проверка пользователей по логину и паролю. При подключении пользователя ему предлагается ввести логин и пароль, после проверки введенных данных пользователю будет либо разрешен вход и назначен определенный уровень привилегий, либо вход будет запрещен. Проверка введенных пользователем данных может осуществляться как на самом коммутаторе, локально, так и на удаленном сервере. Настройка аутентификации на удаленном сервере позволяет упростить управление пользователями и поддерживать централизованную базу данных пользователей вместо распределенных по коммутаторам сети настроек.

Локальная аутентификация

Локальная аутентификация производится на коммутаторе в соответствии с его настройками. Пользователи и пароли записываются в конфигурацию коммутатора и используются при проверке вводимых пользователями данных. Пароли хранятся в зашифрованном виде. Изменить списки пользователей и их паролей возможно только войдя на коммутатор с административным уровнем доступа.

Аутентификация с помощью серверов RADIUS или TACACS+

Для проверки введенных пользователем данных могут быть использованы протоколы RADIUS и TACACS+. Также есть возможность настроить последовательную проверку несколькими методами, на случай, если сервер аутентификации вышел из строя или недоступен с коммутатора.

Общий принцип работы серверов RADIUS и TACACS+ для аутентификации пользователя:

1. Пользователь при входе на коммутатор вводит логин и пароль.
2. Коммутатор соединяется с сервером аутентификации (RADIUS или TACACS+).
3. Коммутатор передает на сервер по защищенному каналу данные, введенные пользователем (логин и пароль), и запрашивает

разрешение на вход.
4. Сервер проверяет полученные от коммутатора данные пользователя, и на основании своих настроек выдает либо разрешение на вход, либо запрет.

3.2. Настройка аутентификации на коммутаторах АЛСиТЕК

Основным элементом, управляющим порядком проверки логинов и паролей пользователей, является список доступа. Список доступа задает порядок проверки введенного пользователем логина и пароля различными методами. По умолчанию на коммутаторе настроено два списка доступа с именами **defaultList** и **networkList**.

Список доступа применяется к определенному типу подключения (Console, Telnet, SSH) и содержит в себе список методов для проверки пользователей (none, local, radius, tacacs). В одном списке доступа может быть несколько методов проверки, при такой настройке они будут проверяться по порядку, пока не будет получено разрешение или запрет на вход пользователя.

Настройка локальных пользователей

Шаг 1. Просмотр текущего списка пользователей

По умолчанию на коммутаторе есть всего два пользователя: admin и guest. Пароли обоих пользователей пустые для облегчения первичной настройки коммутатора. Уровень доступа пользователя admin — Read/Write, что соответствует административному уровню доступа, уровень пользователя guest — Read Only, то есть этот пользователь может выполнять только команды просмотра текущего состояния коммутатора и не может менять настройки.

Для просмотра текущего списка пользователей используется команда:

```
(als_sw) #show users

User Name                      User Access Mode
-------------------------  ----------------
admin                          Read/Write
guest                          Read Only
```

Шаг 2. Добавление нового пользователя

В общем виде команда добавления нового пользователя выглядит так:

```
(als_sw) #configure
(als_sw) (configure) #username <login> password <password> [level <level>] [enc
rypted]
```

Параметры:

- <login> — имя пользователя. Должно быть уникальным в пределах конфигурации коммутатора;
- <password> — пароль пользователя. Может быть задан открытым текстом, либо в зашифрованном виде. Зашифрованный вид также используется при выводе конфигурации. При вводе пароля в открытом виде длина пароля может быть от 8 до 64 символов (при этом минимальная длина пароля может быть настроена в конфигурации с помощью команды passwords min-length <0-64> в контексте configure), разрешены символы от "a" до "z", от "A" до "Z", от "0" до "9" и символ "_". При вводе пароля в зашифрованном виде он должен иметь длину 128 символов, разрешены символы от "a" до "f" и от "0" до "9";
- <level> — необязательный параметр, который может принимать значения 1 или 15, которые соответствуют уровню доступа Read Only и Read/Write. Если параметр опущен, значение принимается равным 1 (Read Only пользователь);
- [encrypted] — флаг, который указывается в случае, если пароль задан в зашифрованном виде.

Добавим нового пользователя с уровнем доступа Read Only:

```
(als_sw) #configure
(als_sw) (configure) #username user1 password 12345678 level 1
```

Здесь мы добавили пользователя с логином "user1", паролем "12345678" и уровнем доступа Read Only.

Добавим еще одного пользователя, указав пароль "12345678" в зашифрованном виде:

```
(als_sw) #configure
(als_sw) (configure) #username "user2" password 928336800ccca32a4c218d5e93108e7
0239a6a11664300a29598ec3b4771b69fd1711bcccd59d465b2309c18879c50347d786118e6d332a
4b21b337c620dcc5c level 1 encrypted
```

Создадим еще одного пользователя с административным доступом:

```
(als_sw) #configure
(als_sw) (configure) #username "admin2" password "VeryStrongPassword123_x8" lev
el 15
```

Шаг 3. Изменение уровня доступа или пароля пользователей

Для того чтобы изменить уровень доступа или пароль пользователя, нужно повторно ввести ту же команду, что и при создании. Пароль или уровень доступа будут обновлены у уже существующего пользователя.

Изменим пароль у пользователя "admin2" из предыдущего примера:

```
(als_sw) #configure
(als_sw) (configure) #username "admin2" password "VeryStrongPassword123_x16" le
vel 15
```

Изменим уровень доступа у пользователя "user1":

```
(als_sw) #configure
(als_sw) (configure) #username user1 password 12345678 level 15
```

Теперь пользователь "user1" имеет уровень доступа Read/Write, в чем можно убедиться, просмотрев список пользователей:

```
(als_sw) #show users

User Name                  User Access Mode
-------------------------  ----------------
...
user1                      Read/Write
...
```

Шаг 4. Удаление пользователя

Для удаления пользователя используется команда:

```
(als_sw) #configure
(als_sw) (configure) #no username <login>
```

Удалим пользователя с логином "user1" из предыдущих примеров:

```
(als_sw) #configure
(als_sw) (configure) #no username user1
```

Этой командой также можно удалить пользователей по умолчанию — "admin" и "guest".

Настройка методов аутентификации

По умолчанию на коммутаторе присутствует два списка доступа для проверки логинов и паролей пользователей: **defaultList** и **networkList**. Кроме того, по умолчанию существует специальный список доступа **enableList**, который управляет методом проверки пароля для команды повышения привилегий **enable**. Текущее состояние списков доступа и методов проверки можно просмотреть командой:

```
(als_sw) #show authentication methods

Login Authentication Method Lists
---------------------------------
defaultList         :  local
networkList         :  local

Enable Authentication Method Lists
----------------------------------
enableList          :  enable

Line     Login Method List    Enable Method List
-------  -----------------    ------------------
Console  defaultList          enableList
Telnet   networkList          enableList
Ssh      networkList          enableList
```

В первой таблице показано, что существует два списка доступа (по умолчанию). Метод проверки логина и пароля при входе у этих списков доступа установлен в **local**. Метод проверки может принимать следующие значения:

- none — не проверять логин и пароль. В этом случае любой вход будет разрешен без проверок, пароль не проверяется;
- local — логин и пароль при входе проверяются по текущему списку пользователей, заданных в конфигурации коммутатора. Значение по умолчанию;
- radius — для проверки входа коммутатор будет обращаться к настроенным RADIUS-серверам в порядке их приоритета. Если сервер не отвечает в течение определенного времени, запрос будет направлен следующему серверу (если он есть);
- tacacs — для проверки входа коммутатор будет обращаться к настроенным TACACS+-серверам в порядке их приоритета. Если сервер не отвечает в течение определенного времени, запрос будет направлен следующему серверу (если он есть);

Изменить метод проверки логина и пароля можно командой:

```
(als_sw) #configure
(als_sw) (configure) #aaa authentication login <default|<list name>> <method> [
<method>]
```

- default — список доступа по умолчанию (для соединений telnet и SSH это **networkList**, для COM это **defaultList**);
- <list name> — имя ранее созданного списка доступа;
- <method> — один из методов проверки логина и пароля, перечисленных выше. Методов можно указать несколько.

Для примера отключим проверку логина и пароля для списка доступа **defaultList**:

```
(als_sw) #configure
(als_sw) (configure) #aaa authentication login defaultList none
```

Для одного списка доступа можно назначить несколько методов проверки.

Например, назначим последовательную проверку по всем методам для списка **networkList**:

```
(als_sw) #configure
(als_sw) (configure) #aaa authentication login networkList tacacs radius local
```

Такая настройка приведет к следующему эффекту: сначала введенный логин и пароль будет проверен на настроенных TACACS+-серверах. Если ни один из TACACS+-серверов не ответил (или ни одного сервера не настроено), проверка продолжится на всех настроенных RADIUS-серверах. Если ни один из RADIUS-серверов не ответил (или они не настроены), проверка будет произведена по локальной базе пользователей. В случае получения ответа на любом этапе (отрицательного или положительного) проверка прекращается.

Вторая таблица показывает, какой метод проверки установлен для пароля команды **enable**. Эта команда используется для повышения привилегий пользователями с уровнем доступа Read Only. То есть пользователь может войти как гость, а затем ввести специальную команду **enable**, после чего ввести верный пароль повышения привилегий и получить права администратора.

Метод проверки пароля в команде **enable** может принимать следующие значения:

- none — не проверять пароль. В этом случае введенный пароль не будет проверяться, любой пользователь сможет получить привилегии администратора;
- enable — пароль проверяется локально и задается в конфигурации коммутатора. По умолчанию пароль пустой, то есть любой гость может получить привилегии администратора;
- radius — для проверки пароля коммутатор будет обращаться к настроенным RADIUS-серверам со специальным запросом проверки разрешения повышения привилегий;
- tacacs — для проверки пароля коммутатор будет обращаться к настроенным TACACS+-серверам со специальным запросом проверки разрешения повышения привилегий.

Способ проверки пароля **enable** можно сменить командой:

```
(als_sw) #configure
(als_sw) (configure) #aaa authentication enable <default|<list name>> <method>
[<method>]
```

- default — список доступа по умолчанию (для всех соединений это список **enableList**);
- <list name> — имя раннее созданного списка доступа;
- <method> — один из методов проверки логина и пароля, перечисленных выше. Методов можно указать несколько.

Состояние по умолчанию установлено в метод **enable**, то есть введенный пароль будет сравниваться с сохраненным в конфигурации коммутатора паролем.

Сам локальный пароль можно задать в конфигурации коммутатора командой:

```
(als_sw) #enable password <password> [encrypted]
```

Флаг **encrypted** указывает на то, что пароль был введен в зашифрованном виде, так же как и при создании пользователей. Пример установки:

```
(als_sw) #enable password "SuperEnablePassword_saved"
```

Третья таблица задает соответствие протоколов входа (COM, Telnet, SSH) и списков доступа. Для разных протоколов можно установить различные списки доступа, в том числе и создать собственные.

Работа со списками доступа

Список доступа содержит список методов проверки аутентификации (none, local, radius, tacacs) и может быть применен к определенному типу подключения (COM, telnet, SSH). Список доступа назначается каждому из явно, в соответствующем контексте.

Шаг 1. Просмотр текущего состояния списков доступа и методов проверки

Текущее состояние можно узнать командой:

```
(als_sw) #show authentication methods

Login Authentication Method Lists
---------------------------------
defaultList         :  local
networkList         :  local

Enable Authentication Method Lists
----------------------------------
enableList          :  enable

Line     Login Method List    Enable Method List
-------  -----------------    ------------------
Console  defaultList          enableList
Telnet   networkList          enableList
Ssh      networkList          enableList
```

Из этих таблиц видно, что для входа по COM-порту (*Console*) используется список доступа **defaultList**, метод проверки которого "local". Для удаленных соединений по Telnet и Ssh используется список доступа **networkList**, метод проверки которого также "local". Для всех трех протоколов (COM, Telnet, SSH) для проверки пароля команды повышения привилегий используется список доступа **enableList**, метод проверки которого установлен в **enable**, то есть локальная проверка пароля, заданного конфигурацией.

По умолчанию сервера RADIUS и TACACS+ не используются.

Шаг 2. Создание пользовательского списка доступа

Для создания нового списка доступа используется команда:

```
(als_sw) #configure
(als_sw) (configure) #aaa authentication login uLoginList local
(als_sw) (configure) #aaa authentication enable uEnableList enable
```

В данном случае мы создали новый список доступа с именем **uLoginList** и назначили метод проверки логина и пароля при входе пользователей в **local**. Также был создан список доступа для проверки пароля команды повышения привилегий с именем **uEnableList** и установлен метод проверки пароля **enable**, то есть проверка будет идти по настроенному в конфигурации паролю.

Шаг 3. Назначение списка доступа

Для того чтобы список доступа заработал, его нужно назначить для протоколов входа командами:

```
(als_sw) #configure
(als_sw) (configure) #line console
(als_sw) (configure) (line) #login authentication uLoginList
(als_sw) (configure) (line) #enable authentication uEnableList
(als_sw) (configure) (line) #exit

(als_sw) (configure) #line telnet
(als_sw) (configure) (telnet) #login authentication uLoginList
(als_sw) (configure) (telnet) #enable authentication uEnableList
(als_sw) (configure) (telnet) #exit

(als_sw) (configure) #line ssh
(als_sw) (configure) (ssh) #login authentication uLoginList
(als_sw) (configure) (ssh) #enable authentication uEnableList
(als_sw) (configure) (ssh) #exit
(als_sw) (configure) #exit
```

В этих трех блоках мы для всех протоколов входа назначили список доступа **uLoginList** (для проверки входа) и список **uEnableList** (для проверки пароля команды повышения привилегий).

После этих команд состояние списков доступа будет следующее:

```
(als_sw) #show authentication methods

Login Authentication Method Lists
---------------------------------
defaultList         :  local
networkList         :  local
uLoginList          :  local

Enable Authentication Method Lists
----------------------------------
enableList          :  enable
uEnableList         :  enable

Line     Login Method List    Enable Method List
-------  -----------------    ------------------
Console  uLoginList           uEnableList
Telnet   uLoginList           uEnableList
Ssh      uLoginList           uEnableList
```

Настройка коммутатора для использования RADIUS-сервера

RADIUS (англ. Remote Authentication in Dial-In User Service) — сетевой протокол, который обеспечивает централизованную аутентификацию, авторизацию и учет использования сетевых сервисов. Этот протокол также используется для систем тарификации.

Шаг 1. Указание IP-адреса сервера

Команда создания RADIUS-сервера в общем виде выглядит следующим образом:

```
(als_sw) #configure
(als_sw) (configure) #radius server host auth <server address> [name <name> [po
rt <port>]]
```

Для каждого RADIUS-сервера помимо IP-адреса можно указать его имя и нестандартный порт. Если параметры опущены, имя принимается равным "Default-RADIUS-Server", а номер порта — 1812.

Создадим новый RADIUS-сервер:

```
(als_sw) #configure
(als_sw) (configure) #radius server host auth 172.17.1.100 name "Main"
```

Серверов RADIUS может быть указано несколько (до 3), в этом случае коммутатор при проверке логина и пароля будет обращаться в порядке приоритета к каждому серверу RADIUS, пока не получит ответ. В случае получения разрешения или запрета аутентификации от сервера, остальные серверы опрашиваться не будут.

Добавим еще один сервер с указанием нестандартного порта:

```
(als_sw) #configure
(als_sw) (configure) #radius server host auth 172.17.1.111 port 18337
```

Также есть возможность назначить основной RADIUS-сервер командой:

```
(als_sw) #configure
(als_sw) (configure) #radius server primary 172.17.1.111
```

Основной RADIUS-сервер будет опрашиваться первым, а затем все остальные, если они есть, в порядке их добавления в конфигурацию.

Шаг 2. Указание ключа шифрования для обмена данными между коммутатором и сервером

Зададим ключ, который будет использоваться в процессе обмена данными между коммутатором и RADIUS-сервером:

```
(als_sw) #configure
(als_sw) (configure) #radius server key auth 172.16.67.39
Enter secret (16 characters max):******
Re-enter secret:******
```

Введенный ключ в поле secret должен совпадать с ключом, который указывается в конфигурации на сервере RADIUS. Этот ключ гарантирует, что переданные данные будут защищены.

Шаг 3. Включение проверки логина и пароля пользователей с помощью сервера RADIUS

Для включения проверки логина и пароля на сервере RADIUS необходимо добавить метод проверки в определенный список доступа:

```
(als_sw) #configure
(als_sw) (configure) #aaa authentication login networkList radius
```

В данном случае мы для списка доступа с именем **networkList** (по умолчанию он используется для доступа по протоколам Telnet и SSH) применили метод проверки "radius". Это значит, что при подключении пользователей по протоколу Telnet или SSH проверка логина и пароля будет проводится коммутатором на настроенных RADIUS-серверах.

Шаг 4. Включение проверки пароля команды повышения привилегий с помощью сервера RADIUS

Для включения проверки пароля команды повышения привилегий **enable** на RADIUS-сервере необходимо добавить метод проверки в определенный список доступа:

```
(als_sw) #configure
(als_sw) (configure) #aaa authentication enable enableList radius
```

В данном случае мы для списка доступа **enableList** применили метод проверки "radius". При попытке повышения привилегий проверка пароля будет проводиться коммутатором на настроенных RADIUS-серверах. Важно отметить, что для корректной проверки на RADUIS-сервере должен быть настроен пользователь **$enab15$** с атрибутом **Service-Type**, равным **Administrative-User**. Пароль этого пользователя и будет паролем команды повышения привилегий.

Настройка коммутатора для использования TACACS+-сервера

TACACS+ (англ. Terminal Access Controller Access Control System plus) — протокол управления доступом, результат усовершенствования протокола TACACS.

Шаг 1. Настройка сервера TACACS+

Для создания сервера TACACS+ используется команда:

```
(als_sw) #configure
(als_sw) (configure) #tacacs-server host 172.16.67.39
(als_sw) (configure) (Tacacs) #key qwerty
(als_sw) (configure) (Tacacs) #priority 32
```

В данном примере был создан TACACS+-сервер с адресом 172.16.67.39, секретный ключ для этого сервера был установлен в "qwerty", а приоритет установлен в 32. Приоритет TACACS+-сервера может принимать значения от 0 до 65535, при этом сервер с наименьшим значением приоритета считается наиболее приоритетным. Серверов TACACS+ может быть указано несколько (до 5), в этом случае коммутатор при проверке логина и пароля будет обращаться в порядке приоритета к каждому серверу TACACS+, пока не получит ответ. В случае получения разрешения или запрета аутентификации от сервера, остальные серверы опрашиваться не будут.

Шаг 2. Включение логирования CLI-команд (Spy Log) на сервер TACACS+ (опционально)

На коммутаторе есть дополнительная функция, которая заключается в отправке введенных на коммутаторе CLI-команд на сервер TACACS+. Введенные команды отправляются в сообщениях TACACS+ Accounting. Кроме того, коммутатор отправляет сообщения о входе и выходе пользователей.

Для включения отправки таких сообщения используется команда:

```
(als_sw) (configure) (Tacacs) #accounting
```

Команда вводится в контексте настройки конкретного сервера TACACS+. После выполнения команды сообщения начинают передаваться на TACACS+-сервер.

Шаг 3. Включение проверки логина и пароля пользователей с помощью сервера TACACS+

Для проверки логина и пароля пользователей на сервере TACACS+ необходимо добавить метод проверки в определенный список доступа:

```
(als_sw) #configure
(als_sw) (configure) #aaa authentication login networkList tacacs
```

В данном случае мы для списка доступа с именем **networkList** (по умолчанию он используется для доступа по Telnet и SSH) применили метод проверки "tacacs". Это значит, что при подключении пользователей по протоколу Telnet или SSH проверка логина и пароля будет проводится коммутатором на TACACS+-сервере.

Шаг 4. Включение проверки пароля команды повышения привилегий с помощью сервера TACACS+

Для проверки пароля команды повышения привилегий на сервере TACACS+ необходимо добавить метод проверки в определенный список доступа:

```
(als_sw) #configure
(als_sw) (configure) #aaa authentication enable enableList tacacs
```

В данном случае мы для списка доступа с именем **enableList** применили метод проверки "tacacs". Это значит, что при попытке повышения привилегий проверка пароля будет проводится коммутатором на TACACS+-сервере. Важно отметить, что для корректной проверки пароля команды **enable** на TACACS+-сервере должен быть настроен привилегированный (**priv-lvl = 15**) пользователь **$enab15$**, его пароль и будет паролем команды повышения привилегий.

3.3. Типовые вопросы и ошибки

Рассмотрим общую схему работы проверки логина и пароля пользователя.

Конфигурация коммутатора для данного примера:

```
network parms 172.17.1.1 255.255.0.0 0.0.0.0
network mgmt_vlan 1
configure
aaa authentication login "networkList" tacacs radius
tacacs-server host 172.17.40.52
key encrypted 9245...b7f5
exit
radius server host auth 172.17.35.87 name server1
radius server key auth 172.17.35.87 encrypted 20b1...f50e
radius server primary 172.17.35.87
radius server host auth 172.17.14.48 name server2
radius server key auth 172.17.14.48 encrypted 0944...a0f4
exit
```

Длинные строки шифрованных ключей опущены в примере. Обратите внимание, что для консольного входа (COM) список доступа оставлен по умолчанию **defaultList**, в нем указан один метод: "local". Эта настройка не отображается в конфигурации.

Схема проверки для данного примера:

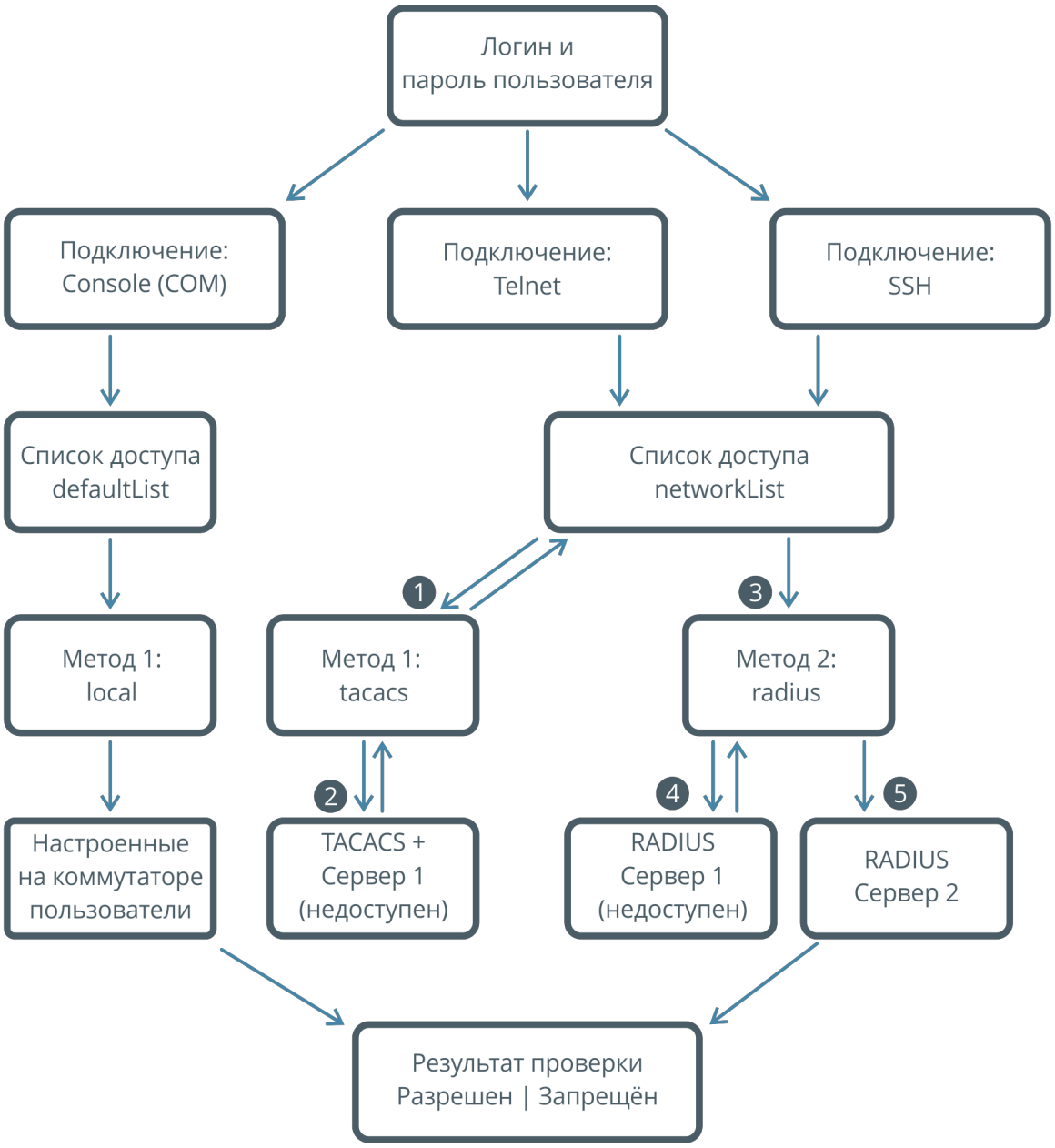

Рисунок 13. Общая схема прохождения проверки логина и пароля пользователя

На данной схеме изображен общий ход проверки. Проследим проверку логина и пароля на примере:

- пользователь входит по протоколу Telnet на коммутатор;
- для протокола Telnet назначен список доступа **networkList**, в котором указано два метода проверки: "tacacs" и "radius";
- первым будет опрошен сервер TACACS+ ("TACACS+ Сервер 1" на схеме, адрес 172.17.40.52 в конфигурации). В примере этот сервер недоступен, и проверка будет продолжена (однозначный результат не получен);
- вторым будет опрошен сервер RADIUS ("RADIUS Сервер 1" на схеме, адрес 172.17.35.87 в конфигурации). В примере этот сервер также недоступен, и проверка будет продолжена;
- третьим будет опрошен следующий сервер RADIUS ("RADIUS Сервер 2" на схеме, адрес 172.17.14.48 в конфигурации). Сервер доступен, и ответ, который он даст, будет определять успешность входа пользователя.

Рассмотрим еще один пример:

- пользователь подключается к коммутатору по COM-порту;
- для этого типа входа назначен список доступа **defaultList**, в котором всего один метод проверки — "local";
- введенный пользователем логин и пароль проверяется по локальной базе пользователей на коммутаторе;
- если пользователь найден в конфигурации коммутатора, и пользователь ввел верный пароль — доступ будет разрешен. В противном случае доступ будет запрещен.

ГЛАВА 4. УПРАВЛЕНИЕ ИНТЕРФЕЙСАМИ

4.1. Настройка

Настройка интерфейсов осуществляется в отдельном контексте (подробнее о контекстах читайте в главе "Концепции конфигурирования"). Любые настройки интерфейсов можно выполнить как для одного, так и для нескольких интерфейсов.

Для настройки одного интерфейса используется следующий вход в контекст:

```
(als_sw) #configure
(als_sw) (configure) #interface <slot>/<port>
(als_sw) (configure) (interface <slot>/<port>) #
```

- <slot> — физическое устройство (0) или логическое в наборе LAG (1);
- <port> — номер интерфейса.

Для настройки нескольких интерфейсов используется запятая (,) для перечисления интерфейсов и дефис (-) для указания диапазона интерфейсов. Возможно использовать комбинированный вариант, как показано ниже:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1,0/4,0/6-0/8,0/10
(als_sw) (configure) (interface 0/1,0/4,0/6-0/8,0/10) #
```

Во всех последующих примерах команд используется первый интерфейс. Выполнение команд в контексте нескольких интерфейсов идентично выполнению команд на каждом интерфейсе по отдельности и служит для удобства настройки.

Выключение и включение

Интерфейс имеет 2 статуса:

- административный — настроенный в конфигурации (Enabled/Disabled);
- оперативный — фактическое состояние линка (Up/Down).

Если интерфейс отключен административно, линк всегда будет в состоянии Down.

Выключение интерфейса

```
(als_sw) (configure) (interface 0/1) #shutdown
```

Включение интерфейса

```
(als_sw) (configure) (interface 0/1) #no shutdown
```

Управление параметрами передачи

Auto Negotiation — это процесс автоматического определения параметров передачи, таких как скорость и режим дуплекса. По умолчанию Auto Negotiation включен на всех интерфейсах коммутатора.

Шаг 1. Выключение Auto Negotiation

```
(als_sw) (configure) (interface 0/1) #no auto-negotiate
```

Шаг 2. Настройка скорости и режима дуплекса

```
(als_sw) (configure) (interface 0/1) #speed <speed> (half-duplex|full-duplex)
```

Значения параметра <speed> могут быть 10, 100, 1000 (Мбит/с). Если физический интерфейс не поддерживает указанную скорость, при вводе команды будет выведена ошибка. Аналогичная ситуация может возникнуть при попытке установить режим **1000 half-duplex**, поскольку полудуплексный режим не поддерживается на скорости 1000 Мбит/с.

Включение Auto Negotiation

```
(als_sw) (configure) (interface 0/1) #auto-negotiate
```

При включении Auto Negotiation настройки скорости и режима дуплекса на интерфейсе сбрасываются.

Настройка максимального размера кадра на интерфейсе

Параметр Maximum Frame Size определяет максимальный размер кадра Ethernet в октетах, который может быть передан интерфейсом. Значение Maximum Frame Size по умолчанию (оно же минимальное) — 1518 октетов. Максимальное значение — 9216 октетов. Увеличение Maximum Frame Size может быть необходимо для передачи больших кадров по технологии Jumbo Frame.

Настройка максимального размера кадра

Для назначения максимального размера кадра используется команда:

```
(als_sw) (configure) (interface 0/1) #max-frame-size 2000
```

Установка максимального размера кадра по умолчанию

Для возвращения максимального размера кадра на интерфейсе в значение по умолчанию используется команда:

```
(als_sw) (configure) (interface 0/1) #no max-frame-size
```

Настройка описания интерфейсов

Каждому интерфейсу можно назначить описание, которое может содержать дополнительную информацию об интерфейсе.

Для добавления описания используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #description "Customer 068834"
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #description "Uplink to aggregation"
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
```

Описание интерфейса может быть длиной до 64 символов. Разрешено использовать любые символы, кроме кавычек. Если в строке описания имеются символы, отличные от латинских букв и цифр, необходимо вводить строку описания в кавычках.

Просмотр описания интерфейсов

Для просмотра описания по всем интерфейсам используется команда:

```
(als_sw) #show port description all

  Interface  Status        Link  Description
  ---------  ------------  ----  -------------------
  0/1        Enable        Up    Customer 068834
  0/2        Disable       Down
  0/3        Disable       Down
  0/4        Disable       Down
  0/5        Disable       Down
  0/6        Disable       Down
  0/7        Disable       Down
  0/8        Disable       Down
  0/9        Enable        Down
  0/10       Enable        Up
  0/11       Enable        Down
  0/12       Enable        Up
  0/13       Disable       Down
  0/14       Disable       Down
  0/15       Disable       Down
  0/16       Disable       Down
  0/17       Disable       Down
  0/18       Disable       Down
  0/19       Disable       Down
  0/20       Disable       Down
  0/21       Disable       Down
  0/22       Enable        Down
  0/23       Enable        Down
  0/24       Enable        Down
  0/25       Disable       Down
  0/26       Disable       Down
  0/27       Enable        Down
  0/28       Enable        Up    Uplink to aggregation
```

Для просмотра описания по конкретному интерфейсу необходимо выполнить команду:

```
(als_sw) #show port description 0/1

  Interface  Status        Link  Description
  ---------  ------------  ----  -------------------
  0/1        Enable        Up    Customer 068834
```

Удалить описание интерфейса можно командой:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #no description
```

Настройка медных SFP-трансиверов

В случае применения медного SFP-трансивера может возникнуть необходимость настройки параметров линии (например, переключение трансивера в режим 100 Мбит/с). По умолчанию медные SFP-трансиверы работают в режиме 1000 Мбит/с.

Шаг 1. Настройка медного SFP-трансивера для работы в режиме 100 Мбит/с с поддержкой Auto-negotiation

Для настройки медного SFP-трансивера для работы на скорости 100 Мбит/с с поддержкой Auto-negotiation необходимо выполнить команду:

```
(als_sw) (configure) (interface 0/20) #sfp copper 100
```

Шаг 2. Настройка медного SFP-трансивера для работы в режиме 10/100/1000 Мбит/с с поддержкой Auto-negotiation

Для настройки медного SFP-трансивера для работы на скоростях 10/100/1000 Мбит/с с поддержкой Auto-negotiation необходимо выполнить команду:

```
(als_sw) (configure) (interface 0/20) #sfp copper 1000
```

Шаг 3. Настройка медного SFP-трансивера для работы в режиме 100FD без Auto-negotiation

Необходимо включить поддержку SFP-Copper, отключить режим Auto-negotiation и включить поддержку 100 Мбит/с Full-duplex:

```
(als_sw) (configure) (interface 0/20) #sfp copper enable
(als_sw) (configure) (interface 0/20) #no sfp copper auto-negotiate
(als_sw) (configure) (interface 0/20) #sfp copper speed 100 full-duplex
```

4.2. Мониторинг

Мониторинг интерфейсов осуществляется из корневого контекста.

Информация по всем интерфейсам

В зависимости от модификации коммутатора количество отображаемых интерфейсов может быть различным. Для примера ниже использовался коммутатор с восемнадцатью интерфейсами:

```
(als_sw) #show port all

  Interface  Status        Autoneg   Link  Speed           Uptime
  ---------  ------------  --------  ----  -------  --------------
  0/1        Enable        Enable    Up    100FD     0d, 00:06:05
  0/2        Enable        Enable    Down  None
  0/3        Enable        Enable    Down  None
  0/4        Enable        Enable    Down  None
  0/5        Enable        Enable    Down  None
  0/6        Enable        Enable    Down  None
  0/7        Enable        Enable    Down  None
  0/8        Enable        Enable    Down  None
  0/9        Enable        Enable    Down  None
  0/10       Enable        Enable    Down  None
  0/11       Enable        Enable    Down  None
  0/12       Enable        Enable    Down  None
  0/13       Enable        Enable    Down  None
  0/14       Enable        Enable    Down  None
  0/15       Enable        Enable    Down  None
  0/16       Enable        Enable    Down  None
  0/17       Enable        Enable    Down  None
  0/18       Enable        Enable    Down  None
```

Последними в списке интерфейсов идут uplink-интерфейсы коммутатора.

Поля таблицы:

- Interface — номер интерфейса;
- Status — показывает, включен или выключен интерфейс в конфигурации;
- Autoneg — показывает, включен или выключен режим Auto Negotiation. Возможные значения: Enable — включен, Disable — выключен;
- Link — показывает наличие или отсутствие соединения. Возможные значения: Up — соединение установлено, Down — соединение не установлено;
- Speed — скорость и режим дуплекса установленного соединения. Возможные значения: 10HD, 10FD, 100HD, 100FD, 1000FD;
- Uptime — время работы интерфейса.

В некоторых случаях интерфейс может быть заблокирован различными службами. В таком случае в поле "Status" будет отображаться причина блокировки:

- LBD Disabled — служба LBD обнаружила петлю за данным интерфейсом;
- UDL Disabled — служба LBDUD обнаружила однонаправленное соединение на данном интерфейсе;
- DAI Disabled — служба DAI (Dynamic ARP Inspection) обнаружила превышение разрешенного количества ARP-пакетов на данном интерфейсе;
- LAG Disabled — интерфейс является членом LAG, который был административно отключен.

Информация об интерфейсе

Посмотреть значения счетчиков пакетов, время работы интерфейса и время последнего сброса счетчиков можно командой:

```
(als_sw) #show interface 0/1

Interface 0/1
Packets Received Without Errors.........  55354
Packets Received With Error.............  0
Broadcast Packets Received..............  0
Packets Transmitted Without Errors......  21015564
Transmit Packet Errors..................  0
Collision Frames........................  0
Time Since Counters Last Cleared........  1 days 19 hours 21 minutes 35 seconds

Uptime..................................  0 days 0 hours 0 minutes 0 seconds
Last Link Status Change.................  07:11:41 ((UTC+0:00)) May 14 2015
```

Счетчики пакетов

Подробную статистику по интерфейсу можно посмотреть командой:

```
(als_sw) #show interface ethernet 0/1

Interface 0/1
Total Packets Received (Octets).........  5572979
Total Packets Transmitted (Octets)......  1347026250
Max Frame Size..........................  1518

Packets RX and TX 64 Octets.............  20963662
Packets RX and TX 65-127 Octets.........  107254
Packets RX and TX 128-255 Octets........  2
Packets RX and TX 256-511 Octets........  0
Packets RX and TX 512-1023 Octets.......  0
Packets RX and TX 1024-1518 Octets......  0

Total Packets Received Without Errors...  55354
Unicast Packets Received................  55352
Multicast Packets Received..............  2
Broadcast Packets Received..............  0

Total Packets Received with MAC Errors..  0
Jabbers Received........................  0
Fragments Received......................  0
Undersize Received......................  0
Alignment Errors........................  0
FCS Errors..............................  0
Overruns................................  0

Total Packets Transmitted Successfully..  21015564
Unicast Packets Transmitted.............  55351
Multicast Packets Transmitted...........  20960212
Broadcast Packets Transmitted...........  1

Total Transmit Errors...................  0
FCS Errors..............................  0
Underrun Errors.........................  0

Rx Oversized............................  0
Tx Oversized............................  0

Total Transmit Packets Discarded........  0
Single Collision Frames.................  0
Multiple Collision Frames...............  0
Excessive Collision Frames..............  0
Port Membership Discards................  0

Time Since Counters Last Cleared........  1 days 19 hours 21 minutes 53 seconds
```

Диагностика кабеля

На коммутаторах АЛСиТЕК предусмотрена диагностика кабеля. Обратите внимание, что выполнение команды может занять некоторое время. Количество анализируемых каналов зависит от типа интерфейса (FE или GE). В примере ниже приведен вывод команды диагностики кабеля, подключенного к FE интерфейсу 0/1, длиной 20 метров, с другой стороны которого нет активного оборудования:

```
(als_sw) #cablestatus 0/1

This operation may take one minute, please wait...

  Channel 1..................................... Open, 20.0m
  Channel 2..................................... Open, 20.0m
```

В примере ниже приведен вывод команды диагностики работоспособного кабеля, подключенного к GE интерфейсу 0/1, длиной 20 метров, с другой стороны которого подключено активное оборудование:

```
(als_sw) #cablestatus 0/1

This operation may take one minute, please wait...

  Channel 1..................................... Normal, 20.0m
  Channel 2..................................... Normal, 20.0m
  Channel 3..................................... Normal, 20.0m
  Channel 4..................................... Normal, 20.0m
```

В примере ниже приведен вывод команды диагностики кабеля с коротким замыканием на 3 канале:

```
(als_sw) #cablestatus 0/1

This operation may take one minute, please wait...

  Channel 1..................................... Open, 51.0m
  Channel 2..................................... Open, 51.0m
  Channel 3..................................... Short, 28.0m
  Channel 4..................................... Open, 51.0m
```

Механизм диагностики обнаружил короткое замыкание 3 канала на длине 28 метров.

Опрос SFP модулей

На коммутаторах АЛСиТЕК есть возможность получить информацию с модуля SFP, установленного в коммутатор. В зависимости от типа SFP-модуля и наличия у модуля поддержки мониторинга (DDM), листинги информации о SFP-модулях отличаются.

Вывод команды для SFP-модуля без поддержки DDM:

```
(als_sw) #show box sfp information 0/10

General Information:
  Vendor........................................ ATOP
  Part.......................................... AP-B35121-3CL20
  Serial........................................ SG35213500988
  Revision.....................................
  Date.......................................... 08-09-11

Additional Information:
  Device Type................................... SFP
  Calibration................................... Unknown
  DDM Support................................... No
```

Вывод команды для SFP-модуля с поддержкой DDM:

```
(als_sw) #show box sfp information 0/25

General Information:
  Vendor..................................... NEOPHOTONICS
  Part....................................... PTD50615C18+
  Serial..................................... A0711609305
  Revision................................... 1.0
  Date....................................... 04-07-11

Additional Information:
  Device Type................................ Unknown
  Calibration................................ Internal
  DDM Support................................ Yes (SFF-8472)

  Severity    Temperature  Vcc       Bias      TX Power   RX Power
  ----------  -----------  --------  --------  ---------  ---------
  Warnings    Good         Good      Good      Good       Good
  Alarms      Good         Good      Good      Good       Good

  Parameter         Current   High Alarm  High Warning  Low Warning  Low Alarm

  ----------------  --------  ----------  ------------  -----------  ---------
  Temperature (C)   60.0000   125.0000    120.0000      -5.0000      -10.0000
  Voltage (V)       3.3098    3.5000      3.4500        3.1800       3.1300
  Bias (mA)         25.5380   70.0000     60.0000       0.0000       0.0000
  TX Power (dBm)    4.7462    8.0000      7.0000        4.0000       3.0001
  RX Power (dBm)    -15.2433  -8.9997     -11.9997      -24.9485     -27.9588
```

ГЛАВА 5. АГРЕГАЦИЯ КАНАЛОВ

5.1. Введение в агрегацию каналов

Агрегация каналов предназначена для объединения нескольких физических каналов в один логический. Может быть использована для:

- увеличения пропускной способности;
- резервирования физических каналов.

Увеличение пропускной способности достигается за счет распределения пакетов между физическими интерфейсами в агрегированном канале. Обычно распределение происходит по полям передаваемого пакета, например по MAC-адресу источника или назначения. Также правило распределения может формироваться на основе нескольких полей пакета, а также дополнительной информации, такой как номер интерфейса, с которого пришел пакет.

Резервирование достигается за счет того, что пакеты распределяются только на те интерфейсы, у которых установлена связь с удаленной стороной. В простейшем случае связь с удаленной стороной определяется по наличию линка. В более сложном случае используются специальные протоколы, которые устанавливают связь с удаленной стороной путем обмена пакетами. К таким протоколам относится LACP (Link Aggregation Control Protocol) — протокол контроля агрегированных интерфейсов.

5.2. Настройка агрегации каналов на коммутаторах АЛСиТЕК

Общие принципы конфигурирования

В коммутаторах АЛСиТЕК возможно создать до 8 логических интерфейсов, в каждый из которых возможно добавить до 8 физических интерфейсов для агрегации. Если на одном из физических интерфейсов пропадает линк, то такой интерфейс будет исключен из распределения пакетов, и включен в распределение только после восстановления линка. Более подробно данный механизм описан в стандарте IEEE 802.1ax.

В коммутаторах АЛСиТЕК поддерживается 6 алгоритмов балансировки трафика. Алгоритм выбирается в зависимости от архитектуры сети.

Настройка агрегации

Рассмотрим настройку агрегации на коммутаторе 1 и коммутаторе 2, соединяющих "L2 сеть 1" и "L2 сеть 2". Коммутаторы 1 и 2 подключены друг к другу через интерфейсы 0/27, 0/28, которые следует объединить в логический интерфейс 1/1. Далее рассмотрим конфигурацию коммутатора 1 по шагам, конфигурация коммутатора 2 будет аналогичной.

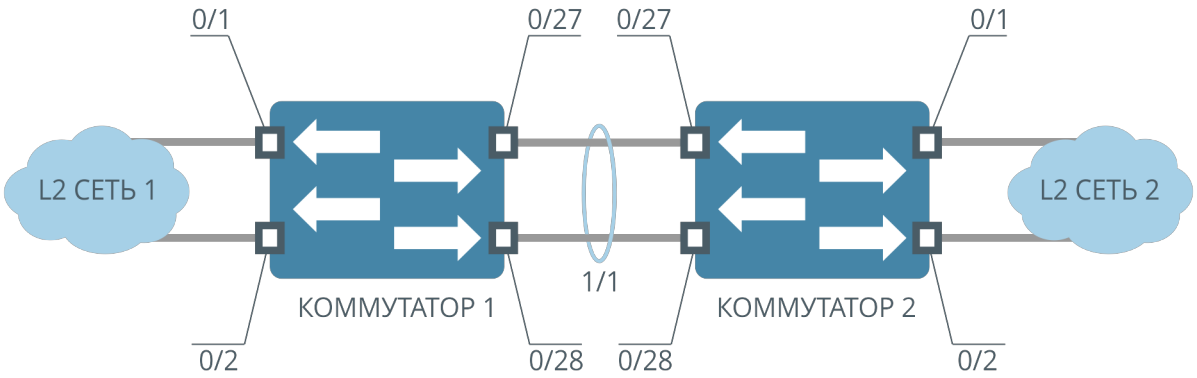

Рисунок 14. Пример агрегации каналов для двух L2-сетей

Шаг 1. Создание агрегированного интерфейса

Создаем логический интерфейс с именем "bound0":

```
(als_sw) #configure
(als_sw) (configure) #port-channel "bound0"
(als_sw) (configure) #exit
```

Созданному интерфейсу динамически назначается номер из диапазона 1/1-1/8. Узнать, какой номер присвоен интерфейсу, можно с помощью команды:

```
(als_sw) #show port-channel brief

Interface  Name             Link  Trap Flag  Type     Active Ports
---------  ---------------  ----  ---------  -------  ------------
1/1        bound0           Down  Enabled    Dynamic
```

Каналу "bound0" присвоен номер интерфейса 1/1.

Шаг 2. Добавление физических интерфейсов в логический

Добавим интерфейсы 0/27 и 0/28 к интерфейсу "bound0" 1/1:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/27-0/28
(als_sw) (configure) (interface 0/27-0/28) #addport 1/1
(als_sw) (configure) (interface 0/27-0/28) #exit
(als_sw) (configure) #exit
```

Шаг 3. Настройка статической агрегации (опционально)

По умолчанию LAG-интерфейс работает под управлением протокола LACP. В некоторых ситуациях может быть необходимо вручную управлять агрегированным каналом. В этой ситуации возможно преобразовать LAG-интерфейс в статический.

Указываем, что "bound0" должен быть статическим интерфейсом:

```
(als_sw) #configure
(als_sw) (configure) #interface 1/1
(als_sw) (configure) (interface 1/1) #port-channel static
(als_sw) (configure) (interface 1/1) #exit
(als_sw) (configure) #exit
```

Шаг 4. Настройка алгоритма балансировки трафика

Установим алгоритм балансировки трафика для логического интерфейса "bound0" 1/1. Трафик будет балансироваться по MAC-адресу источника, VLAN, Ethertype и номеру входящего физического интерфейса. Все алгоритмы балансировки трафика приведены в таблице. По умолчанию используется алгоритм 3.

Номер алгоритма	Поля

1	Src MAC, VLAN, Ethertype, Interface
2	Dst MAC, VLAN, Ethertype, Interface
3	Src/Dst MAC, VLAN, Ethertype, Interface
4	Src IP, TCP/UDP L4 Port
5	Dst IP, TCP/UDP L4 Port
6	Src/Dst IP, TCP/UDP L4 Port

Выбранный алгоритм балансировки имеет номер 1. Для его установки служит команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 1/1
(als_sw) (configure) (interface 1/1) #port-channel load-balance 1
(als_sw) (configure) (interface 1/1) #exit
(als_sw) (configure) #exit
```

Шаг 5. Расширенная настройка балансировки трафика

При некоторых конфигурациях сети возможна ситуация, когда трафик плохо балансируется стандартными алгоритмами (например, хеш-функция всегда имеет четное значение). В этой ситуации дополнительно к выбору алгоритма балансировки трафика можно включить расширенный режим балансировки, использующий другие смещения исходных данных хеш-функции:

```
(als_sw) #configure
(als_sw) (configure) #interface 1/1
(als_sw) (configure) (interface 1/1) #port-channel load-balance advanced auto
(als_sw) (configure) (interface 1/1) #exit
(als_sw) (configure) #exit
```

Теперь обмен трафиком между сетями "L2 сеть 1" и "L2 сеть 2" будет проходить по двум физическим каналам, а суммарная пропускная способность может достигать суммарной пропускной способности всех физических интерфейсов в составе агрегированного канала. Важно понимать, что реальное прохождение трафика зависит от алгоритма балансировки.

ГЛАВА 6. ЗЕРКАЛИРОВАНИЕ

6.1. Введение в зеркалирование

Зеркалирование — дублирование трафика, проходящего через один или несколько интерфейсов, на интерфейс-получатель.

Зеркалирование интерфейсов может понадобиться в следующих ситуациях:

- подключение анализатора пакетов для поиска неисправностей в сети;
- подключение систем обнаружения вторжений для выявления вредоносного трафика.

6.2. Зеркалирование на коммутаторах АЛСиТЕК

При конфигурировании зеркалирования задается набор интерфейсов-источников (MIRRORED или source), трафик с которых будет копироваться на интерфейс-получатель (PROBE или destination). С интерфейсов-источников будет копироваться как входящий, так и исходящий трафик.

Набор интерфейсов-источников и интерфейс-получатель объединяются в сессию зеркалирования.

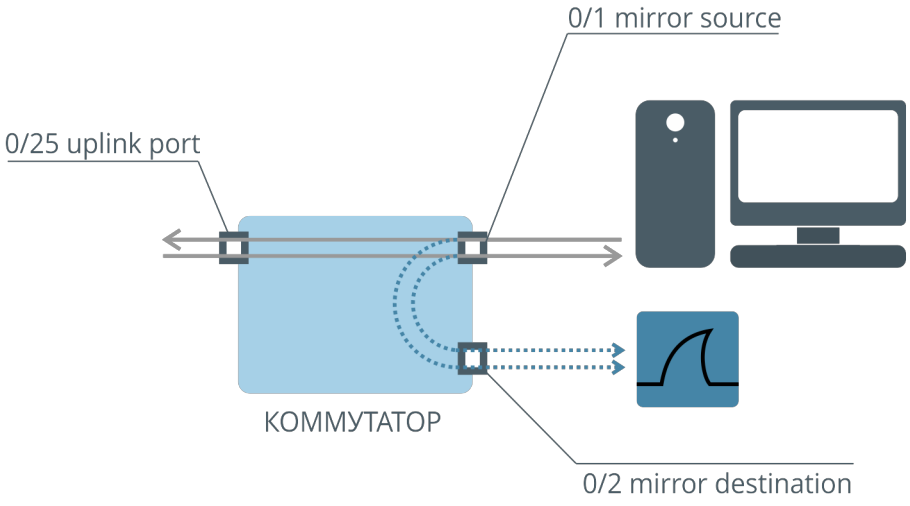

Рисунок 15. Механизм зеркалирования

Настройка зеркалирования

Шаг 1. Настраиваем интерфейсы-источники (MIRRORED)

Указываем один или несколько интерфейсов, входящий и исходящий трафик с которых будет зеркалироваться:

```
(als_sw) #configure
(als_sw) (configure) #monitor session 1 source interface 0/1
```

Интерфейсами-источниками в одной сессии зеркалирования могут выступать все интерфейсы коммутатора одновременно, кроме интерфейса-получателя.

Шаг 2. Настраиваем интерфейс-получатель (PROBE)

Указываем один интерфейс, на который трафик будет копироваться со всех интерфейсов-источников:

```
(als_sw) (configure) #monitor session 1 destination interface 0/2
```

В одной сессии зеркалирования может быть настроен только один интерфейс-получатель.

Шаг 3. Включение и выключение

Для включения сессии зеркалирования используется команда:

```
(als_sw) (configure) #monitor session 1 mode
```

После выполнения этой команды трафик (входящий и исходящий) с интерфейса 0/1 будет копироваться на интерфейс 0/2.

Для выключения зеркалирования необходимо выполнить команду:

```
(als_sw) (configure) #no monitor session 1 mode
```

6.3. Типовые вопросы и ошибки

В: Влияют ли настройки на destination интерфейсе на зеркалируемый трафик?
О: Да, влияют. На destination интерфейсе нежелательны другие настройки.

В: Как сменить интерфейс-получатель?
О: Для изменения интерфейса-получателя необходимо сначала удалить старый интерфейс-получатель, затем назначить новый.

Пример:

```
(als_sw) (configure) #no monitor session 1 destination
(als_sw) (configure) #monitor session 1 destination interface 0/11
```

В: Как сменить набор интерфейсов-источников на другой?
О: Для изменения набора интерфейсов-источников необходимо удалить интерфейсы из текущего набора и добавить новые.

Пример:

```
(als_sw) (configure) #no monitor session 1 source interface 0/1
(als_sw) (configure) #monitor session 1 source interface 0/7
```

В: Можно ли мониторить пакеты с нескольких интерфейсов-источников (MIRRORED)?
О: Да, просто укажите все необходимые интерфейсы.

Пример:

```
(als_sw) (configure) #monitor session 1 source interface 0/1-0/3,0/5
```

ГЛАВА 7. SNMP

7.1. Введение в SNMP

Simple Network Management Protocol (SNMP) — интернет-протокол для управления устройствами в IP-сетях.

Основные участники обмена по протоколу SNMP:

- SNMP-агент — ПО, запускаемое на управляемом устройстве;
- SNMP-менеджер — ПО, запущенное на контролирующем устройстве, например система сетевого управления (Network Management System, NMS).

SNMP-менеджер по протоколу SNMP может обращаться к SNMP-агентам и осуществлять мониторинг их состояния. Кроме того, протокол SNMP позволяет изменять настройки оборудования.

Вся информация о способе доступа к оборудованию собрана в так называемых MIB (Management Information Base), которые представляют собой текстовую запись дерева OID (Object IDentifier) — элементов, к которым возможно обращение по протоколу SNMP.

7.2. SNMP на коммутаторах АЛСиТЕК

SNMP-агент, работающий на коммутаторе, поддерживает версии SNMP 1, 2(c) и 3. Для доступа к SNMP-агенту коммутатора необходимо настроить SNMP-community для версий SNMP 1 и 2c. SNMP-community выступают в роли своеобразных паролей для доступа к агенту. SNMP-community в запросах SNMP к агенту проверяется при каждом запросе и в соответствии с ним выдаются права на выполнение определенных операций. Для доступа к SNMP-агенту коммутатора по версии SNMP 3 необходима настройка типов и ключей шифрования для одного или нескольких пользователей коммутатора, а также указание Engine Id, который будет использоваться для передачи защищенных данных между менеджером и агентом.

Различают два уровня доступа:

- Read Only — возможно получение информации от агента, изменение настроек запрещено;
- Read/Write — возможно получение информации от агента и любые изменения настроек.

Настройка SNMPv1/2c

Аутентификация менеджера SNMP на коммутаторе при использовании протоколов SNMPv1/2c осуществляется с помощью проверки строки SNMP-community. Для того, чтобы использовать определенные SNMP-community, нужно их создать в настройке коммутатора. По умолчанию на коммутаторах АЛСиТЕК присутствуют следующие SNMP-community, которые можно просмотреть командой:

```
(als_sw) #show snmpcommunity

SNMP Community Name   Client IP Address  Client IP Mask      Access Mode     St
atus
--------------------  -----------------  ------------------  --------------  --
------------
public                0.0.0.0            0.0.0.0             Read Only       En
able
private               0.0.0.0            0.0.0.0             Read/Write      En
able
```

SNMP-community с именем "public" предоставляет доступ для чтения, "private" — для чтения и изменения настроек.

Шаг 1. Создание SNMP-community на коммутаторе

Создадим SNMP-community с именем "monitoring" и правами только на чтение:

```
(als_sw) #configure
(als_sw) (configure) #snmp-server community ro "monitoring"
```

Добавим SNMP-community для настройки коммутатора по SNMP с правами чтения и записи:

```
(als_sw) #configure
(als_sw) (configure) #snmp-server community rw "nms"
```

Шаг 2. Ограничение IP-адресов, с которых разрешен доступ (опционально)

Вход по SNMP-community можно ограничить диапазоном разрешенных IP-адресов:

```
(als_sw) #configure
(als_sw) (configure) #snmp-server community ipaddr 172.17.1.100 "nms"
(als_sw) (configure) #snmp-server community ipmask 255.255.255.255 "nms"
(als_sw) (configure) #snmp-server community ipaddr 172.17.1.100 "monitoring"
(als_sw) (configure) #snmp-server community ipmask 255.255.0.0 "monitoring"
```

Ограничение входа не является обязательным, по умолчанию доступ разрешен с любых адресов. В примере вход по SNMP-community с именем "nms" разрешен только с IP-адреса 172.17.1.100, а вход по SNMP-community с именем "monitoring" ограничен диапазоном 172.17.0.0-172.17.255.255.

Шаг 3. Удаление существующих SNMP-community (опционально)

Для удаления SNMP-community используется команда:

```
(als_sw) #configure
(als_sw) (configure) #no snmp-server community "monitoring"
```

Для повышения безопасности есть возможность отключить стандартные SNMP-community "public" и "private", что поможет предотвратить несанкционированный доступ к коммутатору. Для их отключения используется та же команда, что и для удаления:

```
(als_sw) #configure
(als_sw) (configure) #no snmp-server community "public"
(als_sw) (configure) #no snmp-server community "private"
```

Удаление стандартных SNMP-community невозможно.

После выполнения этих команд таблица SNMP-community примет следующий вид:

```
(als_sw) #show snmpcommunity

SNMP Community Name   Client IP Address  Client IP Mask      Access Mode     St
atus
--------------------  -----------------  ------------------  --------------  --
------------
public                0.0.0.0            0.0.0.0             Read Only       Di
sable
private               0.0.0.0            0.0.0.0             Read/Write      Di
sable
nms                   172.17.1.100       255.255.255.255     Read/Write      En
able
```

Из таблицы видно, что SNMP-community по умолчанию не удалены, а заблокированы. Доступ по ним будет невозможен. При этом SNMP-community с именем "monitoring" удалено.

Настройка SNMPv3

SNMPv3 описан в RFC 3411 и RFC 3418 и в настоящий момент считается предпочтительным для использования при настройке сетевых устройств, в то время как версии SNMPv1/2c отмечены как исторические и небезопасные.

Шаг 1. Настройка пользователей

Аутентификация при использовании SNMPv3 осуществляется не по строкам SNMP-community, а по логину пользователя. В данном случае пользователем является обычный пользователь на коммутаторе. По умолчанию пользователи и настройки SNMPv3 для них выглядят следующим образом:

```
(als_sw) #show users

                            User          SNMPv3       SNMPv3          SNMPv3
  User Name                 Access Mode   Access Mode  Authentication  Encrypt
  ------------------------  ------------  -----------  --------------  -------
  admin                     Read/Write    Read/Write   None            None
  guest                     Read Only     Read Only    None            None
```

В этой таблице:

- "User Name" — имя пользователя, логин;
- "User Access Mode" — уровень доступа пользователя при входе в CLI (через консольный COM-порт, telnet или SSH);
- "SNMPv3 Access Mode" — уровень доступа при входе по SNMPv3. Может отличаться от уровня доступа в CLI;
- "SNMPv3 Authentication" — тип хеширования, который будет использоваться для проверки подлинности пароля пользователя при входе. Может принимать значения "none", "md5" и "sha";
- "SNMPv3 Encrypt" — тип шифрования, который будет использоваться при передаче данных между менеджером и агентом. Может принимать значения "none", "aes" и "des".

Пользователь "admin" имеет полный доступ к коммутатору по SNMPv3, при этом аутентификация пользователя проводится без хеширования пароля и передача данных проводится без шифрования. Пользователь "guest" также не использует хеширование пароля и шифрование, но обладает правами только на чтение.

Создание, изменение пароля, изменение уровня доступа CLI и удаление пользователей на коммутаторе можно осуществлять командами:

```
(als_sw) (configure) #username <name> password <password> level <level>
(als_sw) (configure) #no username <name>
```

Подробнее про работу с пользователями можно прочитать в главе "Аутентификация".

Шаг 2. Настройка уровня доступа

Изменить уровень доступа SNMPv3 для пользователя можно следующей командой:

```
(als_sw) (configure) #username snmpv3 accessmode <username> <readonly|readwrite
>
```

Назначим пользователю "admin" права только на чтение при входе по SNMPv3:

```
(als_sw) (configure) #username snmpv3 accessmode "admin" readonly
```

По умолчанию вновь создаваемые пользователи имеют права только на чтение по SNMPv3, вне зависимости от их уровня доступа в CLI.

Шаг 3. Настройка типа хеширования пароля аутентификации

Изменить тип хеширования пароля при аутентификации пользователя по SNMPv3 можно с помощью команды:

```
(als_sw) (configure) #username snmpv3 authentication <username> md5
(als_sw) (configure) #username snmpv3 authentication <username> sha
```

Обратите внимание, что для включения хеширования пароля у пользователя должен быть задан пароль длиной не менее 8 символов.

По умолчанию пароль передается в открытом виде, хеширование отключено:

```
(als_sw) (configure) #username snmpv3 authentication "admin" none
```

Зададим хеширование MD5 пароля аутентификации для пользователя "admin":

```
(als_sw) (configure) #username "admin" password "qwerty110" level 15
(als_sw) (configure) #username snmpv3 authentication "admin" md5
```

Если для пользователя не указать тип хеширования пароля, то доступ по SNMPv3 будет возможен без пароля, даже если у пользователя задан пароль в CLI.

Шаг 4. Настройка типа шифрования передаваемых данных

Изменить тип шифрования передаваемых по SNMPv3 данных можно командами:

```
(als_sw) (configure) #username snmpv3 encryption <username> aes <key>
(als_sw) (configure) #username snmpv3 encryption <username> des <key>
```

Обратите внимание на то, что для включения шифрования у пользователя должен быть задан тип хеширования пароля (MD5 или SHA).

По умолчанию шифрование отключено:

```
(als_sw) (configure) #username snmpv3 encryption <username> none
```

Изменим шифрование для пользователя "admin" на AES с ключом "25SDg5e76dXg":

```
(als_sw) (configure) #username snmpv3 encryption "admin" aes "25SDg5e76dXg"
```

Ключи шифрования в конфигурации коммутатора выводятся в зашифрованном виде.

7.3. SNMP-trap на коммутаторах АЛСиТЕК

При возникновении определенных событий (например, при изменении статуса соединения) коммутатор может отправить сообщение (SNMP-trap) зарегистрированным приемникам SNMP-trap. Регистрация включает задание IP-адреса, SNMP-community для доступа к приемнику и версию SNMP-trap. IP-адрес приемника должен быть доступен с коммутатора, то есть должен находиться в одной с ним подсети, либо быть доступен через шлюз, настроенный на коммутаторе.

Настройка SNMP-trap v1/2

Для отправки SNMP-trap необходимо зарегистрировать один или несколько приемников SNMP-trap. Максимальное количество приемников равно 16. При возникновении события SNMP-trap будет отправлен всем настроенным приемникам.

```
(als_sw) #configure
(als_sw) (configure) #snmptrap "monitor" ipaddr 172.17.1.7 snmpversion snmpv2
(als_sw) (configure) #snmptrap "test" ipaddr 172.17.1.8 snmpversion snmpv1
```

Без указания версии SNMP-trap будут отправляться с версией SNMPv2. После выполнения этих команд все SNMP-trap будут отправляться двум приемникам.

Настройка SNMP-trap v3

Для отправки SNMP-trap по протоколу SNMPv3 необходимы дополнительные настройки. Поскольку протокол SNMPv3 может использовать хеширование при аутентификации и шифрование при передаче данных, необходимо настроить их на коммутаторе для аутентификации на приемнике SNMP-trap. При отправке SNMP-trap есть особенность, они отправляются без установления соединения и ожидания подтверждения, поэтому агент и менеджер не смогут обменяться Engine Id. Он должен быть настроен на коммутаторе.

Шаг 1. Настройка Engine Id

Для настройки Engine Id используется следующая команда:

```
(als_sw) (configure) #snmp-server engineid <engine-id>
```

Параметры:

- <engine-id> — HEX-строка четной длины. Допустимые символы: [a-f0-9].

Пример:

```
(als_sw) (configure) #snmp-server engineid 0987654321
```

Шаг 2. Создание пользователя для отправки SNMP-trap

Для отправки SNMP-trap по протоколу SNMPv3 необходимо создать пользователя, под которым будет проходить аутентификация на приемнике SNMP-trap. Пользователи необходимы только для отправки SNMP-trap и не влияют на обычных пользователей коммутатора. Создать пользователя для отправки SNMP-trap можно командой:

```
(als_sw) (configure) #snmptrap snmpv3 <trap-user>
```

Параметры:

- <trap-user> — имя пользователя для отправки SNMP-trap. Разрешены латинские буквы, цифры и знак "_".

Пример:

```
(als_sw) (configure) #snmptrap snmpv3 "notificator"
```

Шаг 3. Настройка аутентификации

После создания пользователя ему нужно назначить необходимые тип хеширования пароля и сам пароль длиной не менее 8 символов с помощью команды:

```
(als_sw) (configure) #snmptrap snmpv3 authentication <trap-user> none
(als_sw) (configure) #snmptrap snmpv3 authentication <trap-user> md5 <password>

(als_sw) (configure) #snmptrap snmpv3 authentication <trap-user> sha <password>
```

Пример:

```
(als_sw) (configure) #snmptrap snmpv3 authentication "notificator" md5 "notific
ator_password_df5h"
```

Шаг 4. Настройка шифрования

Назначить тип и ключ шифрования для передачи данных SNMP-trap можно командой:

```
(als_sw) (configure) #snmptrap snmpv3 encryption <trap-user> none
(als_sw) (configure) #snmptrap snmpv3 encryption <trap-user> aes <key>
(als_sw) (configure) #snmptrap snmpv3 encryption <trap-user> des <key>
```

Ключ шифрования должен быть длиной от 8 до 64 символов.

Пример:

```
(als_sw) (configure) #snmptrap snmpv3 encryption "notificator" aes "enc_key_not
ificator_6rs31be"
```

Шаг 5. Настройка адреса приемника SNMP-trap

После настройки пользователя для отправки SNMP-trap сообщений версии SNMPv3 необходимо указать адрес приемника SNMP-trap с помощью команды:

```
(als_sw) (configure) #snmptrap snmpv3 ipaddr <trap-user> <ip>
```

Параметры:

- <trap-user> — имя пользователя для отправки SNMP-trap, который будет использоваться для отправки SNMP-trap;
- <ip> — адрес, куда будут отправляться SNMP-trap.

Пример:

```
(als_sw) (configure) #snmptrap snmpv3 ipaddr "notificator" 172.17.13.78
```

Блокировка отправки SNMP-trap

Блокировка отправки SNMP-trap при изменении оперативного состояния интерфейсов:

```
(als_sw) #configure
(als_sw) (configure) #no snmp trap link-status all
```

Просмотр

Просмотреть список приемников и версии отправляемых SNMP-trap можно командой (для SNMPv1/2c):

```
(als_sw) #show snmptrap

SNMP Trap Name       IP Address          SNMP Version
-------------------  ------------------  --------------
monitor              172.17.1.7          snmpv2
test                 172.17.1.8          snmpv1
```

Для версии SNMPv3 просмотреть список приемников SNMP-trap, пользователей, типов хеширования и шифрования можно командой:

```
(als_sw) #show snmptrap snmpv3
                                          SNMPv3          SNMPv3
  User Name                IP Address     Authentication  Encryption
  -----------------------  -------------- --------------  ----------
  notificator              172.17.13.78   MD5             AES
```

ГЛАВА 8. ЛОГИРОВАНИЕ

8.1. Введение в логирование

Логирование — это сохранение текстовых сообщений о событиях, происходящих на устройстве, в памяти устройства. Такими событиями могут быть изменения состояния физических портов, события изменения показаний датчиков и события отказа аппаратуры.

Благодаря логированию можно изучить изменение состояния устройства за некоторый промежуток времени и определить возможные причины выхода из строя или неправильной работы устройства.

Лог бывает локальный и удаленный (обычно Syslog). Локальный лог хранится в памяти устройства, удаленный на Syslog-сервере. Syslog-сервер можно настроить так, что сообщения будут сортироваться и записываться в разные файлы по приоритету. Например, сообщения ядра часто направляются в отдельный файл, так как эти сообщения наиболее важные и должны регулярно просматриваться во избежание серьезных проблем.

8.2. Логирование на коммутаторах АЛСиТЕК

- оперативный лог (buffered logging) — сохраняется в оперативной памяти устройства и хранит сообщения от старта устройства до момента перезагрузки. При перезагрузке оперативный лог уничтожается;
- перманентный лог (permanent/persistent logging) — сохраняется в долговременной памяти устройства (на flash), хранит сообщения с момента последней очистки перманентного лога. Не очищается при перезагрузке или очистке конфигурации. Очищается только при выполнении специальной команды;
- логирование введеных команд (logging cli-command) — вид логирования, который включает запись введеных команд в оперативный и перманентный (если он включен) лог;
- логирование в консоль управления (console logging) — специальный вид логирования, когда сообщения выводятся непосредственно в CLI. Вывод в CLI работает только в рамках сессии управления. Когда сессия управления завершается (по команде, или по таймауту) — вывод прекращается;
- логирование отправленных SNMP-trap сообщений — все отправленные коммутаторов SNMP-trap сообщения заносятся в специальный лог,

который можно просмотреть отдельной командой;
- логирование на Syslog-сервер (logging syslog) — отправка сообщений лога на настроенный Syslog-сервер.

Оперативный и перманентный лог работают по принципу кольцевого буфера. Вновь приходящие сообщения помещаются в конец буфера до тех пор пока он не накопит максимальное количество сообщений. При максимальном значении размера кольцевого буфера сообщения продолжают записываться в журнал, но при этом сообщения в начале кольцевого буфера удаляются.

Размер оперативного и перманентного лога выбирается в соответствии с объемом памяти конкретного устройства, на АЛС-24110LVT размер оперативного лога — 20 тысяч сообщений, перманентного — 4 тысячи сообщений.

По умолчанию на устройстве включен оперативный лог (buffered logging).

8.3. Работа с логированием

Оперативный лог

Настройка

Оперативный лог (buffered logging) включен по умолчанию, дополнительных настроек не требуется.

Просмотр

Для просмотра оперативного лога выполните команду:

```
(als_sw) #show logging buffered [<count>]
```

Здесь [<count>] — количество выводимых сообщений (необязательный параметр). К примеру, значение 10 выведет последние 10 сообщений из оперативного лога. Если количество не указано, будут выведены все сообщения оперативного лога.

Перманентный лог

Настройка

Включение перманентного лога:

```
(als_sw)(configure) #logging persistent <level>
```

Здесь <level> — уровень логирования. Уровень логирования может быть задан числом от 0 до 7 или строкой из следующего списка: emergency, alert, critical, error, warning, notice, info, debug. Значение 0 соответствует уровню логирования emergency, значение 1 — уровню alert и так далее. Уровень по умолчанию: warning (числовое значение 4).

Каждое из сообщений лога имеет определенный уровень важности от 0 (самые важные, emergency) до 7 (отладочные сообщения, debug). Уровень логирования позволяет отбросить неважные сообщения, и сохранять только те, уровень которых численно меньше или равен установленному. Пример: при настройке уровня логирования в 4 (warning) в перманентный лог попадут все сообщения с уровнями от 0 до 4 включительно. Сообщения с уровнем с 5 по 7 будут отброшены и не попадут в перманентный лог.

Чтобы изменить уровень перманентного лога, выполните повторно команду:

```
(als_sw)(configure) #logging persistent <level>
```

Отключение перманентного лога осуществляется командой:

```
(als_sw)(configure) #no logging permanent
```

Просмотр

Просмотр перманентного лога возможен командой:

```
(als_sw)(configure) #show logging permanent [<count>]
```

Здесь [<count>] — количество выводимых сообщений (необязательный параметр). Если значение не указано, будет выведет весь перманентный лог.

Логирование введенных команд

Настройка

Включение логирования введеных команд:

```
(als_sw)(configure) #logging cli-command
```

Отключение:

```
(als_sw)(configure) #no logging cli-command
```

Просмотр

При включении логирования введенных команд команды записываются и в оперативный лог, и в перманентный, если он был включен. Команды CLI не будут записаны в перманентный лог, если его уровень ниже уровня сообщений CLI-команд — 6.

Очистка логов

Очистка всех локальных логов производится командой:

```
(als_sw) (configure) #logging clear
```

Оперативный лог (buffered logging) также очищается при перезагрузке устройства.

Логирование в консоль управления

Настройка

При включении логирования в CLI сообщения выводятся непосредственно в CLI-сессию управления. При этом на другие сессии данная команда не влияет.

Включение логирования в CLI:

```
(als_sw) (configure) #logging console 7
```

Отключение:

```
(als_sw) (configure) #no logging console
```

Логирование в CLI также завершается при завершении сессии. Данная команда не отображается в конфигурации.

Просмотр списка SNMP-trap сообщений

Коммутатор при возникновении некоторых событий может сообщать о них всем настроенным приемникам SNMP-trap сообщений. Каждое сообщение будет занесено в отдельный лог, для просмотра которого используется команда:

```
(als_sw) #show logging traplogs

  Log  System Up Time      Trap
  ---  ------------------  ------------------------------------------------
    1  0 days 00:01:24     Port: Link Up: 0/18
    2  0 days 00:01:24     Port: Link Up: 0/17
    3  0 days 00:01:24     Init: Cold Start
```

В этом логе сохраняются все SNMP-trap сообщения. Запись ведется по принципу кольцевого буфера, максимальный размер которого составляет 256 сообщений. При перезагрузке коммутатора лог очищается.

У данного лога есть несколько особенностей:

- при выводе лога более новые сообщения отображаются в начале;
- сообщения SNMP-trap, отправка которых отключена, не будут занесены в лог SNMP-trap сообщений;
- в лог попадают все SNMP-trap сообщения, вне зависимости от того, настроены ли приемники SNMP-trap сообщений на коммутаторе или нет.

8.4. Syslog

Пошаговая настройка

Шаг 1. Настройка адреса Syslog-сервера

Для назначения адреса Syslog-сервера используется команда:

```
(als_sw) #configure
(als_sw) (configure) #logging host <ipaddress> ipv4 [<port>] [<severitylevel>]
```

Параметры:

- <ipaddress> — IP-адрес Syslog сервера;
- [<port>] — необязательный параметр, порт назначения на Syslog-сервере;
- [<severitylevel>] — необязательный параметр, уровень логирования.

Уровень логирования для Syslog-сервера работает по тому же принципу, что и для перманентного лога. При установке значения 4 (warning) на Syslog-сервер будут отправлены сообщения с уровнями от 0 до 4 включительно.

Шаг 2. Включение отправки сообщений на Syslog-сервер

Для включения отправки используется команда:

```
(als_sw) (configure) #logging syslog
```

После выполнения этой команды сообщения начнут отправляться на указанный Syslog-сервер.

8.5. Типовые вопросы и ошибки

В: Может ли перманентный лог заполнить память устройства и привести к выходу оборудования из строя?
О: Нет, благодаря принципу кольцевого буфера максимальный размер перманентного лога известен заранее и не будет увеличиваться.

В: Уничтожается ли перманентный лог при его выключении?
О: Нет, при выключении перманентный лог просто перестает записываться.

В: Какое время попадает в сообщения лога?
О: Системное время устройства. Если настроена синхронизация времени по SNTP, то время в логе будет корректное, внешнее. Если синхронизация не настроена, то время будет отсчитываться с 1 января 2000 года плюс количество секунд, которые устройство проработало (Uptime).

ГЛАВА 9. PORT SECURITY

9.1. Port Security на коммутаторах АЛСиТЕК

К механизму Port Security на коммутаторах АЛСиТЕК относится ряд функций, предназначенных для повышения безопасности использования оборудования.

Основные настройки Port Security выполняются непосредственно в контексте интерфейса, однако есть некоторые настройки, которые выполняются в режиме глобального конфигурирования.

9.2. Настройка Port Security на интерфейсах

Ограничение MAC-адресов

На коммутаторах АЛСиТЕК предусмотрена возможность ограничения количества динамически изученных MAC-адресов на интерфейсе. Ограничение настраивается в диапазоне от 0 до 600. При значении 0 коммутатор перестает изучать MAC-адреса приходящих с этого интерфейса пакетов. По умолчанию ограничение количества MAC-адресов на интерфейсе отсутствует.

Для работы ограничения необходимо включить Port Security глобально:

```
(als_sw) #configure
(als_sw) (configure) #port-security
```

В примере ниже на интерфейсе 0/1 включается ограничение в 5 MAC-адресов:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #port-security
(als_sw) (configure) (interface 0/1) #port-security max-dynamic 5
```

После ввода этих команд будут изучены не более 5 различных MAC-адресов, при поступлении нового пакета с неизученным MAC-адресом пакет будет отброшен.

Изоляция интерфейсов (Port Isolation)

Данная служба позволяет предотвратить передачу данных между устройствами, подключенными к разным интерфейсам одного коммутатора. Для изоляции нескольких интерфейсов друг от друга необходимо объединить их в одну группу. Один интерфейс не может быть добавлен более чем в одну группу изоляции.

Объединение в одну группу гарантирует, что трафик между интерфейсами этой группы не будет передаваться. Всего разрешено использовать три группы, с номерами от 0 до 2. По умолчанию изоляция выключена, а группы не содержат ни одного интерфейса.

При настройке изоляции интерфейсов трафик будет продолжать передаваться:

- между интерфейсами определенной группы и интерфейсами без группы;
- между интерфейсами разных групп.

Включение изоляции

Для помещения интерфейсов в определенную группу изоляции используется команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1-0/4
(als_sw) (configure) (interface 0/1-0/4) #switchport protected 0
```

В данном примере интерфейсы с 0/1 по 0/4 включены в группу с номером 0 и изолированы друг от друга. Передача пакетов между интерфейсами 0/1-0/4 невозможна.

Выключение изоляции

Для того, чтобы исключить определенные интерфейсы из группы, используется команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #no switchport protected 0
```

В данном примере из группы 0 исключен интерфейс 0/1, интерфейсы с 0/2 по 0/4 из предыдущего примера остались изолированы друг от друга. Если исключить из группы все интерфейсы, изоляция будет отключена для этой группы.

Private VLAN

Private VLAN — технология, позволяющая изолировать интерфейсы в определенных VLAN. В случае указания определенного VLAN как private порты коммутатора разделяются на две группы — isolated (трафик между такими портами не будет проходить в private VLAN) и promiscuous (порты могут передавать любой трафик, в том числе и в private VLAN).

В качестве примера приведем настройку коммутатора, к 0/3 интерфейсу которого подключена сеть провайдера, а к 0/1 и 0/2 интерфейсам подключены сети условных организаций. Каждая организация будет работать в private VLAN, изолированно от другой организации, однако в сеть провайдера и обратно трафик будет передаваться без препятствий.

Количество private VLAN, которые можно создать на устройстве, равно 16. Поддержка технологии private VLAN зависит от модели коммутатора.

Шаг 1. Создание и объявление private VLAN

Создание и объявление private VLAN осуществляется в контексте настройки VLAN с помощью следующих команд:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 10
(als_sw) (Vlan) #vlan 10 private
(als_sw) (Vlan) #exit
```

В данном случае был создан VLAN 10, и для него была включена технология private VLAN.

Также необходимо включить обработку созданного VLAN:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1,0/2,0/3
(als_sw) (configure) (interface 0/1-0/3) #vlan participation include 10
(als_sw) (configure) (interface 0/1-0/3) #vlan tagging 10
```

Шаг 2. Указание promiscuous интерфейсов

При указании private VLAN все интерфейсы считаются isolated, это означает, что трафик между всеми интерфейсами в данном VLAN передаваться не будет.

Для указания определенного порта в качестве promiscuous используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/3
(als_sw) (configure) (interface 0/3) #vlan promiscuous 10
```

После выполнения этой команды трафик между интерфейсами 0/1 и 0/3, а также 0/2 и 0/3, будет передаваться свободно, однако между интерфейсами 0/1 и 0/2 трафик в private VLAN 10 передаваться не будет.

Защита от штормов (Storm Control)

Эта служба используется для ограничения количества передаваемых пакетов определенного типа. Как правило, служба настраивается на интерфейсах абонентов, поскольку они могут посылать нежелательный трафик в сеть. Ограничивать можно входящий broadcast-трафик, multicast-трафик и неизвестный unicast-трафик. Трафик unicast считается неизвестным, если MAC-адрес назначения пакета не изучен на коммутаторе.

Настройка защиты от штормов заключается в указании максимальной скорости входящего трафика. Есть два способа настройки: указание максимальной скорости в пакетах в секунду (pps) или в килобитах в секунду (kbps). На коммутаторе можно настроить только один вариант защиты, настроить совместно ограничение в pps, и в kbps нельзя.

Включение ограничения входящего трафика с максимальной скоростью в пакетах в секунду

Ниже представлены команды для установки значения ограничения для всех трех типов трафика:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #storm-control broadcast rate 20
(als_sw) (configure) (interface 0/1) #storm-control multicast rate 30
(als_sw) (configure) (interface 0/1) #storm-control unicast rate 50
```

В этом примере для интерфейса 0/1 установлено ограничение в 20 пакетов в секунду для broadcast-трафика. К примеру, если на интерфейс поступает broadcast-трафик 21 пакетов в секунду, то пройдут только 20 пакетов, а 21-ый будет отброшен. Для multicast-трафика установлено ограничение в 30 пакетов в секунду, а для DLF пакетов в 50 пакетов в секунду.

Включение ограничения входящего трафика с максимальной скоростью в килобитах в секунду

Ниже представлены команды для установки ограничения для всех типов трафика:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #storm-control broadcast kbps 64
(als_sw) (configure) (interface 0/1) #storm-control multicast kbps 128
(als_sw) (configure) (interface 0/1) #storm-control unicast kbps 192
```

В этом примере для интерфейса 0/1 установлено ограничение в 64 килобита в секунду для broadcast-трафика. К примеру, если на интерфейс будет поступать broadcast-трафик со скоростью 128 килобит в секунду, то пройдет только половина. Для multicast-трафика установлено ограничение в 128 кбит/с, для DLF трафика ограничение установлено в 192 кбит/с.

Отключение ограничения входящего трафика

Отключение может потребоваться перед тем, как сменить режим ограничения с pps на kbps и обратно. Выполнение этих команд приведет к отключению защиты от штормов на интерфейсе 0/1:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #no storm-control broadcast rate
(als_sw) (configure) (interface 0/1) #no storm-control multicast rate
(als_sw) (configure) (interface 0/1) #no storm-control unicast rate
```

После отключения ограничения одного типа (например, в pps) на всех интерфейсах коммутатора можно настроить ограничение другого типа (например, в kbps).

Служба обнаружения петель (LBD)

Данная служба позволяет автоматически определять, есть ли Ethernet-петля за конкретным интерфейсом. Как правило, эта служба включается на абонентских интерфейсах и предотвращает образование петель следующим образом:

- коммутатор периодически шлет специальный Ethernet-пакет на указанные интерфейсы и принимает ответ;
- если отправленный коммутатором пакет вернется на тот же интерфейс, с которого был отправлен — детектируется петля;
- при обнаружении петли физический порт коммутатора выключается;
- выключенный порт коммутатора восстанавливается через некоторое время автоматически, после чего процесс детектирования петель на этом интерфейсе повторяется;
- если петля будет обнаружена вновь — порт снова выключается и процесс повторяется.

При блокировке в таблице интерфейсов статус заблокированного интерфейса примет значение "LBD Disabled". Все события службы LBD записываются в лог. По умолчанию детектирование петель отключено.

Общая схема для службы LBD изображена ниже:

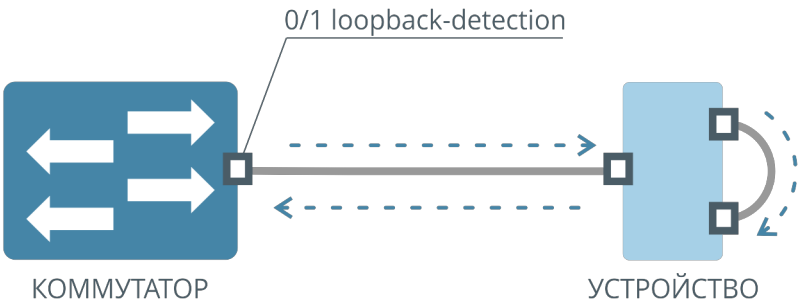

Рисунок 16. Схема работы службы обнаружения петель

Включение LBD

Для включения службы на определенном интерфейсе используется команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #loopback-detection
```

Служба LBD не может быть включена на интерфейсах совместно со службой LBDUD.

Выключение LBD

Для выключения используется команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #no loopback-detection
```

Дополнительные настройки LBD

Для службы обнаружения петель возможно изменение параметров работы. Период восстановления интерфейса, выключенного при обнаружении петли, можно изменить командой:

```
(als_sw) #configure
(als_sw) (configure) #loopback-detection recovery time <seconds>
```

Время указывается в секундах и может принимать значения от 1 до 1800 секунд. По умолчанию время восстановления равно 60 секундам.

Для службы обнаружения петель также возможно установить режим определения петли следующей командой:

```
(als_sw) #configure
(als_sw) (configure) #loopback-detection mode <single-port|multi-port>
```

- single-port — определяет петлю при получении пакета на порту, через который этот пакет был отправлен. Подходит для определения петель на оборудовании абонента, находящимся за портом коммутатора. Установлено по умолчанию;
- multi-port — определяет петлю при получении пакета на любом порту коммутатора, если этот пакет был отправлен с любого порта этого коммутатора. Подходит для определения петель как на оборудовании за портом коммутатора, так и между портами самого коммутатора.

По умолчанию служба LBD выполняет отправку пакетов LBD с интервалом в 1 секунду. Данный интервал можно изменить следующей командой:

```
(als_sw) #configure
(als_sw) (configure) #loopback-detection detection interval <seconds>
```

Время указывается в секундах и может принимать значения от 1 до 1800 секунд. По умолчанию интервал отправки пакетов равен 1 секунде.

По умолчанию служба LBD полностью отключает порт, на котором была обнаружена петля. Есть возможность задать действие при обнаружении петли следующей командой:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #loopback-detection control <action>
```

Параметры:

- <action> может принимать следующие значения:
    - shutdown — при обнаружении петли служба LBD будет отключать интерфейс. Состояние по умолчанию;
    - block — при обнаружении петли служба LBD будет переводить порт в заблокированное состояние (как при блокировке порта службой STP). При этом подключение порта не будет оборвано, но трафик ходить через заблокированный порт не будет.

Служба обнаружения однонаправленных соединений (LBDUD)

Детектирование однонаправленных соединений позволяет устройствам, подключенным однонаправленными оптическими модулями, отслеживать физическую конфигурацию кабелей и обнаруживать появление однонаправленных соединений. При обнаружении однонаправленного соединения соответствующий интерфейс отключается.

Общая схема для службы LBDUD изображена ниже:

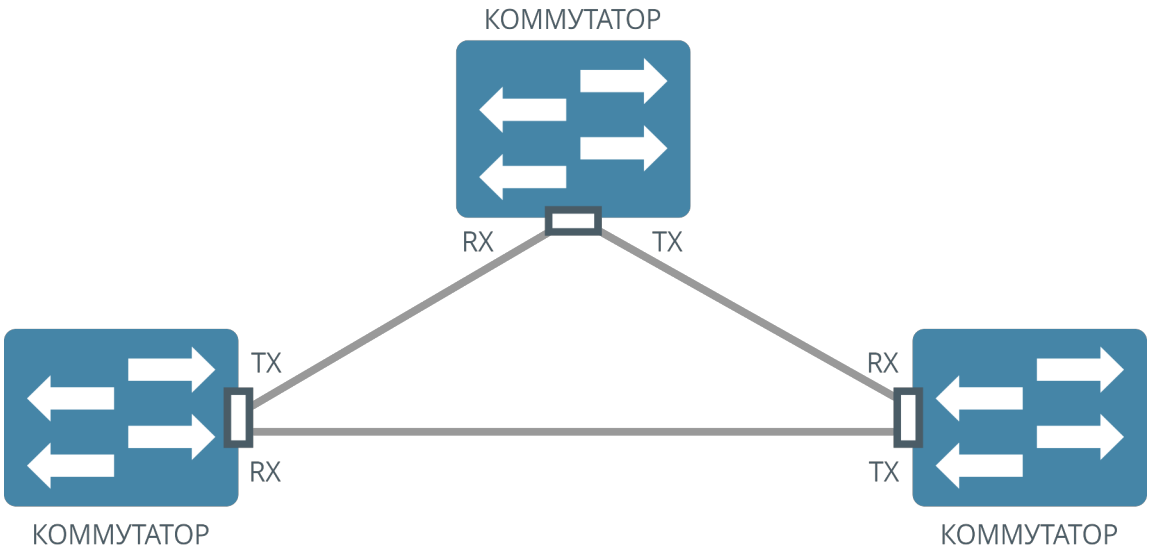

Рисунок 17. Схема работы службы LBDUD

Технология LBDUD несовместима с аналогичными решениями других производителей. Для корректной работы служба LBDUD должна быть включена на всех устройствах в схеме выше.

Включение LBDUD

Для включения службы на определенном интерфейсе используется команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #loopback-detection unidirectional
```

Служба LBDUD не может быть включена на интерфейсах совместно со службой LBD.

Выключение LBDUD

Выключить службу на интерфейсе можно командой:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #no loopback-detection unidirectional
```

Дополнительные настройки LBDUD

Для службы детектирования однонаправленных соединений возможно изменение параметров работы. Период восстановления интерфейса, выключенного при обнаружении однонаправленного соединения, можно изменить командой:

```
(als_sw) #configure
(als_sw) (configure) #loopback-detection unidirectional recovery time <seconds>
```

Время указывается в секундах и может принимать значения от 1 до 1800 секунд. Если указать значение времени равным нулю, интерфейс не будет автоматически восстанавливаться. Значение восстановления по умолчанию — 60 секунд.

Туннель для прозрачного прохождения RMA-пакетов

В некоторых случаях может быть необходимо пробросить пакеты служебных протоколов прозрачно сквозь коммутатор по каким-либо причинам. При построении такого туннеля на входном коммутаторе MAC-адреса пакетов меняются на определенный MAC-адрес, заданный конфигурацией, а на выходе из туннеля адреса меняются обратно в соответствии с протоколом пакета.

Шаг 1. Создание туннеля

Для создания туннеля необходимо включить службу глобально и указать интерфейсы для туннеля:

```
(als_sw) #configure
(als_sw) (configure) #l2protocol-tunnel
(als_sw) (configure) #interface 0/1-0/2
(als_sw) (configure) (interface 0/1-0/2) #l2protocol-tunnel
```

Шаг 2. Указание MAC-адреса для переноса пакетов

```
(als_sw) #configure
(als_sw) (configure) #l2protocol-tunnel mac <mac>
```

Шаг 3. Выбор режима работы интерфейсов

Команда указания режима для поведения туннеля на данном интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #l2protocol-tunnel mode <access|tunnel>
```

Параметры:

- access — при входе RMA-пакетов на этот интерфейс их Destination MAC будет заменен на MAC-адрес, указанный в конфигурации для туннелирования, и они будут переданы обычным образом;
- tunnel — при входе пакетов с Destination MAC, заданным в конфигурации для туннелирования, MAC-адрес будет заменен на изначальный, в зависимости от типа пакета.

Шаг 4. Выбор протоколов для туннелирования

Каждый протокол имеет один или несколько закрепленных за ним MAC-адресов. Включить туннелирование определенного протокола можно командой:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #l2protocol-tunnel <protocol>
```

Параметры:

- <protocol> может принимать следующие значения:
  - stp — Включает/Выключает туннелирование для STP BPDU;
  - fast — Включает/Выключает туннелирование для Fast-Protocols BPDU (например Pause Frame);
  - slow — Включает/Выключает туннелирование для Slow-Protocols BPDU (например LACP);
  - lldp — Включает/Выключает туннелирование для LLDP BPDU;
  - is-is — Включает/Выключает туннелирование для IS-IS Hello и Link-State BPDUs.

Шаг 5. Просмотр состояния туннелирования

Для просмотра настроек туннелирования служит следующая команда:

```
(als_sw) #show l2protocol-tunnel 0/1-0/2

  Interface  Protocol  Rx Counter  Tx Counter
  ---------  --------  ----------  ----------
  0/1        stp       1           0
             fast      1           0
             slow      1           0
             lldp      1           0
             is-is     2           0
  0/2        stp       0           1
             fast      0           1
             slow      0           1
             lldp      0           1
             is-is     0           2
```

Данная команда выводит список интерфейсов, на которых включено туннелирование, а также счетчики входящих и исходящих пакетов для каждого типа пакетов по интерфейсам.

ГЛАВА 10. LLDP

10.1. Введение в LLDP

Link Layer Discovery Protocol (LLDP) — протокол канального уровня, позволяющий сетевым устройствам (LLDP агентам) анонсировать в сеть информацию о себе, а также собирать и накапливать информацию о других устройствах. Протокол описан в стандарте IEEE 802.1AB.

10.2. LLDP на коммутаторах АЛСиТЕК

Коммутатор при включении службы LLDP начинает периодически отправлять LLDP-сообщения соседним сетевым устройствам. LLDP-сообщения, приходящие от других устройств, сохраняется в памяти коммутатора на время, указанное в самом LLDP-сообщении. Независимо от текущего состояния, коммутатор не перенаправляет на другие интерфейсы полученные LLDP-сообщения.

Настройка

Шаг 1. Включение оправки LLDP-сообщений на интерфейсе

Для включения отправки сообщений используется команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #lldp transmit
```

После выполнения этой команды коммутатор будет отправлять с интерфейса 0/1 LLDP-сообщения со следующими полями:

- MAC-адрес коммутатора. Передается в поле "Chassis ID", подтип "MAC address";
- наименование интерфейса. Передается в поле "Port ID", подтип "Locally assigned";
- время хранения. Передается в поле "Time To Live".

По умолчанию коммутатор отправляет LLDP-сообщения каждые 30 секунд.

Шаг 2. Включение приема LLDP-сообщений на интерфейсе

Для включения приема сообщений от других устройств используется команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #lldp receive
```

После выполнения этой команды коммутатор будет принимать и хранить сведения об устройстве, подключенном к интерфейсу 0/1.

Шаг 3. Изменение периода отправки LLDP-сообщений (опционально)

По умолчанию коммутатор отправляет сообщения каждый 30 секунд. Это значение можно изменить командой:

```
(als_sw) #configure
(als_sw) (configure) #lldp timers interval 5
```

Период можно указать от 5 до 32768 секунд.

Шаг 4. Добавление в передаваемые сообщения IP-адреса коммутатора (опционально)

Сведения об устройстве передаются в виде набора TLV, описанные в стандарте IEEE 802.1AB. TLV это совокупность трех полей: поле типа (определяет содержимое TLV), поле размера и поле самих данных. Данные, в свою очередь, также могу иметь сложную структуру, формат которой будет определяться подтипом. Помимо обязательных TLV, таких как "Chassis ID" (TLV Тип 1), "Port ID" (TLV Тип 2) и "Time To Live" (TLV Тип 3), в передаваемые пакеты можно добавить ряд специальных TLV.

Для того, чтобы добавить в передаваемые пакеты IP-адрес коммутатора (TLV с типом 8), выполняем команду:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #lldp transmit-mgmt
```

После выполнения этой команды коммутатор будет передавать свой IP-адрес в сообщениях LLDP. IP-адрес коммутатора передается в виде четырех октетов, с подтипом 1, что и означает IPv4-адрес.

Шаг 5. Добавление в передаваемые сообщения имени устройства (опционально)

С помощью следующей настройки можно включить отправку System Name устройства в TLV с типом 5 (System Name TLV):

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #lldp transmit-tlv sys-name
```

После выполнения этой команды коммутатор будет передавать строку приглашения CLI в сообщениях LLDP. Приглашение можно изменить следующей командой:

```
(als_sw) #set prompt "switch1"
(switch1) #
```

Есть и другой вариант. По стандарту IEEE 802.1AB, если устройство совместимо с RFC 3418, то в качестве System Name нужно использовать sysName объект SNMP. Для включения совместимости с этим стандартом нужно выполнить следующую команду:

```
(als_sw) (configure) #lldp rfc3418
```

Изменить sysName можно с помощью следующей команды:

```
(als_sw) (configure) #snmp-server sysname "switch1"
```

Шаг 6. Добавление в передаваемые сообщения VLAN ID интерфейса (PVID) (опционально)

С помощью следующей настройки можно включить отправку IEEE 802.1 Organizationally Specific TLV с подтипом 1

```
(als_sw) (configure) #lldp transmit-tlv port-vlan
```

Шаг 7. Добавление в передаваемые сообщения настроек агрегации (опционально)

С помощью следующей настройки можно включить отправку IEEE 802.3 Organizationally Specific TLV с подтипом 3

```
(als_sw) (configure) #lldp transmit-tlv link-aggregation
```

Шаг 8. Добавление в передаваемые сообщения максимального размера фрейма (опционально)

С помощью следующей настройки можно включить отправку IEEE 802.3 Organizationally Specific TLV с подтипом 4

```
(als_sw) (configure) #lldp transmit-tlv max-frame-size
```

Просмотр

Просмотр информации о других устройствах, которые сообщили о себе по протоколу LLDP, на всех интерфейсах, настроенных на прием LLDP:

```
(als_sw) #show lldp remote-device all

LLDP Remote Device Summary

Local
Interface  RemID    Chassis ID            Port ID             System Name
---------  -------  --------------------  ------------------  -----------------
-
0/1        1        172.17.1.5            20
```

Более детальную информацию можно получить командой:

```
(als_sw) #show lldp remote-device detail 0/1

LLDP Remote Device Detail:

  Local Interface.............................. 0/1

  Remote Identifier............................ 1
  Chassis ID Subtype........................... Network Address
  Chassis ID................................... 172.17.1.5
  Port ID Subtype.............................. Local
  Port ID...................................... 20
  Time To Live................................. 63
  Port Description.............................
  System Name.................................
  System Description..........................
  System Capabilities Supported...............
  System Capabilities Enabled.................
  Management Address Subtype..................... Reserved
  Management Address..........................
  Port VLAN ID................................. 1
  Aggregation capability....................... not capable of being
                                                aggregated
  Aggregation status........................... not currently in aggregation
  Aggregated port ID........................... 0
  Maximum Frame Size........................... 1518

(als_sw) #
```

10.3. Введение в LLDP-MED

LLDP-MED (Media Endpoint Discovery) — расширение стандарта LLDP, которое позволяет обнаруживать сетевые политики, отслеживать местоположения устройств и топологию, выполнять инвентаризацию устройств в сети и определение их характеристик.

Протокол LLDP-MED делит все устройства на две основные категории:

- Сетевые устройства (Network Connectivity Devices): коммутаторы, маршрутизаторы, беспроводные точки доступа и так далее
- Конечные устройства (Endpoint Devices):
    - Class I. Generic Endpoint. Общий класс для всех конечных устройств
    - Class II. Media Endpoint. Медиашлюзы, медиасерверы
    - Class III. Communication Endpoint. IP-телефоны

LLDP-MED используется между сетевыми устройствами и конечными устройствами.

10.4. LLDP-MED на коммутаторах АЛСиТЕК

Коммутатор при включении службы LLDP-MED начинает отслеживать подключенные к нему конечные устройства.

Настройка

Шаг 1. Включение LLDP-MED

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #lldp med
```

Шаг 2. Добавление в передаваемые сообщения информации для инвентаризации (опционально)

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #lldp med transmit-tlv inventory
```

После включения в отправляемые сообщения информации для инвентаризации туда будут включаться следующие TLV, описанные в стандарте ANSI/TIA-1057:

- Inventory — Hardware Revision TLV (OUI = 00-12-BB, Subtype = 5);
- Inventory — Firmware Revision TLV (OUI = 00-12-BB, Subtype = 6);
- Inventory — Software Revision TLV (OUI = 00-12-BB, Subtype = 7);
- Inventory — Serial Number TLV (OUI = 00-12-BB, Subtype = 8);
- Inventory — Manufacturer Name TLV (OUI = 00-12-BB, Subtype = 9);
- Inventory — Model Name TLV (OUI = 00-12-BB, Subtype = 10).

Просмотр

Просмотр информации о других устройствах, которые сообщили о себе по протоколу LLDP-MED, на всех интерфейсах, настроенных на прием LLDP и с включенным функционалом LLDP-MED:

```
(als_sw) #show lldp med remote-device all

  Interface  RemoteID  Device Class
  ---------  --------  --------------------
  0/17       1         Endpoint Class I

(als_sw) #
```

Более детальную информацию можно получить следующей командой:

```
(als_sw) #show lldp med remote-device detail 0/17

  Local Interface.............................. 0/17

  Remote Identifier............................ 1

Capabilities:

  MED Capabilities Supported.................... capabilities, networkpolicy,
                                                 location, extended-pd,
                                                 inventory
  MED Capabilities Enabled...................... capabilities
  Device Class.................................. Endpoint Class I

Network Policies:

Inventories:

(als_sw) #
```

В приведенном примере конечное устройство класса 1 сообщило, что оно поддерживает отсылку LLDP MED TLV capabilities, networkpolicy, location, extended-pd, inventory. В последнем принятом сообщении присутствует только одна LLDP MED TLV capabilities.

ГЛАВА 11. VLAN

11.1. Введение

VLAN (англ. Virtual Local Area Network, виртуальная локальная сеть) — группа узлов сети, трафик которой, в том числе широковещательный, на канальном уровне (Layer 2) полностью изолирован от трафика других узлов сети. VLAN позволяет создавать полностью изолированные сегменты сети путем логического конфигурирования коммутаторов, не прибегая к изменению физической структуры сети.

Внутри коммутатора настройки VLAN позволяют изолировать трафик между интерфейсами:

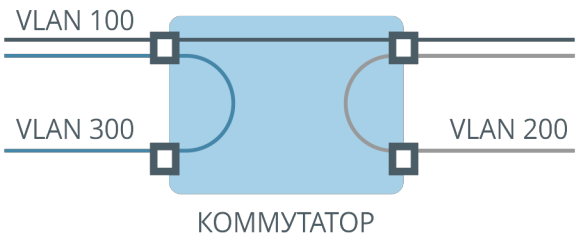

Рисунок 18. Изоляция трафика с помощью VLAN

VLAN строятся на основе тегов. Тег — метка-идентификатор, добавляемая к Ethernet-кадру и используемая для передачи информации о принадлежности кадра к определенному VLAN. Описание приведено в стандарте IEEE 802.1Q.

Коммутатор оперирует только тегированными пакетами. По умолчанию все интерфейсы коммутатора относятся к условной сети VLAN 1 с идентификатором VID, равным 1. Все нетегированные пакеты на входе интерфейса коммутатора тегируются VLAN 1, а на выходе тег VLAN 1 снимается.

Прохождение пакета внутри коммутатора изображено на схеме ниже:

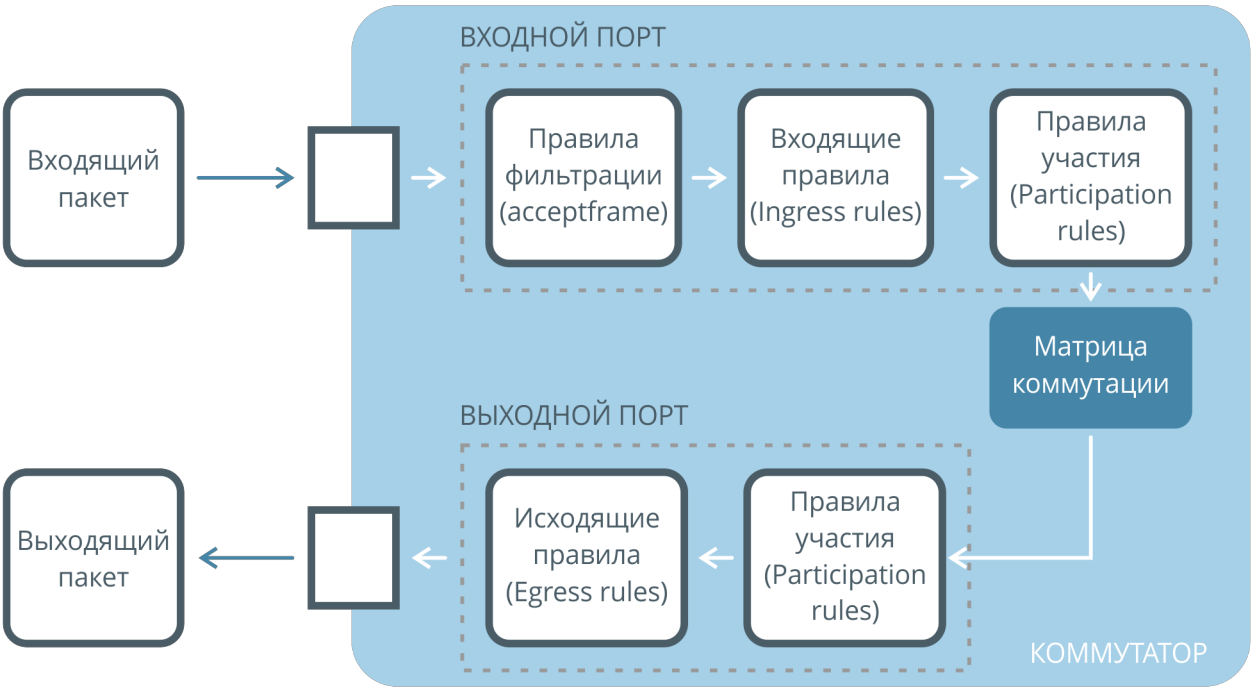

Рисунок 19. Схема прохождения пакета через коммутатор

11.2. Настройка VLAN

Настройка VLAN на коммутаторе проходит в несколько этапов:

- создание VLAN;
- настройка правил фильтрации;
- создание входящих правил;
- создание правил участия;
- создание исходящих правил.

Создание VLAN

Для того чтобы коммутатор мог принимать и обрабатывать VLAN, необходимо его создать. Для создания VLAN используется следующая команда:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan <vlan list>
(als_sw) (Vlan) #exit
```

Например, для создания трех VLAN (244, 600 и 1000 на примере ниже) используют команду:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 244,600,1000
(als_sw) (Vlan) #exit
```

В этой команде можно задавать набор VLAN следующими способами:

- перечисление VLAN через запятую (100,101,102,110);
- указание интервала VLAN через знак "-" (100-102);
- комбинированным способом, используя перечисление и интервал (100-102,110).

Каждому VLAN можно назначить имя, сделать это можно следующей командой:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan name <vlan id> <vlan name>
(als_sw) (Vlan) #exit
```

На примере ниже показаны команды назначения имени "IPTV" для VLAN 600:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan name 600 "IPTV"
(als_sw) (Vlan) #exit
```

Имя отображается в конфигурации коммутатора и в таблицах статистики.

Настройка правил фильтрации

Можно настроить интерфейс так, чтобы он принимал только тегированные пакеты. Для этого используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan acceptframe vlanonly
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
```

Также можно настроить интерфейс так, чтобы он принимал только нетегированные пакеты. Для этого используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan acceptframe admituntaggedonly
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
```

По умолчанию интерфейс принимает тегированные и нетегированные пакеты. Для задания такого поведения используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan acceptframe all
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
```

При настройке одного из режимов (vlanonly или admituntaggedonly), пакеты противоположного типа будут отброшены по правилу acceptframe.

Настройка входящих правил

Для входящего нетегированного пакета можно установить тег по одному из правил. Приоритет правила тегирования указан в порядке убывания (1 — наивысший, 3 — наименьший):

1. На основе MAC-адреса (MAC-based).
2. На основе значения поля Ethertype (Protocol-based).
3. На основе интерфейса (Port-based).

Если для пакета подходят несколько правил, то срабатывает правило, у которого выше приоритет. Например, если установлено правило на основе 0/1 интерфейса тегировать пакеты VLAN 3 и установлено глобальное правило на основе MAC-адреса тегировать пакеты VLAN 5, то пакет будет передан дальше с VLAN 5, если его MAC адрес источника совпадает с указанным в правиле тегирования на основе MAC-адреса.

По умолчанию на всех интерфейсах коммутатора входящий **нетегированный** трафик тегируется VLAN 1.

Настройка MAC-based VLAN

Для того, чтобы настроить тегирование входящего нетегированного трафика определенным VLAN, основываясь на MAC-адресе источника, используется следующая команда:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan association mac <MAC-address> <MAC-address mask> <vlan id
>
(als_sw) (Vlan) #exit
```

В данной команде с помощью поля <MAC-address mask> можно задать диапазон MAC-адресов. Если коммутатор примет пакет, у которого MAC-адрес источника попадает под одно из правил, то этому пакету будет назначен указанный в правиле тег VLAN. Правил может быть несколько. Если пакет попадает под несколько правил, то применяется только самое первое, в порядке создания.

В примере ниже для определенного MAC-адреса назначается VLAN 97:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan association mac 08:60:6e:6f:5b:6c ff:ff:ff:ff:ff:ff 97
(als_sw) (Vlan) #exit
```

Пример использования маски для задания диапазона MAC-адресов:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan association mac 08:60:6e:6f:5b:00 ff:ff:ff:ff:ff:00 97
(als_sw) (Vlan) #exit
```

После выполнения этой команды всем входящим пакетам с MAC-адресами источника в диапазоне от 08:60:6e:6f:5b:00 до 08:60:6e:6f:5b:ff будет добавлен тег VLAN 97. Обратите внимание, что на интерфейсах необходимо включить обработку VLAN 97.

Настройка Protocol-based VLAN

Настройка тегирования входящего нетегированного трафика на основе значения поля Ethertype производится в несколько шагов: сначала создается группа, ей назначается имя и VLAN, затем к группе добавляется протокол, на основе которого будут тегироваться пакеты, потом на интерфейсе применяется созданная группа.

Шаг 1. Создание группы

Для создания группы используется следующая команда (в примере использована группа 1):

```
(als_sw) #configure
(als_sw) (configure) #vlan protocol group 1
(als_sw) (configure) #exit
```

Всего разрешено использовать группы с номерами от 1 до 16 включительно.

Шаг 2. Назначение имени группы

Для назначения имени группы используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #vlan protocol group 1 named "PPPOE"
(als_sw) (configure) #exit
```

В качестве ID группы нужно использовать тот же номер, что был использован при создании. В примере группе с номером 1 назначается имя "PPPOE".

Шаг 3. Назначение VLAN группе

Для задания того, каким VLAN будут тегированы пакеты, используется следующая команда:

```
(als_sw) #vlan database
(als_sw) (Vlan) #protocol group 1 600
(als_sw) (Vlan) #exit
```

В данном примере для группы 1 назначен VLAN 600. Обратите внимание, что VLAN 600 должен быть создан, а для корректной работы нужно настроить его обработку на интерфейсах.

Шаг 4. Добавление протоколов

Для добавления протоколов, на основе которых будет присвоен тег, используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #vlan protocol group add protocol <group id> <protocol>
(als_sw) (configure) #exit
```

Значение параметра <protocol> может принимать следующие значения:

- arp — Address Resolution Protocol (ARP);
- ip — Internet Protocol (IP);
- ipx — Internetwork Packet Exchange (IPX);
- pppoe_discovery — этап обнаружения PPPoE (PPPoED);
- pppoe_session — этап сеанса PPPoE (PPPoES).

Также можно настроить тегирование пакета на основе определенного значения поля Ethertype с помощью команды:

```
(als_sw) #configure
(als_sw) (configure) #vlan protocol group add protocol <group id> ethertype <et
hertype>
(als_sw) (configure) #exit
```

В этой команде вводится числовое значение поля Ethertype. Допустимо использовать 16-ричную запись, например 0x0800.

Шаг 5. Применение группы на интерфейсе

Для того чтобы интерфейс мог обрабатывать пакеты согласно правилам группы, необходимо применить группу на интерфейсе. Для этого используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #protocol vlan group <group id>
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
```

Настройка Port-based VLAN

Для того, чтобы задать тегирование входящего нетегированного трафика определенным VLAN на интерфейсе, используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan pvid <vlan id>
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
```

Настройка правил участия

Интерфейс может обрабатывать только те VLAN, которые прописаны в правилах участия. Если на вход интерфейса (после прохождения правил фильтрации и входящих правил) попадает пакет с тегом, в обработке которого интерфейс не участвует, то такой пакет будет отброшен.

После фильтрации и обработки входящими правилами определяется участие интерфейсов в обработке пакетов с определенными значениями VLAN. Пакет может быть передан на интерфейсы, которые определены как участники VLAN, которым тегирован пакет. Если ни один из интерфейсов не является участником этого VLAN, то пакет будет отброшен.

По умолчанию все интерфейсы обрабатывают только 1 VLAN, пакеты с другими тегами будут отброшены. Для включения обработки VLAN на интерфейсе используется команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan participation include <vlan id>
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
```

Для того чтобы исключить VLAN из обработки на интерфейсе, используется команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan participation exclude <vlan id>
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
```

Команда исключения из обработки также может быть применена для VLAN 1. Это позволит запретить нетегированный трафик на интерфейсе. Того же эффекта можно достигнуть, установив режим обработки пакетов на интерфейсе vlanonly (обрабатывать только тегированные пакеты).

Настройка исходящих правил

Последним шагом обработки пакета является правило для выходящего с интерфейса трафика. На данном шаге определяется для каких значений VLAN трафик с интерфейса пакет будет выходить с тегом, а для каких без тега. Существует два режима: tagging (тег VLAN остается) и untagging (тег VLAN удаляется). По умолчанию все интерфейсы на коммутаторе находятся в режиме untagging для всех VLAN.

Для того чтобы настроить тегирование исходящего трафика, используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan tagging 100
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
```

После ввода этой команды все пакеты, получившие тег VLAN 100 на коммутаторе или пришедшие с ним, будут выходить с интерфейса 0/1 с этим тегом.

Для отключения режима тегирования используется команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #no vlan tagging 100
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
```

В этом случае пакеты, получившие тег VLAN 100 на коммутаторе или пришедшие с ним, будут выходить с интерфейса 0/1 без тега.

Просмотр VLAN

Для просмотра всех VLAN, созданных на устройстве, используется следующая команда:

```
(als_sw) #show vlan

  VLAN ID  VLAN Name                          VLAN Type
  -------  --------------------------------  ---------
  1        Default                           Default
  111                                        Static
  375      IPTV                              Static
  660                                        Static
  900                                        Static
  1000     Management                        Static
```

Эта таблица выводит список созданных VLAN и их имена (если имя было настроено).

Более детальную информацию по конкретному VLAN можно посмотреть следующей командой:

```
(als_sw) #show vlan 375

  VLAN ID   : 375
  VLAN Name : IPTV
  VLAN Type : Static

  Interface  Current     Configured  Tagging
  ---------  ----------  ----------  ----------
  0/1        Include     Include     Untagged
  ...
  0/24       Include     Include     Untagged
  0/25       Include     Include     Tagged
  ...
  0/28       Include     Include     Tagged
```

Поля в этой таблице:

- Interface — номер интерфейса;
- Current — текущее состояние участия данного интерфейса в обработке VLAN. Может быть "Include" (участвует в обработке) и "Exclude" (не

участвует в обработке);
- Configured — настроенное состояние участия данного интерфейса в обработке VLAN. Может быть "Include" (участвует в обработке) и "Exclude" (не участвует в обработке);
- Tagging — исходящее правило для VLAN на данном интерфейсе. Может быть "Tagged" (метка VLAN остается при выходе пакетов с интерфейса) и "Untagged" (метка VLAN снимается).

ГЛАВА 12. DOUBLE VLAN (Q-IN-Q)

12.1. Введение в Q-in-Q

Q-in-Q — технология, позволяющая пакету иметь несколько 802.1q тегов. Описывается стандартом IEEE 802.1ad. Для работы требуется поддержка на оборудовании, так как большинство L2 устройств работают только с одним тегом.

При назначении двойного тега пакет приобретает следующий вид:

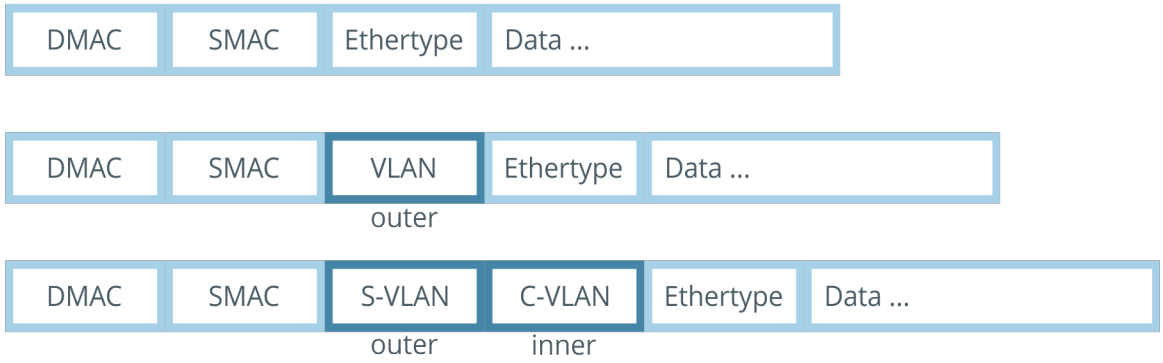

Рисунок 20. Схема пакета Ethernet при добавлении одного 802.1q тега и Double VLAN

Устройства, не поддерживающие технологию Q-in-Q, обрабатывают трафик с двойным тегом по внешнему тегу (S-VLAN на схеме).

12.2. Настройка Q-in-Q на коммутаторах АЛСиТЕК

Коммутаторы АЛСиТЕК поддерживают перечисленные ниже технологии. На входе порта:

- фильтрация пакетов по наличию или отсутствию тега (Accept frame);
- входная трансляция VLAN (Ingress translation);
- назначение S-VLAN по C-VLAN (Selective Q-in-Q);
- назначение VLAN по порту (Port-based Q-in-Q);
- фильтрация по правилам участия VLAN на входе (Ingress filter).

На выходе порта:

- фильтрация пакетов по правилам участия на выходе (Egress filter);
- удаление и сохранение VLAN по правилам VLAN для порта (VLAN tagging);
- выходная трансляция VLAN (Egress translation).

UNI-интерфейсы и NNI-интерфейсы

Интерфейсы коммутатора при работе с двойным тегом делятся на два типа:

- NNI (Network/Network Interface) — интерфейс, за которым находится сеть провайдера;
- UNI (User/Network Interface) — интерфейс, за которым находится клиент, получающий услуги в выделенном VLAN.

Рассмотрим уровень доступа провайдера, где каждый клиент получает услуги связи в выделенном VLAN. Клиент за интерфейсом 0/10 получает услуги во VLAN 10, клиент за 0/11 во VLAN 11 и так далее. Чтобы не смешивать клиентские VLAN с разных коммутаторов, можно применить Q-in-Q и поместить весь трафик с коммутатора 1 в S-VLAN 338, а с коммутатора 2 в S-VLAN 448. В свою очередь трафик, предназначенный клиентам коммутатора 1, будет идти с двойной меткой S-VLAN 338 или 448 и C-VLAN конкретного клиента.

Схема сети для данного примера изображена ниже:

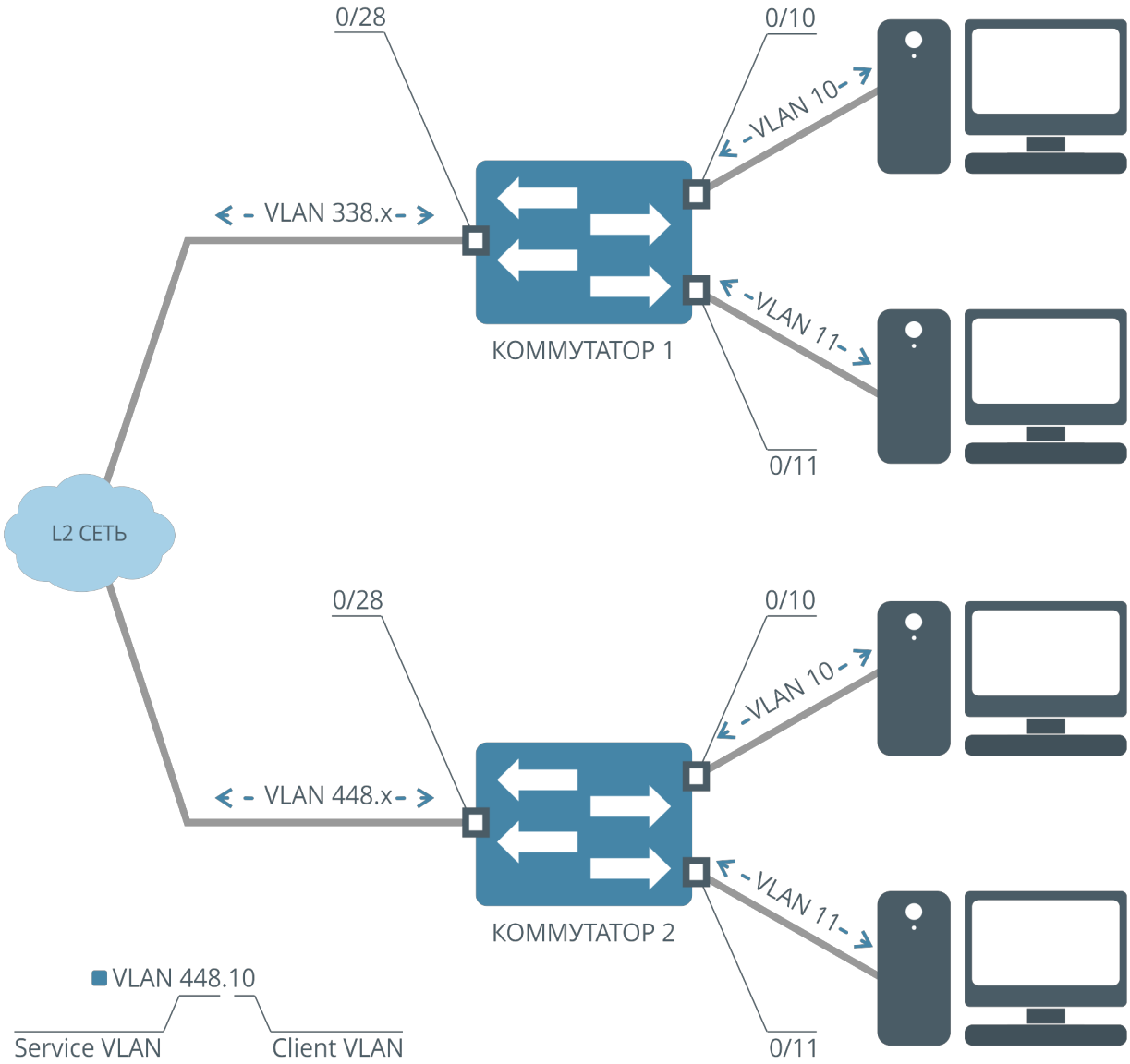

Рисунок 21. Схема предоставления услуг с использованием Double VLAN

Данное поведение можно реализовать с помощью назначения UNI и NNI интерфейсов. В нашем примере NNI-интерфейсами будет интерфейс 0/28, UNI-интерфейсами будут 0/10, 0/11, 0/12 и другие интерфейсы коммутаторов, которые ведут к клиентам. Вне зависимости от того, есть ли у пакета VLAN, UNI-интерфейс добавляет тег S-VLAN. NNI-интерфейс принимает пакет с двойным тегом и передает его согласно L2 таблице и правилам участия VLAN на NNI- и UNI-интерфейсах.

Для указания NNI-интерфейсов используется команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #mode dvlan-tunnel nni
```

Также для указания NNI-интерфейсов можно использовать сокращенный вариант команды:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #mode dvlan-tunnel
```

При этом, если назначен хоть один NNI-интерфейс, остальные интерфейсы станут UNI-интерфейсами.

Для указания UNI-интерфейсов используется команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #mode dvlan-tunnel uni
```

При этом, если назначен хоть один UNI-интерфейс, остальные интерфейсы станут NNI-интерфейсами.

Следует помнить, что все интерфейсы коммутатора можно назначить NNI-интерфейсами, применив команды "mode dvlan-tunnel" или "mode dvlan-tunnel nni" на всех интерфейсах.
Однако нельзя назначить все интерфейсы коммутатора UNI-интерфейсами, убрав со всех интерфейсов команду "mode dvlan-tunnel" или применив на них настройку "mode dvlan-tunnel uni". В этом случае поддержка технологии Q-in-Q выключится и коммутатор будет обрабатывать пакеты по внешнему тегу.

Фильтрация пакетов по наличию или отсутствию тега (Accept frame)

Фильтрация пакетов на входе позволяет:

- Отбрасывать пакеты без VLAN;
- Отбрасывать пакеты с VLAN;
- Принимать все пакеты (состояние по умолчанию).

Следующая настройка позволит интерфейсу 0/1 принимать только тегированные пакеты (пакеты с VLAN):

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan acceptframe vlanonly
```

Следующая настройка позволит интерфейсу 0/1 принимать только нетегированные пакеты:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan acceptframe admituntaggedonly
```

Следующая настройка позволит интерфейсу 0/1 принимать все пакеты, как с VLAN, так и без (настройка по умолчанию):

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan acceptframe all
```

Входная трансляция VLAN (Ingress translation)

Входная трансляция позволяет поменять внешний VLAN при входе пакета на коммутатор.

Пример работы показан на схеме ниже:

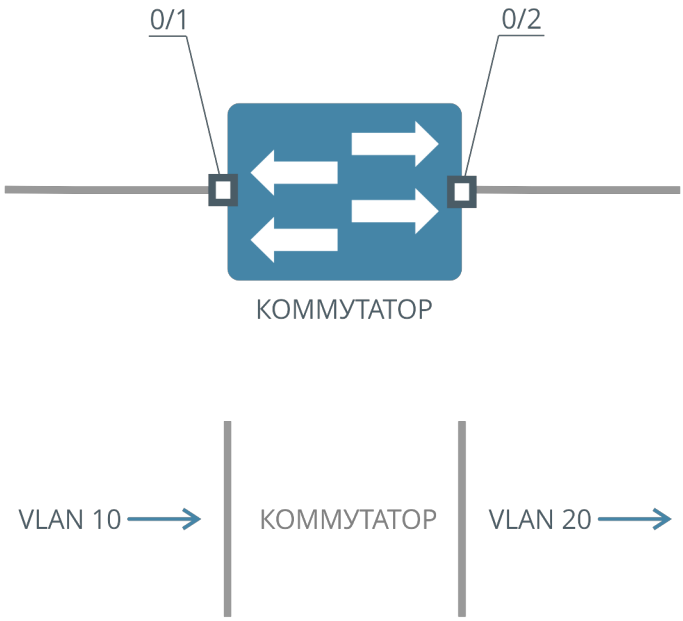

Рисунок 22. Схема работы входной трансляции

Конфигурация для схемы:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 10,20
(als_sw) (Vlan) #exit
(als_sw) #configure
(als_sw) (configure) #dvlan selective
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #dvlan cvid 10 cvid 20
(als_sw) (configure) (interface 0/1) #vlan participation include 20
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #interface 0/2
(als_sw) (configure) (interface 0/2) #vlan participation include 20
(als_sw) (configure) (interface 0/2) #vlan tagging 20
(als_sw) (configure) (interface 0/2) #exit
(als_sw) (configure) #exit
(als_sw) #
```

На интерфейс 0/1 приходит пакет с VLAN 10. Пакет обрабатывается входной трансляцией, так как на интерфейсе 0/1 имеется правило "dvlan cvid 10 cvid 20", и трансляция включена командой "dvlan selective". VLAN 10 в пакете будут заменен на VLAN 20 и пакет будет передан на интерфейс 0/2. Чтобы пакет мог пройти с интерфейса 0/1 на интерфейс 0/2, на интерфейсах должно быть настроено правило участия для VLAN 20. MAC-адрес пакета будет выучен во VLAN 10.

Назначение S-VLAN по C-VLAN (Selective Q-in-Q)

Selective Q-in-Q — технология, позволяющая назначать тег S-VLAN на пакеты с определенным C-VLAN. Selective Q-in-Q настраивается и применяется на входе UNI-интерфейсов.

Рассмотрим схему, где на интерфейс 0/1 коммутатора приходят пакеты во VLAN 10 и 20. На выходе пакеты с C-VLAN 10 должны получить S-VLAN 30, а пакеты с C-VLAN 20 — S-VLAN 40:

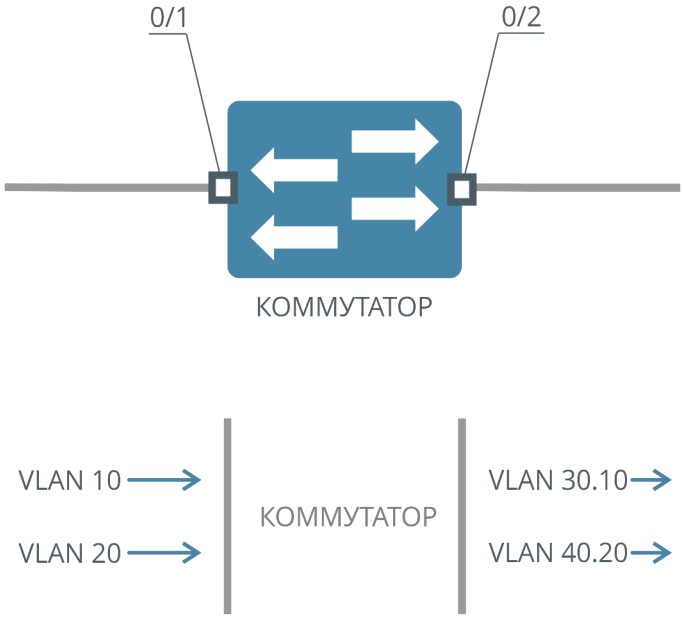

Рисунок 23. Схема добавления S-VLAN по определенному C-VLAN (Selective Q-in-Q)

Для данной схемы конфигурация будет следующей:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 10,20,30,40
(als_sw) (Vlan) #exit
(als_sw) #configure
(als_sw) (configure) #dvlan selective
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #dvlan cvid 10 svid 30
(als_sw) (configure) (interface 0/1) #dvlan cvid 20 svid 40
(als_sw) (configure) (interface 0/1) #vlan participation include 30,40
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #interface 0/2
(als_sw) (configure) (interface 0/2) #mode dvlan-tunnel
(als_sw) (configure) (interface 0/2) #vlan participation include 30,40
(als_sw) (configure) (interface 0/2) #vlan tagging 30,40
(als_sw) (configure) (interface 0/2) #exit
(als_sw) (configure) #exit
(als_sw) #
```

Selective Q-in-Q настраивается только в том случае, если на коммутаторе на одном из интерфейсов включен "mode dvlan-tunnel". На интерфейс 0/1 приходят пакеты с VLAN 10, интерфейс 0/1 является UNI-интерфейсом, так как на интерфейсе 0/2 присутствует настройка "mode dvlan-tunnel". Пакеты во VLAN 10 по правилу "dvlan cvid 10 svid 30" будут получать S-VLAN 30. Далее пакеты передаются на интерфейс 0/2 с S-VLAN 30 и C-VLAN 10. Пакеты во VLAN 20 по правилу "dvlan cvid 20 svid 40" будут получать S-VLAN 40. Далее пакеты передаются на 0/2 интерфейс с S-VLAN 40 и C-VLAN 20.

На интерфейсах 0/1 и 0/2 должно быть настроено правило участия для VLAN 30 и 40.

Назначение VLAN по порту (Port-based Q-in-Q)

При работе с двойным тегом может возникнуть необходимость поместить весь трафик с интерфейса клиента в определенный S-VLAN. Для этого используется Port-based Q-in-Q, аналог Port-based VLAN. Port-based Q-in-Q будет работать только в том случае, если пакет не попал во входную трансляцию и Selective Q-in-Q.

Рассмотрим схему, где на интерфейс 0/1 приходят пакеты во VLAN 10 и 20, всем пакетам на входе нужно добавить S-VLAN 30:

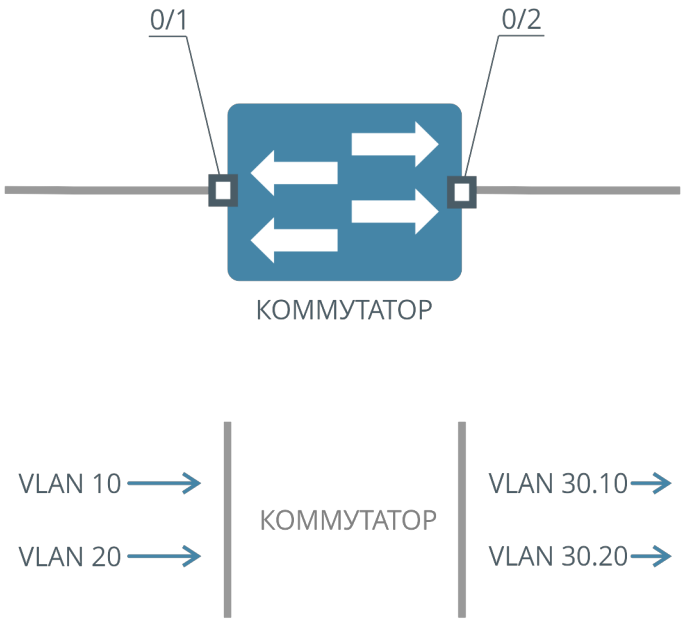

Рисунок 24. Схема добавления S-VLAN по порту (Port-based Q-in-Q)

Для данной схемы конфигурация будет следующей:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 30
(als_sw) (Vlan) #exit
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan pvid 30
(als_sw) (configure) (interface 0/1) #vlan participation include 30
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #interface 0/2
(als_sw) (configure) (interface 0/2) #mode dvlan-tunnel
(als_sw) (configure) (interface 0/2) #vlan participation include 30
(als_sw) (configure) (interface 0/2) #vlan tagging 30
(als_sw) (configure) (interface 0/2) #exit
(als_sw) (configure) #exit
(als_sw) #
```

На интерфейс 0/1 приходят пакеты с VLAN 10, интерфейс 0/1 является UNI-интерфейсом, так как на интерфейсе 0/2 присутствует настройка "mode dvlan-tunnel". Пакеты во VLAN 10 по правилу "vlan pvid 30" будут получать S-VLAN 30. Далее пакеты передаются на интерфейс 0/2 с S-VLAN 30 и C-VLAN 10. Пакеты во VLAN 20 по правилу "vlan pvid 30" будут получать S-VLAN 30. Далее пакеты передаются на 0/2 интерфейс с S-VLAN 30 и C-VLAN 20. На интерфейсах 0/1 и 0/2 должно быть правило участия для VLAN 30.

Фильтрация по правилам участия VLAN на входе (Ingress filter)

Фильтрация по правилам участия позволяет отбросить на входе интерфейса пакеты с VLAN, не удовлетворяющим правилам участия.

На входе UNI-интерфейса отбросим пакеты во VLAN 10. Пакеты во VLAN 20 должны получать S-VLAN 30 по правилу "Port-based Q-in-Q":

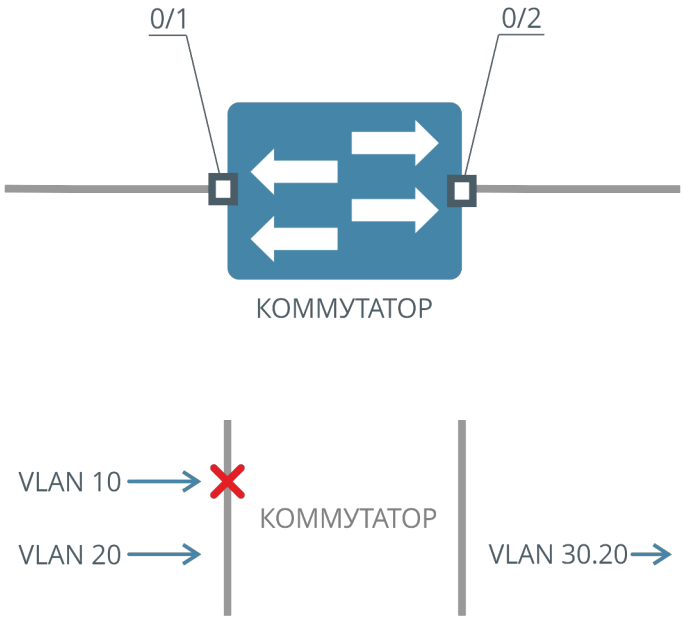

Рисунок 25. Схема работы Ingress Filter

Для данной схемы конфигурация будет следующей:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 10,20,30,500
(als_sw) (Vlan) #exit
(als_sw) #configure
(als_sw) (configure) #dvlan selective
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #dvlan cvid 10 svid 500
(als_sw) (configure) (interface 0/1) #vlan pvid 30
(als_sw) (configure) (interface 0/1) #vlan participation include 30
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #interface 0/2
(als_sw) (configure) (interface 0/2) #mode dvlan-tunnel
(als_sw) (configure) (interface 0/2) #vlan participation include 30,500
(als_sw) (configure) (interface 0/2) #vlan tagging 30
(als_sw) (configure) (interface 0/2) #exit
(als_sw) (configure) #exit
(als_sw) #
```

На интерфейс 0/1 приходят пакеты с VLAN 10, интерфейс 0/1 является UNI-интерфейсом, так как на интерфейсе 0/2 присутствует настройка "mode dvlan-tunnel". Пакеты во VLAN 10 по правилу "dvlan cvid 10 svid 500" будут получать S-VLAN 500. Далее пакеты не передаются, так как VLAN 500 отсутствует в правилах участия на интерфейсе 0/1. Пакеты во VLAN 20 по правилу "vlan pvid 30" будут получать S-VLAN 30. Далее пакеты передаются на интерфейс 0/2, так как VLAN 30 присутствует в правилах участия на интерфейсах 0/1 и 0/2.

Фильтрация пакетов по правилам участия на выходе (Egress filter)

Фильтрация по правилам участия позволяет отбросить на выходе интерфейса пакеты с VLAN, не удовлетворяющим правилам участия.

На выходе NNI интерфейса отбросим пакеты с S-VLAN 500. Пакеты во VLAN 20 должны получать S-VLAN 30 по правилу "Port-based Q-in-Q":

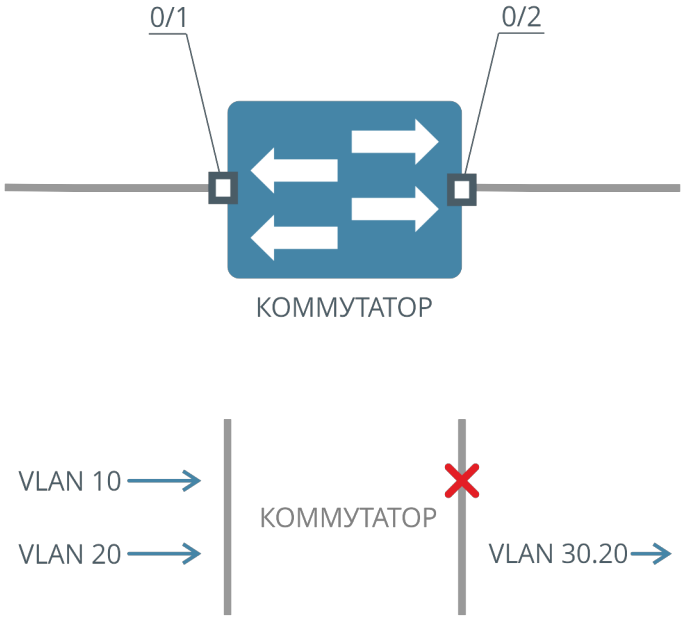

Рисунок 26. Схема работы Egress Filter

Для данной схемы конфигурация будет следующей:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 10,20,30,500
(als_sw) (Vlan) #exit
(als_sw) #configure
(als_sw) (configure) #dvlan selective
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #dvlan cvid 10 svid 500
(als_sw) (configure) (interface 0/1) #vlan pvid 30
(als_sw) (configure) (interface 0/1) #vlan participation include 30,500
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #interface 0/2
(als_sw) (configure) (interface 0/2) #mode dvlan-tunnel
(als_sw) (configure) (interface 0/2) #vlan participation include 30
(als_sw) (configure) (interface 0/2) #vlan tagging 30
(als_sw) (configure) (interface 0/2) #exit
(als_sw) (configure) #exit
(als_sw) #
```

На интерфейс 0/1 приходят пакеты с VLAN 10, интерфейс 0/1 является UNI-интерфейсом, так как на интерфейсе 0/2 присутствует настройка "mode dvlan-tunnel". Пакеты во VLAN 10 по правилу "dvlan cvid 10 svid 500" будут получать S-VLAN 500. Далее пакеты не передаются, так как VLAN 500 присутствует только на входе интерфейса 0/1 и отсутствует на других интерфейсах. Пакеты во VLAN 20 по правилу "vlan pvid 30" будут получать S-VLAN 30. Далее пакеты передаются на интерфейс 0/2, так как VLAN 30 присутствует в правилах участия на интерфейсах 0/1 и 0/2.

Удаление и сохранение VLAN по правилам VLAN для порта (VLAN tagging)

По умолчанию все VLAN-теги на интерфейсах коммутатора снимаются при выходе пакета. Такое поведение соответствует режиму "untagging" для VLAN тег на интерфейсе. Чтобы изменить это поведение и оставить VLAN в пакете, применяется режим "tagging".

Для установки режима тегирования на интерфейсе необходимо выполнить следующую команду:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan tagging 10
```

В данном примере для всего трафика во VLAN 10, исходящего с интерфейса 0/1, VLAN 10 будет оставлен в пакете.

Добавление внутренней метки VLAN в режиме Q-in-Q

В режиме Q-in-Q для UNI-интерфейсов можно добавить внутреннюю метку VLAN. При этом будет сначала добавлена внутренняя метка VLAN, затем метка VLAN по правилу PVID. На выходе NNI-интерфейса пакет будет иметь 2 метки. Добавление внутренней метки VLAN работает на входе UNI интерфейсов.

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan inner insert 10
```

Удаление внутренней метки VLAN в режиме Q-in-Q

В режиме Q-in-Q для UNI-интерфейсов можно добавить правило удаления внутренней метки VLAN. Внутренняя метка VLAN удаляется на выходе из UNI-интерфейса. Важно отметить, что сохранение или удаление внешней метки данное правило не затрагивает.

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan inner remove
```

Выходная трансляция

Выходная трансляция позволяет заменить VLAN при выходе пакета с интерфейса. Выходная трансляция всегда меняет внешний тег VLAN.

Рассмотрим случай применения трансляции. На схеме VLAN 10 и 20 объединяются в 100 VLAN, VLAN 30 преобразуется в 300 VLAN:

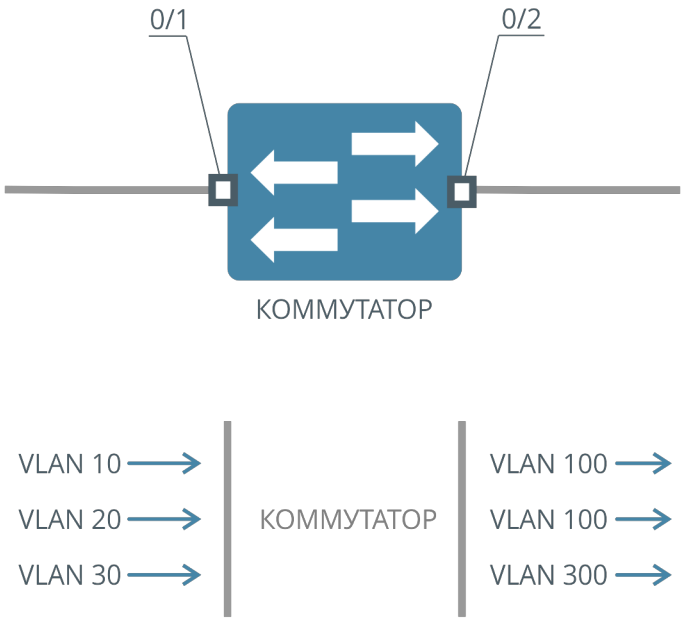

Рисунок 27. Схема работы выходной трансляции

Конфигурация для схемы выше:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 10,20,30,100,300
(als_sw) (Vlan) #exit
(als_sw) #configure
(als_sw) (configure) #dvlan selective
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan participation include 10,20,30
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #interface 0/2
(als_sw) (configure) (interface 0/2) #vlan participation include 10,20,30
(als_sw) (configure) (interface 0/2) #vlan tagging 10,20,30,100,300
(als_sw) (configure) (interface 0/2) #dvlan svid 10 svid 100
(als_sw) (configure) (interface 0/2) #dvlan svid 20 svid 100
(als_sw) (configure) (interface 0/2) #dvlan svid 30 svid 300
(als_sw) (configure) (interface 0/2) #exit
(als_sw) (configure) #exit
(als_sw) #
```

На интерфейс 0/1 приходят пакеты с VLAN 10, 20 и 30. Ни на одном из интерфейсов не включен "mode dvlan-tunnel". Пакеты по правилу участия будут переданы на интерфейс 0/2. На выходе интерфейса 0/2 по правилу выходной трансляции "dvlan svid 10 svid 100" все пакеты во VLAN 10 будут транслированы в VLAN 100, аналогично пакеты во VLAN 20 будут также транслированы во VLAN 100 по правилу "dvlan svid 20 svid 100". Пакеты во VLAN 10 и 20 после трансляции выйдут с одним тегом VLAN 100. Пакеты во VLAN 30 обрабатываются по правилу выходной трансляции "dvlan svid 30 svid 300" и выходят с одним тегом VLAN 300.

Схема обработки пакета

Прямоугольники обозначают действие, которое может быть совершено с пакетом. У таких прямоугольников может быть два выхода, помеченные как "yes" и "no". В случае, если пакет попал под обработку данным блоком, то переход осуществляется по стрелке "yes", иначе по стрелке "no". Например: блок входной трансляции имеет два выхода, если внешний VLAN в пакете будет найден в таблице входной трансляции, то переход будет осуществлен по стрелке "yes", во всех других случаях переход будет по стрелке "no".

Ромбами обозначены некоторые условия, которым удовлетворяет или не удовлетворяет пакет. Если пакет удовлетворяет условиям, переход осуществляется по стрелке "yes", в других случаях по стрелке "no".

Конечные блоки обозначены овалами с финальными действиями, которые производятся над пакетом.

Поясним на примере конфигурации:

```
configure
interface 0/1
mode dvlan-tunnel uni
exit
interface 0/2
mode dvlan-tunnel nni
exit
exit
```

Интерфейс 0/1 является UNI-интерфейсом. Интерфейс 0/2 является NNI-интерфейсом.

Если настройки "mode dvlan-tunnel nni" и "mode dvlan-tunnel uni" отсутствуют в конфигурации, то все интерфейсы являются обычными интерфейсами. И не являются NNI- или UNI-интерфейсами.

```
configure
interface 0/1
exit
interface 0/2
exit
exit
```

Обработка пакета на входе интерфейса

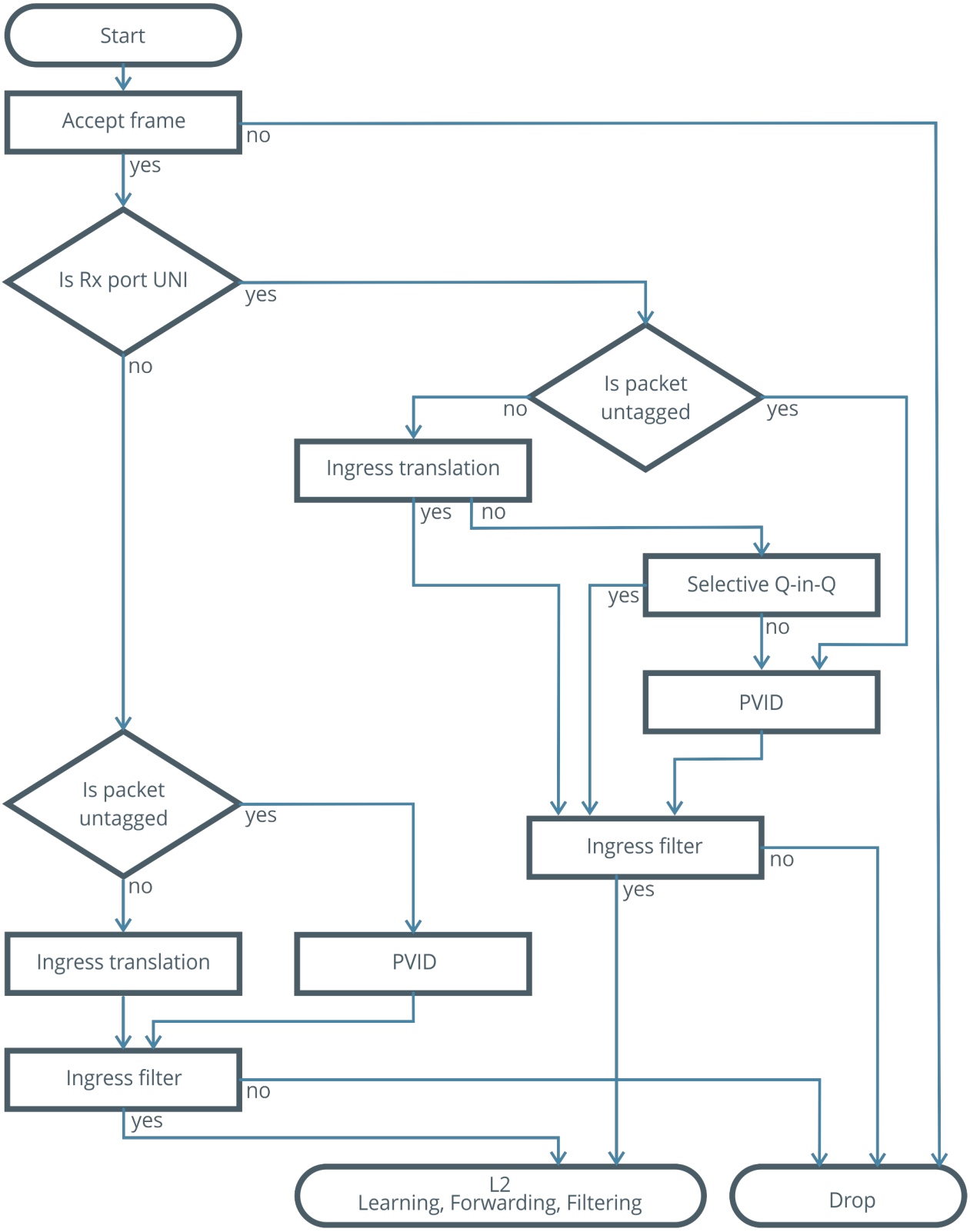

Рисунок 28. Схема обработки пакета на входе интерфейса

На схеме:

- **Accept frame** — блок фильтрации пакетов по наличию или отсутствию VLAN. Если блок пропускает пакет, то переход происходит по стрелке "yes", иначе — "no";
- **Is Rx port UNI** — проверка, является ли входной интерфейс UNI интерфейсом. Если порт является UNI-интерфейсом, переход осуществляется по стрелке "yes", в остальных случаях по стрелке "no";
- **Is Packet untagged** — проверка, есть ли у пакета VLAN. Если VLAN отсутствует, то осуществляется переход по стрелке "yes", в остальных случаях по стрелке "no";
- **Ingress translation** — входная трансляция VLAN. Если пакет попадает под правило входной трансляции, то VLAN в пакете заменяется согласно правилу и осуществляется переход по стрелке "yes", в остальных случаях по стрелке "no";
- **Selective Q-in-Q** — назначение S-VLAN по C-VLAN. Если пакет попадает под правило "Selective Q-in-Q", то пакету добавляется еще один VLAN (S-VLAN) и переход осуществляется по стрелке "yes", в остальных случаях по стрелке "no";
- **PVID** — в пакет добавляется VLAN согласно правилу "vlan pvid <VLAN>" на порту;
- **Ingress filter** — фильтрация пакетов по правилам участия на входе, с правилами участия сравнивается внешний VLAN. Если пакет проходит проверку, то переход осуществляется по стрелке "yes", в остальных случаях по стрелке "no";
- **L2 Learning, Forwarding, Filtering** — на данном этапе пакет может быть отброшен по правилам ACL, изучен по внешнему VLAN, обработан внутренней логикой коммутатора, например с помощью IGMP Snooping, или передан согласно таблице коммутации на другие порты;
- **Drop** — пакет отбрасывается, изучение MAC-адреса пакета не производится.

Обработка пакета на выходе интерфейса

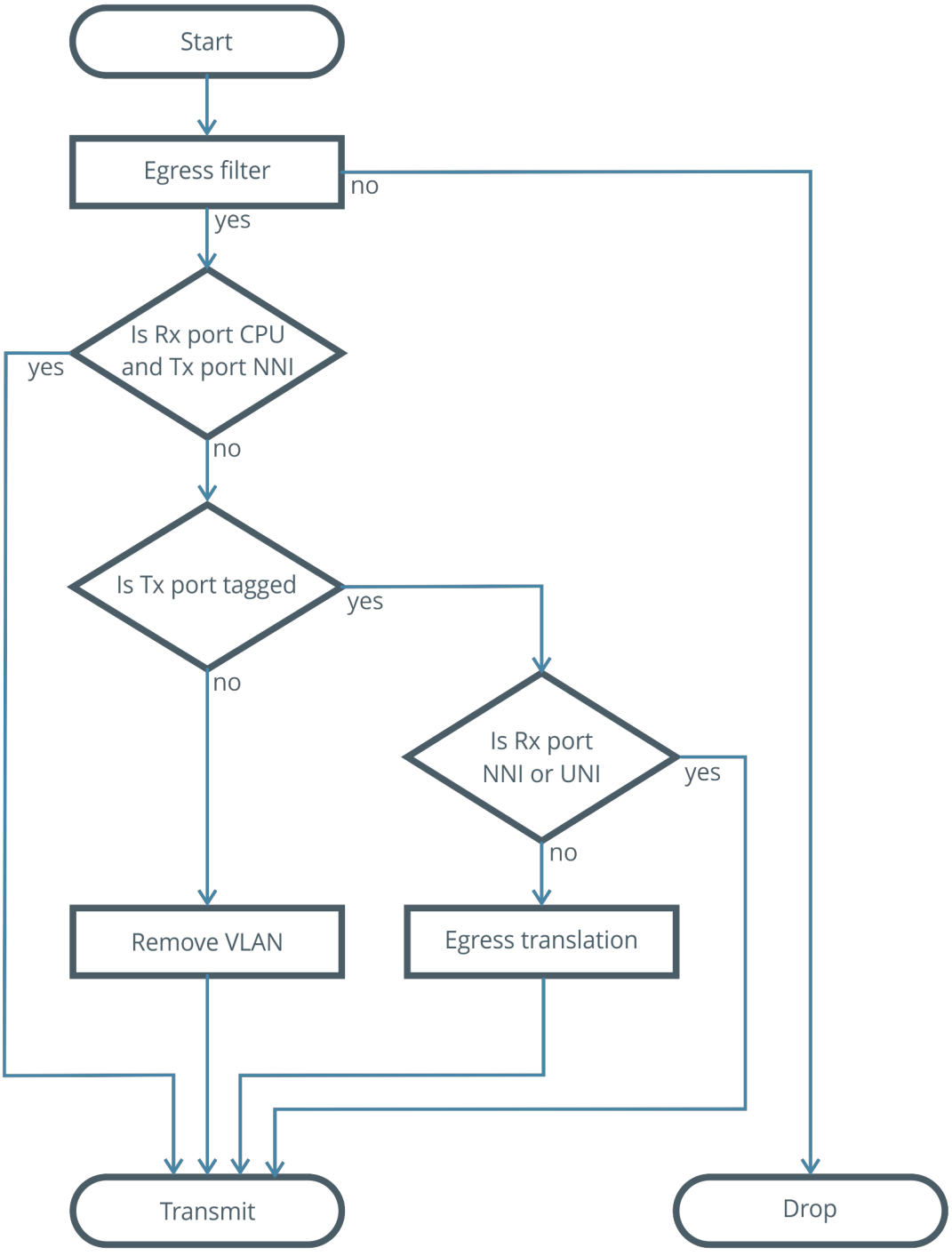

Рисунок 29. Схема обработки пакета на выходе интерфейса

На схеме:

- **Egress filter** — фильтрация пакетов по правилам участия на выходе, с правилами участия сравнивается внешний VLAN. Если пакет проходит проверку, то переход осуществляется по стрелке "yes", в остальных

случаях по стрелке "no";
- **Is Rx port CPU and Tx port NNI** — проверка, если пакет был отправлен с CPU-интерфейса (например пакет отправил IGMP Snooping) и интерфейс, на который отправляется пакет, является NNI-интерфейсом, то осуществляется переход по стрелке "yes", в остальных случаях — по стрелке "no";
- **Is Rx port NNI or UNI** — проверка, был ли пакет принят на NNI- или UNI-интерфейсе. Если в конфигурации на любом интерфейсе присутствуют строки "mode dvlan-tunnel uni" или "mode dvlan-tunnel nni", то осуществляется переход по стрелке "yes", в остальных случаях по стрелке "no";
- **Is Tx port tagged** — проверка, присутствует ли настройка сохранения тега на интерфейсе (настройка "vlan tagging <VLAN>"), куда отправляется пакет. Если такая настройка присутствует для внешнего тега VLAN в пакете, то переход осуществляется по стрелке "yes", в остальных случаях — по стрелке "no";
- **Remove VLAN** — удаление внешнего VLAN из пакета;
- **Egress translation** — выходная трансляция VLAN. Если пакет попадает под правило выходной трансляции, VLAN в пакете заменяется согласно правилу и осуществляется переход по стрелке "yes", в остальных случаях — по стрелке "no";
- **Transmit** — отправка пакета с текущими тегом VLAN;
- **Drop** — пакет отбрасывается.

Пример обработки пакета на входе

Рассмотрим пример обработки пакета при следующей конфигурации:

```
configure
dvlan selective
interface 0/1
mode dvlan-tunnel uni
dvlan cvid 20 cvid 10
dvlan cvid 30 svid 10
vlan pvid 40
vlan participation include 10,40
exit
interface 0/2
mode dvlan-tunnel nni
vlan participation include 10,20,40
vlan tagging 10,20,40
exit
exit
```

Пусть пакет с одним тегом VLAN 10 приходит на интерфейс 0/1. По схеме рассмотрим прохождение пакета. В самом начале пакет попадает в блок "Accept frame". Так как в конфигурации не отображается команда "vlan acceptframe", то применена следующая команда по умолчанию:

```
configure
interface 0/1
vlan acceptframe all
exit
exit
```

Режим "Accept frame" пропускает все пакеты. Принятый пакет имеет VLAN 10 и проходит данный блок. Переходим по стрелке "yes" к условию "Is Rx port UNI". В данном блоке проверяется, является ли интерфейс 0/1 UNI-интерфейсом.

Смотрим в конфигурацию:

```
configure
interface 0/1
mode dvlan-tunnel uni
exit
interface 0/2
mode dvlan-tunnel nni
exit
exit
```

На интерфейсе 0/1 есть настройка "mode dvlan-tunnel uni", значит интерфейс 0/1 является UNI-интерфейсом. На интерфейсе 0/2 есть настройка "mode dvlan-tunnel nni", значит интерфейс 0/2 является NNI-интерфейсом. Перемещаемся по стрелке "yes" от условия "Is Rx port UNI" к условию "Is Packet untagged", которое определяет, является ли пакет нетегированным. В нашем примере пакет с 10 VLAN. Следовательно переходим по стрелке "no" от условия "Is Packet untagged" к блоку "Ingress translation".

В блоке "Ingress translation" происходит поиск по 10 VLAN в таблице входной трансляции. Текущая конфигурация входной трансляции:

```
configure
dvlan selective
interface 0/1
dvlan cvid 20 cvid 10
exit
exit
```

Команда "dvlan selective" разрешает входную трансляцию, Selective Q-in-Q и выходную трансляцию. В таблице входной трансляции на интерфейсе 0/1 одна запись: "dvlan cvid 20 cvid 10", трансляция из VLAN 20 во VLAN 10. Принятый на интерфейсе 0/1 пакет имеет 10 VLAN, следовательно трансляция производится не будет. Переходим по ветке "no" от блока "Ingress translation" к блоку "Selective Q-in-Q".

В блоке "Selective Q-in-Q" происходит поиск по 10 VLAN в таблице Selective Q-in-Q.

Текущая конфигурация:

```
configure
dvlan selective
interface 0/1
dvlan cvid 30 svid 10
exit
exit
```

Команда "dvlan selective" разрешает входную трансляцию, Selective Q-in-Q и выходную трансляцию. В таблице Selective Q-in-Q одна запись: "dvlan cvid 30 svid 10", по C-VID 30 добавляется S-VID 10. Принятый на интерфейсе 0/1 пакет имеет 10 VLAN, следовательно Selective Q-in-Q не обработает пакет. Переходим по ветке "no" от блока "Selective Q-in-Q" к блоку "PVID".

Текущие настройки "PVID" на интерфейсе 0/1:

```
configure
interface 0/1
vlan pvid 40
exit
exit
```

Всем пакетам, приходящим на интерфейс 0/1 и не попавшим под другие правила, добавляется VLAN 40. Пакет на выходе из блока будет иметь 2 VLAN: S-VID 40 и C-VID 10. От блока "PVID" существует только один безусловный переход в блок "Ingress filter".

Текущие настройки "Ingress Filter" на интерфейсе 0/1:

```
configure
interface 0/1
vlan participation include 10,40
exit
exit
```

Разрешены VLAN 10, 40 и 1 ("vlan participation include 1" является настройкой по умолчанию и не отображается в конфигурации). Пакет имеет внешний тег VLAN 40. Значит пакет пройдет проверку, переходим по стрелке "yes" от блока "Ingress Filter" к блоку "L2 Learning, Forwarding, Filtering".

Обработка пакета на входе завершена. Пакет имеет два тега VLAN: внешний 40 и внутренний 10.

Пример обработки пакета на выходе

По правилам участия VLAN пакет может быть передан на интерфейс 0/2. Рассмотрим обработку пакета на выходе с интерфейса 0/2.

Пакет попадает на блок "Egress filter". Текущие настройки "Egress filter" на интерфейсе 0/2:

```
configure
interface 0/2
vlan participation include 10,20,40
exit
exit
```

В правила участия включены VLAN 10, 20, 40 и 1 ("vlan participation include 1" является настройкой по умолчанию и не отображается в конфигурации). Пакет имеет два тега VLAN: внешний 40 и внутренний 10. Обработка идет по внешнему тегу. Пакет с внешним VLAN 40 пройдет проверку. Переходим по стрелке "yes" от "Egress filter" к условию "Is Rx port CPU and Tx Port NNI".

Условие "Is Rx port CPU and Tx Port NNI" проверяет, является ли порт, с которого пришел пакет, CPU-портом, и включен ли режим "mode dvlan-tunnel nni" на любом из интерфейсов коммутатора.

Конфигурация:

```
configure
interface 0/1
mode dvlan-tunnel uni
exit
interface 0/2
mode dvlan-tunnel nni
exit
exit
```

Настройка "mode dvlan-tunnel nni" присутствует на интерфейсе 0/2, значит этот интерфейс является NNI-интерфейсом. Пакет пришел с интерфейса 0/1, который не является CPU-интерфейсом. Общее условие выполнено не будет, переходим по стрелке "no" от условия "Is Rx port CPU and Tx Port NNI" к условию "Is Tx port tagged"

В условии "Is Tx port Tagged" проверяется, есть ли настройка сохранения тега на выходном интерфейсе:

```
configure
interface 0/2
vlan tagging 10,20,40
exit
exit
```

В настройках указано, что для 10, 20, 40 VLAN будет сохранен внешний VLAN тег на выходе из интерфейса. Условие выполняется. Переходим по стрелке "yes" от условия "Is Tx port Tagged" к блоку "Is Rx port NNI or UNI".

Условие "Is Rx port NNI or UNI" проверяет, является ли интерфейс, с которого был принят пакет, UNI- или NNI-интерфейсом. Пакет был принят с интерфейса 0/1.

Текущая конфигурация:

```
configure
interface 0/1
mode dvlan-tunnel uni
exit
interface 0/2
mode dvlan-tunnel nni
exit
exit
```

На интерфейсе 0/2 есть настройка "mode dvlan-tunnel nni", значит интерфейс 0/2 является NNI-интерфейсом. На интерфейсе 0/1 есть настройка "mode dvlan-tunnel uni", значит интерфейс 0/1 является UNI-интерфейсом. Переходим по стрелке "yes" из условия "Is Rx port NNI or UNI" в условие "Transmit".

В блоке "Transmit" происходит отправка пакета с текущим набором тегов VLAN, пакет будет отправлен с внешним тегом VLAN 40 и внутренним 10.

Модель услуг C-VLAN

Согласно модели C-VLAN каждый сервис доставляется клиенту в отдельном VLAN. Это может быть доступ в интернет, IPTV, VoIP.

Шаг 1. Настройка VLAN управления

В первую очередь настроим VLAN управления на коммутаторе. Обычно VLAN управления идет с одной меткой, без добавления S-VLAN. В примере VLAN управления имеет значение 600, номер uplink-интерфейса — 0/28.

Схема прохождения пакетов будет следующей:

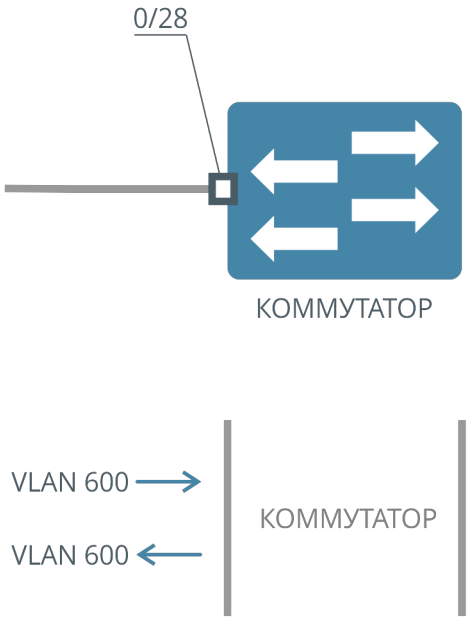

Рисунок 30. Схема прохождения управляющих пакетов во VLAN 600

Создаем VLAN 600, указываем VLAN 600 как управляющий, настраиваем правила участия на uplink-интерфейсе.

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 600
(als_sw) (Vlan) #exit
(als_sw) #network mgmt_vlan 600
(als_sw) #configure
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #vlan participation include 600
(als_sw) (configure) (interface 0/28) #vlan tagging 600
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
(als_sw) #
```

Шаг 2. Настройка UNI- и NNI-интерфейсов

В примере клиенту 1 нужно предоставить доступ в интернет во VLAN 10, клиенту 2 — во VLAN 20. Все клиентские VLAN должны выходить с коммутатора с S-VLAN 100, то есть трафик от клиента 1 должен выходить с uplink-интерфейса коммутатора с S-VLAN 100, C-VLAN 10, от клиента 2 — с S-VLAN 100, C-VLAN 20.

Схема прохождения пакетов представлена ниже:

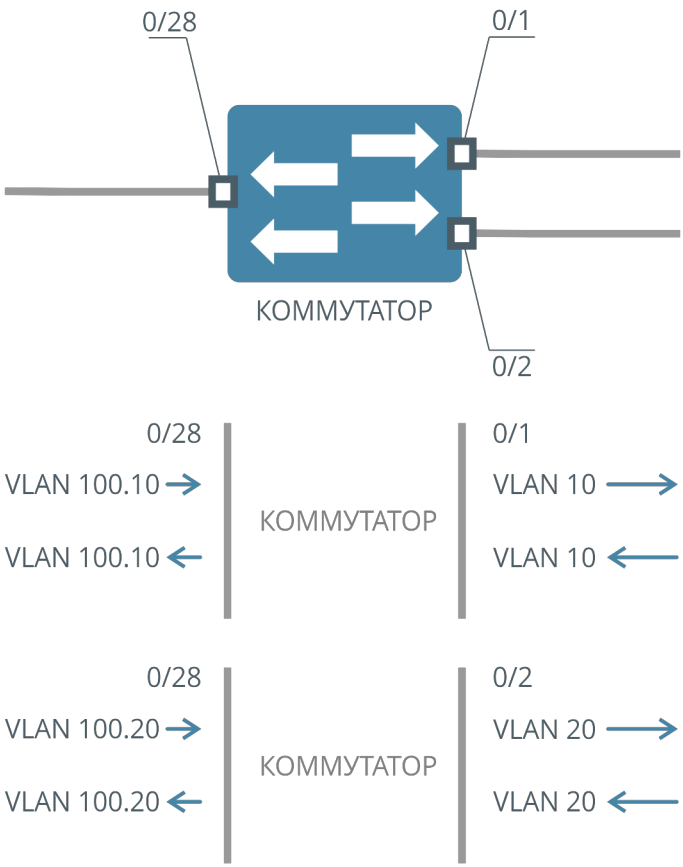

Рисунок 31. Схема прохождения клиентских пакетов

За интерфейсом 0/28 расположена сеть провайдера, где должны проходить пакеты с двойным тегом, значит интерфейс 0/28 будет NNI-интерфейсом, остальные UNI-интерфейсами. После назначения первого NNI-интерфейса, остальные интерфейсы автоматически станут UNI-интерфейсами.

Конфигурация:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #mode dvlan-tunnel
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
(als_sw) #
```

Шаг 3. Запрещение прохождения не тегированного трафика

Так как используется модель услуг C-VLAN, запрещаем нетегированный трафик на абонентских интерфейсах и uplink-интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1-0/2,0/28
(als_sw) (configure) (interface 0/1-0/2,0/28) #vlan acceptframe vlanonly
(als_sw) (configure) (interface 0/1-0/2,0/28) #exit
(als_sw) (configure) #exit
(als_sw) #
```

Также следует помнить, что на интерфейсах по умолчанию включена команда "vlan participation include 1", которая не отображается в конфигурации. Исключим VLAN 1 из правил участия:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1-0/2,0/28
(als_sw) (configure) (interface 0/1-0/2,0/28) #vlan participation exclude 1
(als_sw) (configure) (interface 0/1-0/2,0/28) #exit
(als_sw) (configure) #exit
(als_sw) #
```

Шаг 4. Настройка VLAN на клиентских интерфейсах

Пакеты во VLAN 10 и 20, от клиента 1 и 2 должны получать S-VLAN 100. Данное поведение можно реализовать с помощью Port-based Q-in-Q.

Конфигурация:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 100
(als_sw) (Vlan) #exit
(als_sw) #configure
(als_sw) (configure) #interface 0/1-0/2
(als_sw) (configure) (interface 0/1-0/2) #vlan pvid 100
(als_sw) (configure) (interface 0/1-0/2) #vlan participation include 100
(als_sw) (configure) (interface 0/1-0/2) #exit
(als_sw) (configure) #exit
(als_sw) #
```

Шаг 5. Настройка VLAN на uplink-интерфейсе

На выходе интерфейса 0/28 настроим правило участия и правило сохранения тега для VLAN 100:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #vlan participation include 100
(als_sw) (configure) (interface 0/28) #vlan tagging 100
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
(als_sw) #
```

Шаг 6. Настройка VLAN для IPTV на клиентских интерфейсах

В примере IPTV должно идти к клиентам во VLAN 200. Причем VLAN 200 не должен иметь дополнительных тегов S-VLAN.

Схема прохождения трафика IPTV представлена ниже:

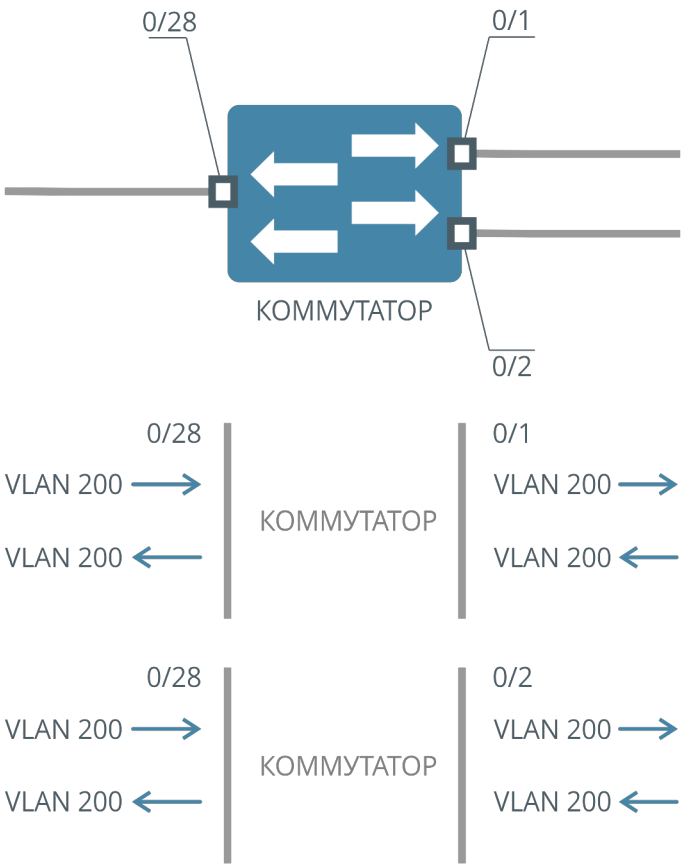

Рисунок 32. Схема прохождения трафика IPTV

Так как интерфейсы 0/1 и 0/2 являются UNI-интерфейсами, то любой пакет получит S-VLAN 100. Чтобы предотвратить назначение S-VLAN, воспользуемся Selective Q-in-Q. Все пакеты, приходящие во VLAN 200 от клиентов, будем помещать в S-VLAN 200. Так как по схеме на uplink NNI-интерфейс 0/28 приходят пакеты с одним VLAN 200, то на UNI-интерфейсах 0/1 и 0/2 нужно настроить правило сохранения тега VLAN 200.

Конфигурация:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 200
(als_sw) (Vlan) #exit
(als_sw) #configure
(als_sw) #dvlan selective
(als_sw) (configure) #interface 0/1-0/2
(als_sw) (configure) (interface 0/1-0/2) #dvlan cvid 200 svid 200
(als_sw) (configure) (interface 0/1-0/2) #vlan participation include 200
(als_sw) (configure) (interface 0/1-0/2) #vlan tagging 200
(als_sw) (configure) (interface 0/1-0/2) #exit
(als_sw) (configure) #exit
(als_sw) #
```

Шаг 7. Настройка VLAN для IPTV на uplink-интерфейсе

Настроим правила участия для VLAN 200 на uplink-интерфейсе. Так как с downlink-интерфейсов будут приходить пакеты с двумя тегами VLAN: S-VLAN 200 и C-VLAN 200, то настраивать правило сохранения тега на интерфейсе 0/28 не будем, чтобы пакет вышел с одним тегом C-VLAN 200.

Конфигурация:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #vlan participation include 200
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
(als_sw) #
```

Шаг 8. Проверка настроек

Текущая конфигурация коммутатора после всех настроек будет выглядеть следующим образом:

```
(als_sw) #show running-config
network mgmt_vlan 600
vlan database
vlan 100,200,600
exit
configure
dvlan selective
interface 0/1
mode dvlan-tunnel uni
vlan acceptframe vlanonly
dvlan cvid 200 svid 200
vlan pvid 100
vlan participation exclude 1
vlan participation include 100,200
vlan tagging 200
exit
interface 0/2
mode dvlan-tunnel uni
vlan acceptframe vlanonly
dvlan cvid 200 svid 200
vlan pvid 100
vlan participation exclude 1
vlan participation include 100,200
vlan tagging 200
exit
interface 0/3
mode dvlan-tunnel uni
exit
...
interface 0/27
mode dvlan-tunnel uni
exit
interface 0/28
mode dvlan-tunnel nni
vlan acceptframe vlanonly
vlan participation exclude 1
vlan participation include 100,200,600
vlan tagging 100,600
exit
exit
```

Добавление двух тегов 802.1q для абонентских устройств

В некоторых случаях трафику необходимо назначить сразу две метки VLAN на одном коммутаторе, которые должны будут добавиться к трафику без тега. К примеру, на порт коммутатора поступает трафик без тега, который должен выйти с коммутатора в двойном теге 10.30. Обратный трафик соответственно наоборот: приходит на коммутатора в двойном теге 10.30, выйти должен без тегов.

Шаг 1. Создание необходимых VLAN

Создаем необходимые для прохождения трафика VLAN:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 10,30
```

Шаг 2. Настройка правил участия VLAN

Настраиваем правила участия VLAN. Коммутатор обрабатывает правила участия по внешнему VLAN, в нашем случае это VLAN 10:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan participation include 10
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #interface 0/18
(als_sw) (configure) (interface 0/18) #vlan participation include 10
```

Шаг 3. Включаем режим Q-in-Q и назначаем роли интерфейсов UNI и NNI

Назначаем интерфейс 0/18 NNI-интерфейсом, это автоматически сделает остальные интерфейсы UNI-интерфейсами:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/18
(als_sw) (configure) (interface 0/18) #mode dvlan-tunnel
```

Шаг 4. Настройка сохранения внешнего тега на NNI-интерфейсе

Добавляем правило сохранения внешнего тега на NNI интерфейсе. В нашем случае VLAN 10 является внешним:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/18
(als_sw) (configure) (interface 0/18) #vlan tagging 10
```

Шаг 5. Настройка добавления внутреннего тега на NNI-интерфейсе

Настраиваем правило добавления внутреннего VLAN, согласно схеме прохождения трафика это VLAN 30. На входе в интерфейс 0/1 будет добавлена внутренняя метка VLAN 30.

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan inner insert 30
```

Шаг 6. Настройка удаления метки внутреннего VLAN

Настраиваем правило удаления внутреннего VLAN, чтобы трафик с интерфейса 0/1 выходил без тегов:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan inner remove
```

Шаг 7. Проверка настроек

Конфигурация коммутатора после всех настроек будет выглядеть следующим образом:

```
(als_sw) #show running-config
vlan database
vlan 10,30
exit
configure
interface 0/1
mode dvlan-tunnel uni
vlan pvid 10
vlan participation include 10
vlan inner insert 30
vlan inner remove
exit
interface 0/2
mode dvlan-tunnel uni
exit
...
interface 0/17
mode dvlan-tunnel uni
exit
interface 0/18
mode dvlan-tunnel nni
vlan participation include 10
vlan tagging 10
exit
exit
```

В итоге трафик без тега, приходящий на 0/1 интерфейс, будет получать 2 тега (внешний 10 и внутренний 30), и выходить с 0/18 интерфейса. Трафик с двумя тегами 10.30, приходящий на 0/18 интерфейс, будет терять внешний и внутренний тег при выходе с 0/1.

ГЛАВА 13. ПРОТОКОЛЫ SPANNING TREE

13.1. Введение

Протокол STP

STP (англ. Spanning Tree Protocol, протокол связующего дерева) — сетевой протокол, работающий на канальном уровне модели OSI и предназначенный для приведения сети с множественными связями к древовидной топологии, исключающей кольцевые соединения. Протокол STP описан в стандарте IEEE 802.1D.

Построение древовидной топологии (дерева) происходит путем автоматического блокирования избыточных связей коммутаторами. Преднамеренное добавление избыточных связей и включение протокола STP используется для создания отказоустойчивых сетей. Если в сети разрывается какая-либо активная в данный момент связь, то коммутаторы обнаруживают это и задействуют заблокированные связи для восстановления дерева. Процесс восстановления дерева называется схождением, а время, за которое сеть восстанавливается — временем сходимости. Время сходимости протокола STP составляет порядка 1 минуты.

Суть работы протокола заключается в том, что поддерживающие его коммутаторы сети Ethernet обмениваются друг с другом информацией о себе. На основании определенных условий (обычно в соответствии с настройками) один из коммутаторов выбирается корневым (Root), после чего все остальные коммутаторы по алгоритму связующего дерева выбирают для работы интерфейсы, ближайшие к корневому коммутатору (учитывается количество посредников и скорость линий). Все прочие сетевые интерфейсы, ведущие к корневому коммутатору, блокируются. Таким образом формируется несвязное дерево с корнем в выбранном коммутаторе.

Основные понятия:

- BPDU (Bridge Protocol Data Unit) — фрейм (единица данных) протокола управления сетевыми мостами;
- Bridge Priority — приоритет коммутатора. Чем меньше значение, тем выше приоритет. Если у коммутаторов одинаковые приоритеты, то приоритет коммутатора определяется по MAC-адресу. Приоритет коммутатора задается множителем 4096xN, где N — число от 0 до 15. По умолчанию N = 8, то есть приоритет равен 32768 (80:00 в

шестнадцатеричном представлении);
- Bridge Identifier — идентификатор коммутатора. Складывается из приоритета коммутатора и MAC-адреса коммутатора. Для примера у коммутатора с MAC-адресом 12:34:56:78:90:12 и приоритетом 32768 идентификатор будет равен 80:00:12:34:56:78:90:12;
- Designated Root — идентификатор назначенного корневого коммутатора;
- Root Path Cost — стоимость кратчайшего пути до корневого коммутатора;
- Root Port Identifier — идентификатор корневого интерфейса;
- Bridge Max Age — максимальный возраст отправляемых сообщений (в секундах). Меняется в зависимости от настроек корневого коммутатора;
- Bridge Forwarding Delay — время перехода коммутатора в новое состояние. Устанавливается корневым коммутатором;
- Hello Time — определяет период отправки конфигурационных сообщений. Значение устанавливается корневым коммутатором.

Основные понятия, применимые к интерфейсу:

- Port Priority — приоритет интерфейса. Задается множителем 16*N, где N — число от 0 до 15. По умолчанию N = 8, то есть приоритет равен 128 (80 в шестнадцатеричной системе счисления);
- Port Identifier — идентификатор интерфейса, который складывается из приоритета и номера интерфейса. По умолчанию для интерфейса 0/1 это значение будет равно 80:01, где 80 — приоритет по умолчанию в шестнадцатеричной системе (128 в десятичной), 01 — номер интерфейса;
- Port Forwarding State — состояние интерфейса. Порт может находиться в 5 состояниях: блокировки (blocking), прослушивания (listening), обучения (learning), передачи (forwarding) и в выключенном состоянии (disabled);
- Port Role — роль порта. Может быть в одном из четырех статусов: корневой (Root), назначенный (Designated), альтернативный (Alternate) и резервный (Backup);
- Port Path Cost — стоимость пути соединения на интерфейсе;
- Designated Root и Designated Bridge — идентификаторы корневого коммутатора и назначенного коммутатора для данного интерфейса соответственно;
- Root Path Cost — стоимость пути до корневого коммутатора на данном интерфейсе;
- Designated Port Identifier — идентификатор назначенного интерфейса.

Выбор корневого коммутатора происходит по следующему алгоритму:

1. Выбирается коммутатор с наименьшим значением приоритета
2. Если приоритеты у коммутаторов совпадают, то выбирается коммутатор с меньшим MAC-адресом
3. Если приоритет и MAC-адрес совпадают, то считается, что пакет пришел с этого же коммутатора и один из интерфейсов коммутатора станет резервным

Протокол RSTP

RSTP (англ. Rapid STP, быстрый STP) — протокол быстрого построения древовидной топологии. Главное отличие от STP — намного меньшее время сходимости, которое составляет порядка 6 секунд. В RSTP вводится понятие Edge Port (граничный интерфейс). Граничный интерфейс сразу начинает работать в состоянии forwarding и считает что за ним нет коммутаторов. Если на граничный интерфейс приходит BPDU, то он начинает работать как обычный STP или RSTP интерфейс. По умолчанию на коммутаторе используется протокол RSTP. Если на какой-либо интерфейс приходит STP BPDU, то интерфейс начинает работать в режиме STP.

Протокол MSTP

MSTP (англ. Multiple Spanning Tree) — версия протокола STP для построения древовидной топологии. MSTP по логике работы схож с RSTP, однако MSTP позволяет построить разные топологии сети в разных VLAN. Это позволяет повысить отказоустойчивость сети, а также распределить нагрузку. Также данный протокол позволяет разбить сегменты сети на регионы. В каждом регионе будет своя топология и свой root коммутатор.

13.2. Настройка Spanning Tree

В ходе настройки мы будем использовать типовую схему, изображенную на рисунке ниже:

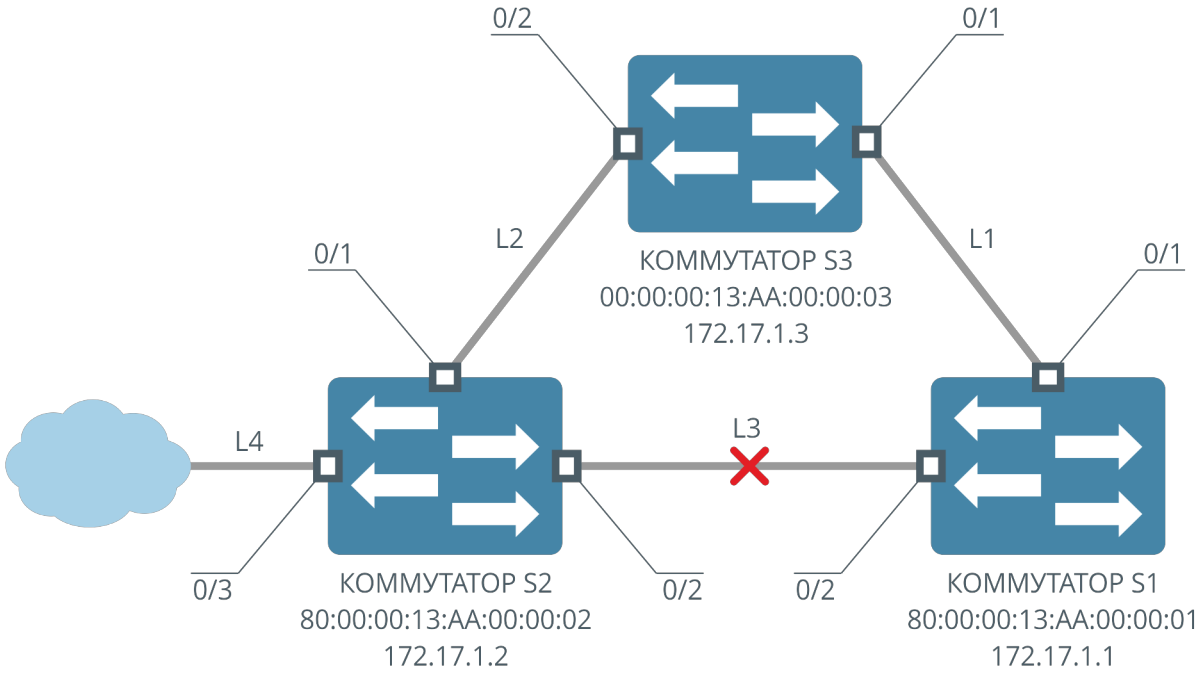

Рисунок 33

После выполнения шагов по настройке коммутаторы будут работать следующим образом:

- коммутатор S3 станет корневым, трафик будет идти к коммутатору S1 по соединению L1, к коммутатору S2 по соединению L2;
- интерфейсы 0/1 и 0/2 коммутатора S3 будут иметь Role = Designated;
- соединение L3 будет неактивно, трафик по нему ходить не будет;
- соединение L4 будет активно;
- интерфейсы 0/1 коммутаторов S1 и S2 будут иметь Role = Root;
- интерфейс 0/2 коммутатора S1 будет иметь Role = Designated;
- интерфейс 0/2 коммутатора S2 будет иметь Role = Alternate;
- интерфейс 0/3 коммутатора S2 будет иметь Role = Designated.

Настройка службы Spanning Tree производится в несколько шагов. Сначала необходимо включить службу глобально и на интерфейсах. Затем нужно настроить приоритет коммутатора. Потом, в зависимости от требуемой топологии, необходимо настроить дополнительные параметры.

Шаг 1. Предварительная настройка

Для того чтобы коммутаторы корректно работали и их легко было различать в процессе настройки, необходимо произвести следующие предварительные настройки:

На коммутаторе S1 настраивается IP-адрес 172.17.1.1 и устанавливается приглашение als_sw1:

```
(als_sw) #set prompt als_sw1
(als_sw1) #network parms 172.17.1.1 255.255.0.0
```

На коммутаторе S2 настраивается IP-адрес 172.17.1.2 и устанавливается приглашение als_sw2:

```
(als_sw) #set prompt als_sw2
(als_sw2) #network parms 172.17.1.2 255.255.0.0
```

На коммутаторе S3 настраивается IP-адрес 172.17.1.3 и устанавливается приглашение als_sw3:

```
(als_sw) #set prompt als_sw3
(als_sw3) #network parms 172.17.1.3 255.255.0.0
```

Шаг 2. Включение Spanning Tree на коммутаторах

Для того чтобы служба Spanning Tree заработала на коммутаторе, ее необходимо включить. Для включения используется команда:

```
(als_sw) #configure
(als_sw) (configure) #spanning-tree
```

Для работы примера необходимо включить службу Spanning Tree на всех трех коммутаторах.

Шаг 3. Включение Spanning Tree на интерфейсах

Для того чтобы интерфейс стал участвовать в построении дерева, необходимо включить на нем службу Spanning Tree. Для включения используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #spanning-tree port mode
```

Для работы примера необходимо включить службу Spanning Tree на интерфейсах 0/1-0/2 на всех трех коммутаторах, а также на интерфейсе 0/3 коммутатора S2.

Шаг 4. Выбор версии протокола

По умолчанию на коммутаторе включен протокол MSTP. Для принудительного переключения версии протокола на RSTP используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #spanning-tree forceversion 802.1w
```

Для переключения версии протокола на STP используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #spanning-tree forceversion 802.1d
```

Для переключения версии протокола на MSTP используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #spanning-tree forceversion 802.1s
```

Шаг 5. Добавление MSTP instance (опционально)

По умолчанию на коммутаторе создан 0 instance, в который включены все созданные VLAN. Для примера создадим instance 1 и добавим в него новые VLAN: 100 и 200.

Создаем instance 1

```
(als_sw) #configure
(als_sw) (configure) #spanning-tree mst instance 1
```

Создаем VLAN на коммутаторе:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 100,200
```

Добавляем VLAN 100, 200 в instance 1

```
(als_sw) #configure
(als_sw) (configure) #spanning-tree mst vlan 1 100,200
```

Шаг 6. Настройка региона MSTP (опционально)

Для того, чтобы все коммутаторы были в одном регионе, необходимо, чтобы на всех трех коммутаторах были одинаковые настройки "configuration name" и "revision", а также одинаковое распределение VLAN по instance.

Установим "configuration name":

```
(als_sw) #configure
(als_sw) (configure) #spanning-tree configuration name "test1"
```

Установим "revision":

```
(als_sw) #configure
(als_sw) (configure) #spanning-tree configuration revision 15
```

Шаг 7. Установка приоритета коммутатора

Для того чтобы управлять топологией дерева, а в частности выбрать корневой коммутатор, меняют приоритет коммутатора. Сделать это можно следующей командой:

```
(als_sw) #configure
(als_sw) (configure) #spanning-tree mst priority <instance> <priority>
```

- <instance> — номер MSTP instance, в котором назначается приоритет (для версий STP и RSTP указывается как 0);
- <priority> — приоритет коммутатора.

Для работы примера необходимо на коммутаторе S3 установить приоритет 0.

```
(als_sw) #configure
(als_sw) (configure) #spanning-tree mst priority 0 0
```

Шаг 8. Установка приоритета интерфейса (опционально)

Если стоимость пути до корневого коммутатора совпадает для разных интерфейсов, то приоритет имеет интерфейс, у которого меньше Designated Bridge ID. Если данные ID совпадают, то приоритетный порт выбирается по Designated Port ID. Если и они совпадают, то интерфейсы выбираются по приоритету интерфейса. В конечном итоге, если и приоритеты интерфейсов совпадают, то выбирается интерфейс с меньшим номером.

Для того чтобы установить приоритет интерфейса, используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #spanning-tree mst <instance> port-priorit
y <priority>
```

- <instance> — номер MSTP instance, в котором назначается приоритет (для версий STP и RSTP указывается как 0);
- <priority> — приоритет порта коммутатора.

Шаг 9. Установка максимального возраста сообщений (опционально)

На корневом коммутаторе можно установить максимальный возраст отправляемых сообщений, тогда все остальные коммутаторы в дереве изменят свои настройки максимального возраста в соответствии с установленными на корневом коммутаторе. Сделать это можно следующей командой:

```
(als_sw) #configure
(als_sw) (configure) #spanning-tree max-age <max-age>
```

Также на корневом коммутаторе можно установить время перехода коммутатора в новое состояние, тогда все остальные коммутаторы в дереве изменят свои настройки времени перехода в соответствии с установленными на корневом коммутаторе. Сделать это можно с помощью следующей команды:

```
(als_sw) #configure
(als_sw) (configure) #spanning-tree forward-time <forward-time>
```

Шаг 10. Установка ограничения на количество отправляемых пакетов (опционально)

Иногда при построении топологии происходит слишком много изменений и с интерфейсов может выходить много BPDU-пакетов. Для того, чтобы ограничить количество BPDU-пакетов, выходящих с интерфейса, используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #spanning-tree hold-count <hold-count>
```

Hold Count устанавливает максимальное значение счетчика. При каждом отправленном BPDU значение счетчика уменьшается на единицу. Если значение счетчика стало равным нулю, то с данного интерфейса BPDU больше не отправляются. Раз в секунду значение счетчика увеличивается на единицу.

Шаг 11. Установка граничного интерфейса (опционально)

Чтобы интерфейс сразу начинал принимать и отправлять пакеты, если на данный интерфейс не приходят BPDU, его можно сделать граничным. Для этого используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #spanning-tree edgeport
```

Для работы примера необходимо на коммутаторе S2 сделать интерфейс 0/3 граничным.

Шаг 12. Установка BPDU фильтра (опционально)

Для того, чтобы интерфейс отбрасывал приходящие на него BPDU, а также не отправлял BPDU, используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #spanning-tree bpdufilter
```

Шаг 13. Установка стоимости соединения (опционально)

По умолчанию стоимость соединения определяется автоматически в зависимости от скорости интерфейса согласно стандарту. Для того чтобы установить стоимость соединения вручную, используется следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #spanning-tree mst <instance> cost <cost>
```

- <instance> — номер MSTP instance, в котором назначается приоритет (для версий STP и RSTP указывается как 0);
- <cost> — стоимость пути до интерфейса.

Протокол MSTP использует две стоимости соединения: внешнюю, между регионами, и внутреннюю в пределах рагиона.

Указанная команда изменяет обе стоимости. Для того, что бы изменить только внешнюю стоимость:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #spanning-tree mst 0 external-cost <cost>
```

Для того чтобы стоимость соединения определялась автоматически, используется следующие команды:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #spanning-tree mst <instance> cost auto
```

- <instance> — номер MSTP instance, в котором назначается приоритет (для версий STP и RSTP указывается как 0).

Шаг 14. Настройка фильтрации TCN-сообщений (опционально)

Опционально на портах коммутатора можно включить фильтрацию сообщений topology change notification. Фильтрация включается на портах, ведущих к корневому коммутатору.

Фильтрация сообщений возможна на входе порта:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #spanning-tree tcnguard rx
```

В этом случае при приходе TCN-пакета на интерфейс 0/1 сообщение TCN не будет распространено на другие порты коммутатора. Однако счетчик "Topology Change Count" будет увеличен.

Фильтрация сообщений возможна на выходе порта:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #spanning-tree tcnguard tx
```

В этом случае при приеме сообщения TCN на другом интерфейсе коммутатор не будет отправлять через интерфейс 0/1 сообщение TCN.

Просмотр состояния коммутатора

Вывод команд будет представлен в соответствии с построенным деревом из примера. Ниже представлена конфигурация для всех коммутаторов.

Коммутатор S1:

```
(als_sw1) #show running-config
set prompt "als_sw1"
network parms 172.17.1.1 255.255.0.0 0.0.0.0
spanning-tree
interface 0/1
spanning-tree port mode
exit
interface 0/2
spanning-tree port mode
exit
exit
```

Коммутатор S2:

```
(als_sw2) #show running-config
set prompt "als_sw2"
network parms 172.17.1.2 255.255.0.0 0.0.0.0
spanning-tree
interface 0/1
spanning-tree port mode
exit
interface 0/2
spanning-tree port mode
interface 0/3
spanning-tree port mode
spanning-tree edgeport
exit
exit
```

Коммутатор S3:

```
(als_sw2) #show running-config
set prompt "als_sw3"
network parms 172.17.1.3 255.255.0.0 0.0.0.0
spanning-tree
spanning-tree mst priority 0 0
interface 0/1
spanning-tree port mode
exit
interface 0/2
spanning-tree port mode
exit
exit
```

Для просмотра состояния службы Spanning Tree на коммутаторе используется следующая команда:

```
(als_sw) #show spanning-tree

Bridge Priority.............................. 32768
Bridge Identifier............................ 80:00:00:13:AA:00:00:01
Time Since Topology Change................... 0 day 0 hr 12 min 14 sec
Topology Change Count........................ 1
Topology Change in progress.................. FALSE
Designated Root.............................. 00:00:00:13:AA:00:00:03
Root Path Cost............................... 200000
Root Port Identifier......................... 80:01
Bridge Max Age............................... 20
Bridge Max Hops.............................. 20
Bridge Tx Hold Count......................... 6
Bridge Forwarding Delay...................... 15
Hello Time................................... 2
CST Regional Root............................ 00:00:00:13:AA:00:00:03
Regional Root Path Cost...................... 0
```

Из вывода команды после применения ее на коммутаторе S1 видно, что коммутатор S3 является корневым и Root интерфейсом коммутатора S1 является интерфейс 0/1.

Для просмотра состояния службы Spanning Tree на интерфейсе коммутатора используется следующая команда:

```
(als_sw) #show spanning-tree mst port detailed 0 0/3

Port Identifier.............................. 80:03
Port Priority................................ 128
Port Forwarding State........................ Forwarding
Port Role.................................... Designated
Root Guard Status............................ Disabled
Root Guard State............................. Forwarding
Auto-calculate External Port Path Cost....... Enabled
External Port Path Cost...................... 200000
Auto-calculate Port Path Cost................ Enabled
Port Path Cost............................... 200000
Designated Root.............................. 00:00:00:13:AA:00:00:03
Root Path Cost............................... 200000
Designated Bridge............................ 80:00:00:13:AA:00:00:02
Designated Port Identifier................... 80:03
Edge Port.................................... TRUE
Edge Port Status............................. TRUE
CST Regional Root............................ 00:00:00:13:AA:00:00:03
CST Internal Root Path Cost.................. 200000
```

Из вывода команды после применения ее на интерфейсе 0/3 коммутатора S2 видно, что коммутатор S3 является корневым, а интерфейс 0/3 коммутатора S2 является граничным.

ГЛАВА 14. СПИСКИ КОНТРОЛЯ ДОСТУПА (ACL)

14.1. Введение в списки контроля доступа

Списки контроля доступа — общее название механизма, обеспечивающего задание правил доступа к объектам. В сетевых технологиях данный термин служит для определения механизма, обеспечивающего задание правил прохождения трафика через сетевое устройство и до него (коммутатор, маршрутизатор, firewall и другие). Далее данный термин используется в контексте сетевых технологий.

Назначение списков контроля доступа

В первую очередь списки контроля доступа используются для ограничения прохождения нежелательного либо паразитного трафика в сети, ограничения доступа к различным узлам для обеспечения их безопасности. Также ACL может использоваться для ограничения доступа к сервисам внутри сети. К примеру, ACL может быть использован для запрета подключения к определенному порту сервера из определенной IPv4 подсети.

Принципы работы списков контроля доступа

Список контроля доступа представляет из себя набор правил, применяющихся к трафику в последовательности, определяемой приоритетом. Правило списка контроля доступа должно содержать в себе тип трафика и действие, применяемое к данному типу трафика: разрешение либо запрет на прохождение. Трафик классифицируется на типы по полям, содержащимся в заголовке пакета.

14.2. Настройка ACL на коммутаторах АЛСиТЕК

Общие концепции конфигурирования

Правила ACL условно можно разделить на четыре больших группы:

- MAC ACL — позволяет задавать правила на основе Src/Dst MAC, Ethertype, а также на основе метки 802.1p (CoS) и VLAN;
- IPv4 ACL — позволяет задавать правила на основе Src/Dst IPv4, Src/Dst L4 Port, а также на основе полей IP Protocol, TOS и DSCP;
- IPv6 ACL — позволяет задавать правила на основе Src/Dst IPv6-адресов, а также на основе полей NextHeader IP Protocol, Traffic Class и DSCP;
- User-defined ACL — позволяет задавать правила на основе собственных шаблонов; применяется в случаях, когда встроенных средств MAC/IPv4/IPv6 ACL не хватает для фильтрации пакета.

Все настройки ACL можно применять на интерфейсах коммутатора. Для IPv4 и IPv6 ACL допустимо применение на VLAN и на интерфейсе управления.

Создание списков контроля доступа

Имя создаваемого списка доступа должно быть длиной от 1 до 32 символов, разрешенные символы: [A-Za-z0-9,._]. После ввода команды создания нового списка доступа осуществляется автоматический переход в контекст добавления правил ACL. Редактирование списка доступа запрещено, поэтому выход из данного контекста должен осуществляться только после добавления всех необходимых правил.

Добавление правил в список контроля доступа

В контексте добавления правил ACL осуществляется добавление всех необходимых правил, определяющих прохождение трафика. Все правила ACL применяются исключительно к входящему трафику.

Правила в списке доступа делятся на разрешающие (permit) и запрещающие (deny). Правила применяются в порядке их объявления в списке доступа, что позволяет задавать сначала частные правила, затем общие. Корректировать отдельные правила после их объявления в списке доступа нельзя, необходимо удалить список полностью и затем создать новый с необходимым набором правил. В одном списке может быть не более 24 правил.

Основные правила применения списков доступа:

- списки доступа применяются в соответствии с приоритетом;
- наименьшее число в параметре "приоритет" соответствует наивысшему приоритету;
- при добавлении нового списка доступа без указания приоритета ему автоматически назначается приоритет, следующий за максимальным

приоритетом среди уже примененных списков;
- если при добавлении списка доступа на интерфейс с указанием приоритета на интерфейсе уже есть список с таким же приоритетом, он будет заменен на добавляемый;
- правила внутри списка доступа применяются последовательно в порядке их добавления;
- при применении нескольких списков доступа на интерфейсе, в первую очередь будут применены правила, входящие в список доступа с наивысшим приоритетом, а затем правила, входящие в список доступа с более низким приоритетом;
- если пакет попал под конкретное правило списка доступа, проверка прекращается, пакет признается разрешенным либо запрещенным, в зависимости от типа правила (permit или deny);
- если пакет при проверке не подошел ни к одному правилу, он будет отброшен.

Правила ACL имеют вид (на примере IPv4 ACL):

```
permit ip 172.17.1.1 0.0.0.0 172.17.1.4 0.0.0.0
permit ip 192.168.1.1 0.0.0.0 192.168.1.4 0.0.0.0
```

Одна строка — это одно правило IPv4 ACL. Рассмотрим пакет, подходящий под следующее правило:

```
permit ip 172.17.1.1 0.0.0.0 172.17.1.4 0.0.0.0
```

Это должен быть IPv4-пакет с адресом источника 172.17.1.1 и адресом назначения 172.17.1.4. То есть все атрибуты пакета, упомянутые в правиле, должны соответствовать указанным в правиле значениям.

Правила взаимодействия списков контроля доступа

Применять списки контроля доступа можно и на интерфейсе и на VLAN. При этом:

- первыми будут применены правила из списков на интерфейсах, затем

правила из списков на VLAN, затем неявные запрещающие правила;
- если пакет попадает под какое-то из правил ACL (запрещающее или разрешающее на интерфейсе или на VLAN), дальнейшая проверка пакета прекращается;
- если список не содержит явного разрешающего или запрещающего "правила по умолчанию" (например, *permit ip any any*), то подразумевается наличие неявного запрещающего правила (*deny ip any any* для IPv4 ACL) и оно будет применено после правил из всех списков;
- если список IPv4 или IPv6 ACL применен на VLAN, неявное запрещающее правило будет учитывать этот VLAN и работать только для пакетов в этом VLAN;
- если список IPv4 или IPv6 ACL применен на интерфейсе, неявное запрещающее правило будет работать для всех пакетов.

Подробнее о неявных запрещающих правилах будет рассказано в следующем разделе.

Рассмотрим пример, когда несколько списков контроля доступа (MAC и IPv4) применены на интерфейсе:

```
configure

mac access-list extended "acl4000"
deny any 01:00:01:cc:cc:cd 00:00:00:00:00:00
deny any 01:00:0c:cc:cc:cc 00:00:00:00:00:00
deny any any 0x888e
exit

mac access-list extended "acl4010"
permit any any pppoed
permit any any pppoes
permit any any 0x9000
permit any 01:80:c2:00:00:00 00:00:00:00:00:00
permit any any arp
exit

ip access-list "acl3005"
permit ip 192.168.1.1 0.0.255.255 any
permit ip 11.11.1.1 0.0.0.255 any
permit udp any any eq 67
exit

interface 0/1
description "PPPOE-CLIENT"
mac access-group "acl4000" in 1
ip access-group "acl3005" in 2
mac access-group "acl4010" in 3
exit

exit
```

Чтобы разобраться в данной конфигурации, необходимо выписать списки доступа ACL, примененные на интерфейсе:

```
mac access-group "acl4000" in 1
ip access-group "acl3005" in 2
mac access-group "acl4010" in 3
```

Затем заменить названия списков доступа на конкретные правила из этих списков:

```
! mac access-group "acl4000" in 1
deny any 01:00:01:cc:cc:cd 00:00:00:00:00:00 ! MAC ACL acl4000
deny any 01:00:0c:cc:cc:cc 00:00:00:00:00:00 ! MAC ACL acl4000
deny any any 0x888e ! MAC ACL acl4000

! ip access-group "acl3005" in 2
permit ip 192.168.1.1 0.0.255.255 any ! IPv4 ACL acl3005
permit ip 11.11.1.1 0.0.0.255 any ! IPv4 ACL acl3005
permit udp any any eq 67 ! IPv4 ACL acl3005

! mac access-group "acl4010" in 3
permit any any pppoed ! MAC ACL acl4010
permit any any pppoes ! MAC ACL acl4010
permit any any 0x9000 ! MAC ACL acl4010
permit any 01:80:c2:00:00:00 00:00:00:00:00:00 ! MAC ACL acl4010
permit any any arp ! MAC ACL acl4010
```

Далее необходимо добавить неявное запрещающее правило:

```
! mac access-group "acl4000" in 1
deny any 01:00:01:cc:cc:cd 00:00:00:00:00:00 ! MAC ACL acl4000
deny any 01:00:0c:cc:cc:cc 00:00:00:00:00:00 ! MAC ACL acl4000
deny any any 0x888e ! MAC ACL acl4000

! ip access-group "acl3005" in 2
permit ip 192.168.1.1 0.0.255.255 any ! IPv4 ACL acl3005
permit ip 11.11.1.1 0.0.0.255 any ! IPv4 ACL acl3005
permit udp any any eq 67 ! IPv4 ACL acl3005

! mac access-group "acl4010" in 3
permit any any pppoed ! MAC ACL acl4010
permit any any pppoes ! MAC ACL acl4010
permit any any 0x9000 ! MAC ACL acl4010
permit any 01:80:c2:00:00:00 00:00:00:00:00:00 ! MAC ACL acl4010
permit any any arp ! MAC ACL acl4010

! неявное запрещающее правило
deny any any
```

Получается список правил в той последовательности, в которой правила будут применены на интерфейсе и будут проверятся. Важно отметить, что если сработает хоть одно из правил, то будет применено его действие "permit" или "deny", соответствие остальным правилам проверяться не будет.

Неявные запрещающие правила

Неявные запрещающие правила не отображаются в конфигурации, однако они добавляются в конце списка ACL автоматически, если не указано явное разрешающее правило.

Неявное запрещающее правило для MAC ACL работает для всех типов пакетов:

```
deny any any
```

Неявное запрещающее правило для IPv4 ACL работает только для IPv4 пакетов:

```
deny ip any any
```

Неявное запрещающее правило для IPv6 ACL работает только для IPv6 пакетов:

```
deny ipv6 any any
```

Неявное запрещающее правило для User-defined ACL работает для всех типов пакетов:

```
deny any
```

Если список доступа применен на VLAN, то неявные запрещающие правила будут работать только для тех VLAN, на которых применен список доступа.

Рассмотрим следующую конфигурацию, демонстрирующую работу неявных запрещающих правил:

```
vlan database
vlan 10
exit

configure

ip access-list "acl7001"
permit ip 172.17.1.4 0.0.0.0 any
permit ip 172.17.1.8 0.0.0.0 any
exit

ip access-list "acl7002"
permit ip 192.168.1.4 0.0.0.0 any
permit ip 192.168.1.8 0.0.0.0 any
exit

ip access-group "acl7002" vlan 10 in 1

interface 0/1
vlan participation include 10
ip access-group "acl7001" in 1
exit

exit
```

Перечень ACL для интерфейса 0/1:

```
ip access-group "acl7001" in 1
ip access-group "acl7002" vlan 10 in 1
```

Заменяем списки доступа на правила:

```
permit ip 172.17.1.4 0.0.0.0 any ! IP ACL acl7001
permit ip 172.17.1.8 0.0.0.0 any ! IP ACL acl7001
permit ip 192.168.1.4 0.0.0.0 any ! IP ACL acl7002 (VLAN 10)
permit ip 192.168.1.8 0.0.0.0 any ! IP ACL acl7002 (VLAN 10)
```

Неявное запрещающее правило для VLAN будет создано после IP ACL, примененного на VLAN. Работать оно будет только для VLAN 10:

```
permit ip 172.17.1.4 0.0.0.0 any ! IP ACL acl7001
permit ip 172.17.1.8 0.0.0.0 any ! IP ACL acl7001
permit ip 192.168.1.4 0.0.0.0 any ! IP ACL acl7002 (VLAN 10)
permit ip 192.168.1.8 0.0.0.0 any ! IP ACL acl7002 (VLAN 10)
deny ip any any ! IP ACL acl7002 (VLAN 10) неявное запрещающее правило для VLAN
 10
```

Общее неявное запрещающее правило для интерфейса 0/1 будет создано после всех правил IP ACL:

```
permit ip 172.17.1.4 0.0.0.0 any ! IP ACL acl7001
permit ip 172.17.1.8 0.0.0.0 any ! IP ACL acl7001
permit ip 192.168.1.4 0.0.0.0 any ! IP ACL VLAN 10
permit ip 192.168.1.8 0.0.0.0 any ! IP ACL VLAN 10
deny ip any any ! IP ACL acl7002 (VLAN 10) неявное запрещающее правило для VLAN
 10
deny ip any any ! IP ACL acl7001 общее неявное запрещающее правило
```

Таким образом, отработают сначала правила, примененные на интерфейсе, затем правила на VLAN, затем общие запрещающие правила.

Списки контроля доступа для интерфейса управления

На устройстве возможно применение ACL на интерфейсе управления. Это означает, что трафик будет фильтроваться на входе в CPU-интерфейс, но эти правила не затрагивают прохождение трафика сквозь устройство.

Основные правила работы списков контроля доступа на интерфейсе управления:

- список контроля доступа на интерфейсе управления работает после ACL на интерфейсе и VLAN. Если ACL на VLAN или интерфейсе заблокировали пакет, то ACL на интерфейсе управления не сможет

разблокировать данный пакет;
- если список контроля доступа на интерфейсе управления заблокировал пакет, то данный пакет может пройти через коммутатор (если не заблокирован другими правилами ACL);
- работа ACL на интерфейсе управления не фильтрует пакеты, предназначенные для следующих механизмов: DHCP Snooping, IGMP Snooping, MLD Snooping, IPv6 Snooping, LDRA.

User-defined ACL

User-defined ACL сделан специально для фильтрации пакетов по полям, готовые шаблоны для которых отсутствуют в MAC/IPv4/IPv6 ACL.

Для User-defined ACL требуется сначала определить поля, по которым будет проходить обработка пакета. Для добавления нового поля в User-defined ACL требуется указать:

- Имя поля — это то имя, по которому можно будет добавлять поле в правила;
- Заголовок — это заголовок пакета, от которого будет отсчитываться смещение. Положение заголовков в кадре может варьироваться от пакета к пакету. К примеру, положение IP-заголовка в пакете с VLAN и без VLAN будет отличаться. Чтобы поле не зависело от нижележащих заголовков, требуется указать, от какого заголовка отсчитывать смещение;
- Смещение — это смещение от начала выбранного заголовка в байтах, по которому находится необходимое поле;
- Длину — это размер поля, по которому будет проходить проверка;

Доступны следующие варианты заголовков, от которых будет отсчитываться смещение:

- raw — смещение отсчитывается от начала пакета (от первого байта Dst MAC-адреса Ethernet-пакета);
- l2 — смещение отсчитывается от Ethertype (для кадров Ethernet II) или Length (для кадров 802.2 и др.);
- l3 — смещение отсчитывается от начала заголовка L3 (IPv4/IPv6, ARP/RARP);
- l4 — смещение отсчитывается от начала заголовка L4 (TCP/UDP/ICMP/IGMP).

Пример создания поля:

```
configure
user-defined access-list template setup
udf "IP_Protocol" offset l3 9 length 2
exit
exit
```

Параметры:

- IP_Protocol — имя поля;
- offset l3 — заголовок, от которого будет отсчитываться смещение;
- 9 — смещение в байтах, в данном случае от начала L3-заголовка;
- length 2 — длина поля — 2 байта.

Созданное один раз поле можно использовать в любом количестве правил ACL.

Настройка MAC ACL

MAC ACL предназначен для фильтрации пакетов по полям Ethernet-заголовка. Для использования доступны следующие поля:

- Source MAC address;
- Destination MAC address;
- VLAN;
- CoS;
- Ethertype.

Шаг 1. Создание списков контроля доступа

Настройка начинается с создания списка доступа с указанием его имени:

```
(als_sw) #configure
(als_sw) (configure) #mac access-list extended "NameACL"
```

После создания списка доступа необходимо создать правила для этого списка.

Шаг 2. Добавление правил в список доступа

Для MAC-адресов в этих командах используются инвертированные (wildcard) маски.

Добавим разрешающее правило для ARP-трафика с MAC-адресами источника 00:13:AA:XX:XX:XX и любыми MAC-адресами назначения:

```
(als_sw) #configure
(als_sw) (configure) #mac access-list extended "NameACL"
(als_sw) (configure) (mac-access-list "NameACL") #permit 00:13:AA:00:00:00 00:0
0:00:FF:FF:FF any 0x0806
```

Параметры:

- permit — ключевое слово — означает то, что данное правило разрешающее;
- 00:13:AA:00:00:00 00:00:00:FF:FF:FF — MAC-адрес источника (Source MAC) с wildcard-маской;
- any — MAC-адрес назначения (Destination MAC) — означает разрешение любых Dst MAC-адресов;
- 0x0806 — Ethertype ARP-пакета.

Добавим запрещающее правило для любого типа трафика с любым MAC-адресом источника и MAC-адресом назначения 00:14:BB:24:CB:4A.

```
(als_sw) (configure) (mac-access-list "NameACL") #deny any 00:14:BB:24:CB:4A 00
:00:00:00:00:00
```

Параметры:

- deny — ключевое слово — означает то, что данное правило запрещающее;
- any — MAC-адрес источника (Source MAC) — означает разрешение любых Src MAC-адресов;
- 00:14:BB:24:CB:4A 00:00:00:00:00:00 — MAC-адрес назначения (Destination MAC) с wildcard-маской.

Разрешающее правило для IPv4 трафика с адресами источника 00:13:AA:XX:XX:XX и MAC-адресами назначения 00:14:BB:24:CB:XX.

```
(als_sw) (configure) (mac-access-list "NameACL") #permit 00:13:AA:00:00:00 00:0
0:00:FF:FF:FF 00:14:BB:24:CB:00 00:00:00:00:00:FF ipv4
```

Параметры:

- permit — ключевое слово — означает то, что данное правило разрешающее;
- 00:13:AA:00:00:00 00:00:00:FF:FF:FF — MAC-адрес источника (Source MAC) с wildcard-маской;
- 00:14:BB:24:CB:00 00:00:00:00:00:FF — MAC-адрес назначения (Destination MAC) с wildcard-маской;
- ipv4 — ключевое слово — означает то, что правило работает для IPv4 трафика (Ethertype 0x0800).

Маска для MAC-адресов в правилах указывается инвертированная (wildcard). Значение 0xFF в маске означает, что на этой позиции в MAC-адресе могут находиться любые значения. При значении 0x00 требуется полное совпадение со значением MAC-адреса на этой позиции.

Шаг 3. Применение списков доступа на интерфейсе

Созданные списки доступа применяются на интерфейсе с указанием приоритета:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #mac access-group "NameACL" in 5
```

Параметры:

- "NameACL" — уникальное имя созданного ранее списка доступа;
- 5 — Приоритет списка доступа на данном интерфейсе, может иметь значение от 1 до 10000, наименьшее число соответствует наивысшему приоритету (необязательный параметр).

Просмотр конфигурации

После выполнения команд из примера получится следующая конфигурация:

```
configure

mac access-list extended "NameACL"
permit 00:13:AA:00:00:00 00:00:00:FF:FF:FF any 0x0806
deny any 00:14:BB:24:CB:4A 00:00:00:00:00:00
permit 00:13:AA:00:00:00 00:00:00:FF:FF:FF 00:14:BB:24:CB:00 00:00:00:00:00:FF
ipv4
exit

interface 0/1
mac access-group "NameACL" in 5
exit

exit
```

Итоговая схема прохождения трафика через устройство будет выглядеть следующим образом:

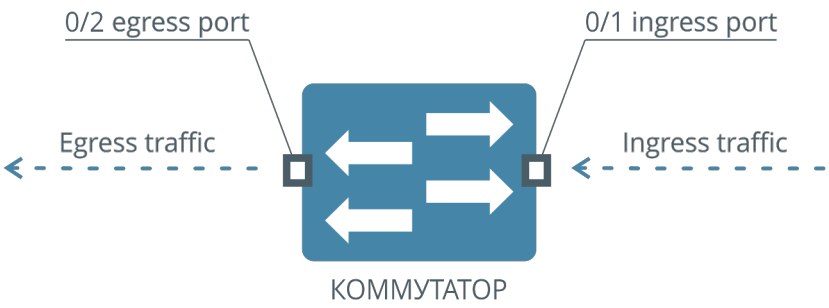

Рисунок 34. Схема прохождения трафика

Source MAC	Destination MAC	Ethertype	Правило	Прохождение
00:13:AA:12:34:60	00:13:AA:12:34:62	ARP (0x0806)	#1 списка "NameACL"	Разрешено
00:13:AA:12:34:60	00:14:BB:24:CB:4A	IPv4 (0x0800)	#2 списка "NameACL"	Запрещено

00:13:AA:00:B0:30	00:14:BB:24:CB:5C	IPv4 (0x0800)	#3 списка "NameACL"	Разрешено
00:13:AA:12:34:60	00:13:AA:12:34:62	IPv6 (0x86DD)	-	Запрещено

При настройке правил MAC ACL есть особенность. К примеру, нужно запретить трафик, описанный определенным правилом (deny), а весь остальной трафик разрешить. Для корректной работы такой настройки нужно после правила deny поместить разрешающее правило для всего остального трафика:

```
configure

mac access-list extended "NameACL"
deny any 00:14:BB:24:CB:00 00:00:00:00:00:FF
permit any any
exit

interface 0/1
mac access-group "NameACL" in 5
exit

exit
```

В данном примере на интерфейсе 0/1 запрещено прохождение трафика с Destination MAC 00:14:BB:24:CB:XX, а весь остальной трафик разрешен.

Настройка IPv4 ACL

IPv4 ACL предназначен для фильтрации IPv4-пакетов по полям заголовков L3 (IPv4) и L4 (TCP/UDP). Для использования доступны следующие поля:

- IP protocol;
- Source IPv4 address;
- Destination IPv4 address;
- ToS;
- DSCP;
- Source L4 port;
- Destination L4 port.

Шаг 1. Создание списков контроля доступа

Настройка начинается с создания списка доступа с указанием его имени:

```
(als_sw) #configure
(als_sw) (configure) #ip access-list "NameACL"
```

Шаг 2. Добавление правил в список контроля доступа

Для IP-адресов в этих командах используются инвертированные (wildcard) маски.

Создадим разрешающее правило для TCP-трафика с IP-адресами источника 82.52.0.0/16, любыми IP-адресами назначения и L4-портом назначения 22:

```
(als_sw) #configure
(als_sw) (configure) #ip access-list "NameACL"
(als_sw) (configure) (ip-access-list "NameACL") #permit tcp 82.52.0.0 0.0.255.2
55 any eq 22
```

Параметры:

- permit — ключевое слово — означает то, что данное правило разрешающее;
- tcp — IP protocol (возможно задание в числовом виде, например "6" для TCP);
- 82.52.0.0 0.0.255.255 — IP-адрес источника (Source IP) с wildcard-маской;
- any — IP-адрес назначения (Destination IP) — означает разрешение любых Dst IP-адресов;
- eq 22 — L4-порт назначения (Destination L4 port, в данном случае Dst-порт TCP).

L4-порт источника указывается аналогичным образом, но после IP-адреса источника.

Запрещающее правило для IPv4-трафика с любым IP-адресом источника и IP-адресом назначения 82.54.172.5:

```
(als_sw) (configure) (ip-access-list "NameACL") #deny ip any 82.54.172.5 0.0.0.
0
```

Параметры:

- deny — ключевое слово — означает то, что данное правило запрещающее;
- ip — IP protocol, в данном случае под правило попадут любые IP протоколы;
- any — IP-адрес источника (Source IP) — означает разрешение любых Src IP-адресов
- 82.54.172.5 0.0.0.0 — IP-адрес назначения (Destination IP) с wildcard-маской.

Разрешающее правило для IPv4-трафика с IP-адресами источника 82.52.0.0/16 и IP-адресами назначения 82.54.172.0/24:

```
(als_sw) (configure) (ip-access-list "NameACL") #permit ip 82.52.0.0 0.0.255.25
5 82.54.172.0 0.0.0.255
```

Параметры:

- permit — ключевое слово — означает то, что данное правило разрешающее;
- ip — IP protocol, в данном случае под правило попадут любые IP протоколы;
- 82.52.0.0 0.0.255.255 — IP-адрес источника (Source IP) с wildcard-маской;
- 82.54.172.0 0.0.0.255 — IP-адрес назначения (Destination IP) с wildcard-маской;

Шаг 3. Применение списков доступа на интерфейсе (опционально)

Созданные списки контроля доступа применяются на интерфейсе с указанием приоритета:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #ip access-group "NameACL" in 5
```

Параметры:

- "NameACL" — уникальное имя созданного ранее списка доступа;
- 5 — приоритет списка доступа на данном интерфейсе, может иметь значение от 1 до 10000, наименьшее число соответствует наивысшему приоритету (необязательный параметр).

Шаг 4. Применение списков доступа на VLAN (опционально)

Созданные списки контроля доступа применяются на VLAN с указанием приоритета:

```
(als_sw) #configure
(als_sw) (configure) #ip access-group "NameACL" vlan 1 in 5
```

Параметры:

- "NameACL" — уникальное имя созданного ранее списка доступа;
- 1 — VLAN ID, к которому будет применен список доступа;
- 5 — приоритет списка доступа на данном VLAN, может иметь значение от 1 до 10000, наименьшее число соответствует наивысшему приоритету (необязательный параметр).

Просмотр конфигурации

Ниже приведена полная конфигурация для варианта применения описанного списка контроля доступа на интерфейсе:

```
configure
```

```
ip access-list "NameACL"
permit tcp 82.52.0.0 0.0.255.255 any eq 22
deny ip any 82.54.172.5 0.0.0.0
permit ip 82.52.0.0 0.0.255.255 82.54.172.0 0.0.0.255
exit

interface 0/1
ip access-group "NameACL" in 5
exit

exit
```

Итоговая схема прохождения трафика через устройство будет выглядеть следующим образом:

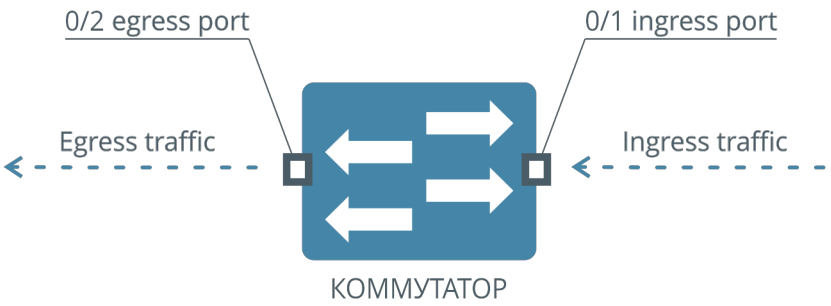

Рисунок 35. Схема прохождения трафика

Source IP	Src L4 port	Destination IP	Dst L4 port	IP protocol	Правило	Прохождение
82.52.0.0/16	-	-	22	TCP (6)	#1	Разрешено
-	-	82.54.172.5	-	-	#2	Запрещено
82.52.0.0/16	-	82.54.172.0/24	-	-	#3	Разрешено

При настройке правил IPv4 ACL есть особенность. К примеру, нужно запретить трафик, описанный определенным правилом (deny), а весь остальной трафик разрешить. Для корректной работы такой настройки нужно после правила deny поместить разрешающее правило для всего остального трафика:

```
configure

ip access-list "NameACL"
deny ip 82.54.172.5 0.0.0.0 any
permit ip any any
exit

interface 0/1
ip access-group "NameACL" in 5
exit

exit
```

В данном примере на интерфейсе 0/1 запрещено прохождение IPv4-трафика с Source IP 82.54.172.5, а весь остальной IPv4-трафик разрешен.

Настройка IPv6 ACL

IPv6 ACL предназначен для фильтрации IPv6-пакетов по полям заголовков L3 (IPv6) и L4 (TCP/UDP). Для использования доступны следующие поля:

- Source IPv6 address;
- Destination IPv6 address;
- IPv6 protocol (NextHeader);
- Source L4 port;
- Destination L4 port;
- Traffic Class;
- DSCP.

Шаг 1. Создание списков контроля доступа

Настройка ACL начинается с создания списка доступа с указанием его имени:

```
(als_sw) (configure) #ipv6 access-list "NameACL"
```

После создания списка доступа необходимо создать правила для этого списка.

Шаг 2. Добавление правил в список контроля доступа

Создадим разрешающее правило для TCP-трафика с IPv6-адресами источника 2001:1:a100:b500::/64, любыми IPv6-адресами назначения и L4-портом назначения 22:

```
(als_sw) #configure
(als_sw) (configure) #ipv6 access-list "NameACL"
(als_sw) (configure) (ipv6-access-list "NameACL") #permit tcp 2001:1:a100:b500:
:/64 any eq 22
```

Параметры:

- permit — ключевое слово — означает то, что данное правило разрешающее;
- tcp — IPv6 Next Header (возможно задание в числовом виде, например "6" для TCP);
- 2001:1:a100:b500::/64 — IPv6-адрес источника (Source IP) с указанием префикса;
- any — IPv6-адрес назначения (Destination IP) — означает разрешение любых Dst IPv6-адресов;
- eq 22 — L4-порт назначения (Destination L4 port, в данном случае Dst-порт TCP).

L4-порт источника указывается аналогичным образом, но после IP-адреса источника.

Запрещающее правило для IPv6-трафика с любым IPv6-адресом источника и IPv6-адресом назначения 2001:175:a500:b100::1/128:

```
(als_sw) (configure) (ipv6-access-list "NameACL") #deny ipv6 any 2001:175:a500:
b100::1/128
```

Параметры:

- deny — ключевое слово — означает то, что данное правило запрещающее;
- ipv6 — IPv6 Next Header, в данном случае под правило попадут любые IP протоколы;
- any — IPv6-адрес источника (Source IP) — означает разрешение любых Src IPv6-адресов;
- 2001:175:a500:b100::1/128 — IPv6-адрес назначения (Destination IP) с указанием префикса.

Разрешающее правило для IPv6-трафика с IPv6-адресами источника 2001:1:a100:b500::/64 и IPv6-адресами назначения 2001:175:a500:b100::/112:

```
(als_sw) (configure) (ipv6-access-list "NameACL") #permit ipv6 2001:1:a100:b500
::/64 2001:175:a500:b100::/112
```

Параметры:

- permit — ключевое слово — означает то, что данное правило разрешающее;
- ipv6 — IPv6 Next Header, в данном случае под правило попадут любые IP протоколы;
- 2001:1:a100:b500::/64 — IPv6-адрес источника (Source IP) с указанием префикса;
- 2001:175:a500:b100::0/112 — IPv6-адрес назначения (Destination IP) с указанием префикса.

Шаг 3. Применение списков контроля доступа на интерфейсе (опционально)

Созданные списки контроля доступа применяются на интерфейсе с указанием приоритета:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #ipv6 traffic-filter "NameACL" in 5
```

Параметры:

- "NameACL" — уникальное имя созданного ранее списка доступа;
- 5 — приоритет списка доступа на данном интерфейсе, может иметь значение от 1 до 10000, наименьшее число соответствует наивысшему приоритету (необязательный параметр).

Шаг 4. Применение списков контроля доступа на VLAN (опционально)

Созданные списки контроля доступа применяются на VLAN с указанием приоритета:

```
(als_sw) #configure
(als_sw) (configure) #ipv6 traffic-filter "NameACL" vlan 1 in 5
```

Параметры:

- "NameACL" — уникальное имя созданного ранее списка доступа;
- 1 — VLAN ID, к которому будет применен список доступа;
- 5 — приоритет списка доступа на данном VLAN, может иметь значение от 1 до 10000, наименьшее число соответствует наивысшему приоритету (необязательный параметр).

Просмотр конфигурации

Ниже приведена полная версия конфигурации для варианта применения описанного списка доступа на интерфейсе:

```
configure

ipv6 access-list "NameACL"
permit tcp 2001:1:a100:b500::/64 any eq 22
deny ipv6 any 2001:175:a500:b100::1/128
permit ipv6 2001:1:a100:b500::/64 2001:175:a500:b100::/112
exit

interface 0/1
ipv6 traffic-filter "NameACL" in 5
exit

exit
```

Итоговая схема прохождения трафика через устройство будет выглядеть следующим образом:

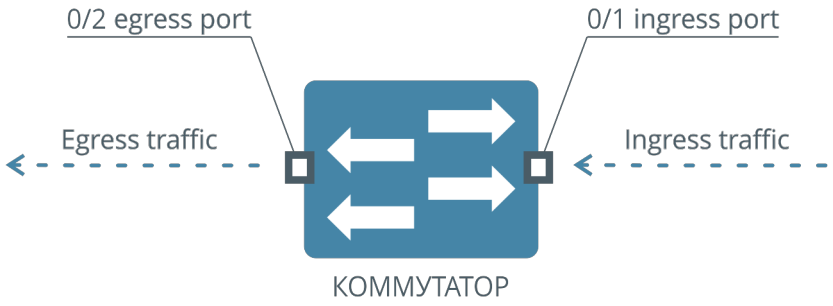

Рисунок 36. Схема прохождения трафика

Source IP	Src L4 port	Destination IP	Dst L4 port	IP protocol	Правило	Прохождение
2001:1:a100:b500::/64	-	-	22	TCP (6)	#1	Разрешено
—	-	2001:175:a500:b100::1	-	-	#2	Запрещено
2001:1:a100:b500::/64	-	2001:175:a500:b100::/112	-	-	#3	Разрешено

При настройке правил IPv6 ACL есть особенность. К примеру, нужно запретить трафик, описанный определенным правилом (deny), а весь остальной трафик разрешить. Для корректной работы такой настройки нужно после правила deny поместить разрешающее правило для всего остального трафика:

```
configure

ipv6 access-list "NameACL"
deny ipv6 2001:175:a500:b100::1/128 any
permit ipv6 any any
exit

interface 0/1
ipv6 traffic-filter "NameACL" in 5
exit

exit
```

В данном примере на интерфейсе 0/1 запрещено прохождение IPv6-трафика с Source IP 2001:175:a500:b100::1, а весь остальной IPv6-трафик разрешен.

Настройка User-defined ACL

User-defined ACL предназначен для фильтрации пакетов по полям, заданным пользователем.

Данный пример демонстрирует блокировку пакетов IGMPv1/v3 Membership Report, разрешая только IGMPv2.

Шаг 1. Создание полей User-defined ACL

Для блокировки IGMP Membership Report требуются следующие поля:

- IP protocol;
- IGMP type;

Создание полей:

```
(als_sw) #configure
(als_sw) (configure) #user-defined access-list template setup
(als_sw) (configure) (user-defined-template) #udf "IP_Protocol" offset l3 9 len
gth 2
(als_sw) (configure) (user-defined-template) #udf "IGMP_Type" offset l4 0 lengt
h 2
(als_sw) (configure) (user-defined-template) #exit
(als_sw) (configure) #exit
```

Описание команд:

- udf "IP_Protocol" offset l3 9 length 2 — IP протокол: смещение от начала L3-заголовка — 9 байт, длина поля — 2 байта;
- udf "IGMP_Type" offset l4 0 length 2 — тип IGMP-пакета: смещение от начала L4-заголовка — 0 байт, длина поля — 2 байта.

Шаг 2. Создание списков контроля доступа

Создадим список доступа:

```
(als_sw) #configure
(als_sw) (configure) #user-defined access-list "BlockIgmpReports"
```

Шаг 3. Добавление правил в список контроля доступа

Для полей User-defined ACL в этих командах используются инвертированные (wildcard) маски.

Заблокируем IGMPv1 Membership Report:

```
(als_sw) (configure) (user-defined-access-list "BlockIgmpReport") #deny udf "IP
_Protocol" 0x0200 0x00ff udf "IGMP_Type" 0x1200 0x00ff
```

Параметры:

- "IP_Protocol" 0x0200 0x00ff — IP Protocol для IGMP имеет значение 0x02 и занимает старший байт поля (используется wildcard-маска);
- "IGMP_Type" 0x1200 0x00ff — IGMP Type для IGMPv1 Membership Report имеет значение 0x12 и занимает старший байт поля (используется wildcard-маска);

Заблокируем IGMPv3 Membership Report:

```
(als_sw) (configure) (user-defined-access-list "BlockIgmpReport") #deny udf "IP
_Protocol" 0x0200 0x00ff udf "IGMP_Type" 0x2200 0x00ff
```

Параметры:

- "IP_Protocol" 0x0200 0x00ff — IP Protocol для IGMP имеет значение 0x02 и занимает старший байт поля (используется wildcard-маска);
- "IGMP_Type" 0x2200 0x00ff — IGMP Type для IGMPv3 Membership Report имеет значение 0x22 и занимает старший байт поля (используется wildcard-маска);

Добавляем разрешающее правило, чтобы разрешить остальные типы трафика:

```
(als_sw) (configure) (user-defined-access-list "BlockIgmpReport") #permit any
```

Шаг 4. Применение списков контроля доступа на интерфейсе

Созданные списки контроля доступа применяются на интерфейсе с указанием приоритета:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #user-defined access-group "BlockIgmpRepor
ts" in 5
```

Параметры:

- "BlockIgmpReports" — уникальное имя созданного ранее списка доступа;
- 5 — приоритет списка доступа на данном интерфейсе, может иметь значение от 1 до 10000, наименьшее число соответствует наивысшему приоритету (необязательный параметр).

Просмотр конфигурации

После выполнения команд из примера получится следующая конфигурация:

```
configure

user-defined access-list template setup
udf "IP_Protocol" offset l3 9 length 2
udf "IGMP_Type" offset l4 0 length 2
exit

user-defined access-list "BlockIgmpReports"
deny udf "IP_Protocol" 0x0200 0x00ff udf "IGMP_Type" 0x1200 0x00ff
deny udf "IP_Protocol" 0x0200 0x00ff udf "IGMP_Type" 0x2200 0x00ff
permit any
exit

interface 0/1
user-defined access-group "BlockIgmpReports" in 5
exit

exit
```

Итоговая схема прохождения трафика через устройство будет выглядеть следующим образом:

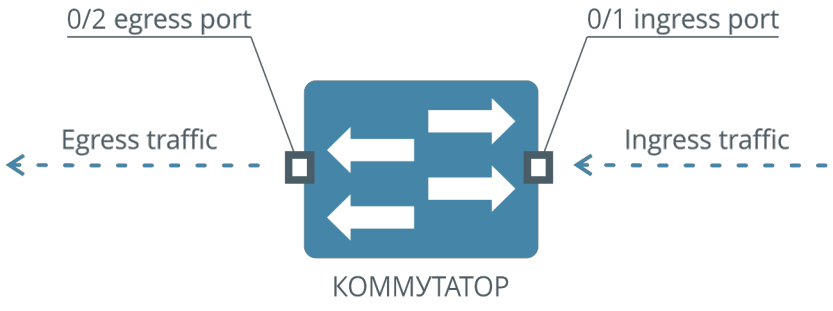

Рисунок 37. Схема прохождения трафика

IP Protocol	IGMP Type	Правило	Прохождение

IGMP (0x02)	IGMPv1 Report (0x12)	#1	Запрещено
IGMP (0x02)	IGMPv3 Report (0x22)	#2	Запрещено
-	-	#3	Разрешено

В результате на входе интерфейса 0/1 будут заблокированы IGMPv1 и IGMPv3 Membership Report, остальные пакеты будут разрешены.

Настройка IPv4 ACL на интерфейсе управления

ACL на интерфейсе управления требуется для того, чтобы ограничить доступ на устройство, при этом сохранив прохождение трафика сквозь устройство.

- Запретим доступ к коммутатору по telnet всем пользователям из подсети 192.168.1.0/24;
- Запретим доступ к коммутатору адресам из подсети 10.11.0.0/24.

Шаг 1. Создание списка контроля доступа

Создание списка доступа для IPv4 ACL на управление не отличается от создания списка доступа для обычного IPv4 ACL:

```
(als_sw) #configure
(als_sw) (configure) #ip access-list "RestrictAccess"
```

Шаг 2. Добавление правил в список контроля доступа

Запретим доступ к коммутатору по telnet всем пользователям из подсети 192.168.1.0/24:

```
(als_sw) (configure) (ip-access-list "RestrictAccess") #deny ip 192.168.1.0 0.0
.0.255 any eq 23
```

Запретим доступ к коммутатору адресам из подсети 10.11.0.0/24:

```
(als_sw) (configure) (ip-access-list "RestrictAccess") #deny ip 10.11.0.0 0.0.2
55.255 any
```

Разрешаем доступ к коммутатору всем адресам, которые не попали под вышеперечисленные правила:

```
(als_sw) (configure) (ip-access-list "RestrictAccess") #permit ip any any
```

Шаг 3. Применение списка контроля доступа на интерфейсе управления

Применяем список контроля доступа на интерфейсе управления. Если необходимо, указываем приоритет правила:

```
(als_sw) #configure
(als_sw) (configure) #ip access-group "RestrictAccess" management in 1
```

Просмотр конфигурации

После выполнения команд из примера получится следующая конфигурация:

```
configure

ip access-list "RestrictAccess"
deny ip 192.168.1.0 0.0.0.255 any eq 23
deny ip 10.11.0.0 0.0.255.255 any
permit ip any any
exit

ip access-group "RestrictAccess" management in 1

exit
```

Важные замечания:

- если в данной конфигурации включить IGMP/DHCP Snooping, то пакеты, предназначенные данным механизмам, с адресов, заблокированных для интерфейса управления, заблокированы не будут;
- IPv4 ACL не блокирует прохождение не-IP-протоколов, в частности, ARP, поэтому устройство сможет отвечать на ARP;
- в IPv4 ACL на интерфейсе управления можно использовать IP-адрес назначения. Эта возможность оставлена для блокировки broadcast-пакетов и создания ACL для конкретного интерфейса управления, если их несколько;

Настройка IPv6 ACL на интерфейсе управления

ACL на интерфейсе управления требуется для того, чтобы ограничить доступ на устройство, при этом сохранив прохождение трафика сквозь устройство.

- Разрешим доступ к коммутатору с Link-local адресов fe80:17ae::/64;
- Разрешим доступ к коммутатору по SSH только из определенной подсети: 2001:17ae::/64.

Шаг 1. Создание списка контроля доступа

Создание списка доступа для IPv6 ACL на управление не отличается от создания списка доступа для обычного IPv6 ACL:

```
(als_sw) #configure
(als_sw) (configure) #ipv6 access-list "RestrictAccess"
```

Шаг 2. Добавление правил в список контроля доступа

Разрешим доступ к коммутатору с Link-local адресов fe80:17ae::/64:

```
(als_sw) (configure) (ipv6-access-list "RestrictAccess") #permit ipv6 fe80:17ae
::/64 any
```

Разрешим доступ к коммутатору по SSH только из определенной подсети: 2001:17ae::/64:

```
(als_sw) (configure) (ipv6-access-list "RestrictAccess") #permit ipv6 2001:17ae
::/64 any eq 22
```

Шаг 3. Применение спика контроля доступа на интерфейсе управления

Применяем список контроля доступа на интерфейсе управления. Если необходимо, указываем приоритет правила:

```
(als_sw) #configure
(als_sw) (configure) #ipv6 traffic-filter "RestrictAccess" management in 1
```

Просмотр конфигурации

После выполнения команд из примера получится следующая конфигурация:

```
configure

ipv6 access-list "RestrictAccess"
permit ipv6 fe80:17ae::/64 any
permit ipv6 2001:17ae::/64 any eq 22
exit

ipv6 traffic-filter "RestrictAccess" management in 1

exit
```

Важные замечания:

- если в данной конфигурации включить LDRA/MLD/IPv6 Snooping, то пакеты, предназначенные данным механизмам, с адресов, заблокированных для интерфейса управления, заблокированы не будут;
- IPv6 ACL не блокирует прохождение ICMPv6-сообщений Neighbor Solicitation, Neighbor Advertisement, Router Solicitation, Router Advertisement, так как эти сообщения необходимы для нормального функционирования IPv6-сети;

14.3. Типовые вопросы и ошибки

ACL — мощный механизм управления трафиком, поэтому при его использовании следует принимать меры предосторожности. Неверное конфигурирование списков доступа может привести к блокировке полезного трафика, в том числе к потере удаленного управления устройством.

Список того, на что стоит обращать внимание при конфигурировании ACL:

- весь трафик, не подпадающий ни под одно из правил ACL, примененных для входящего трафика, будет **блокирован**;
- конфигурирование ACL на управляющем VLAN или интерфейсе, ведущем к вышестоящему оборудованию, может привести к потере управления коммутатором. При конфигурировании списков доступа необходимо уделить особое внимание настройкам удаленного управления коммутатором;
- списки доступа, примененные на интерфейсе, имеют приоритет над списками, примененными на VLAN. Трафик, заблокированный правилами на интерфейсе, не может быть разрешен правилами, примененными на VLAN.

ГЛАВА 15. MULTICAST

15.1. Введение в Multicast

В сетях Ethernet существует три способа адресации для передачи пакетов:

- unicast — одноцелевая передача пакетов, подразумевает передачу пакета единственному адресату;
- broadcast — широковещательная передача пакетов, подразумевает доставку пакета всем адресатам, находящимся в одном L2-домене;
- multicast — групповая передача, форма широковещания, при которой адресом назначения сетевого пакета является multicast группа.

Multicast-адресация используется в том случае, если нужно передавать одинаковую информацию многим адресатам. Если для этой цели использовать unicast-адресацию, то придется передавать копию информации для каждого получателя. Сеть будет перегружена пакетами с одинаковым содержанием, но с разными адресами назначения.

В дальнейшем IPv4 multicast-адрес назначения будет называться multicast-группой. Multicast-группа — multicast IPv4 адрес, используемый в качестве адреса назначения в многоадресной рассылке.

Протокол IGMP

Multicast router — источник или ретранслятор многоадресных пакетов, данный узел должен иметь возможность независимо останавливать и возобновлять вещание любой multicast-группы.

Протокол IGMP разработан для управления подпиской на multicast-группы. В протоколе IGMP клиент с сервером передают информацию друг другу с помощью сообщений. Можно выделить следующие типы сообщений:

- IGMP Membership Report — сообщение клиента серверу, клиент желает добавить multicast-группу к списку получаемых;
- IGMP Leave — сообщение клиента серверу, клиент желает убрать multicast-группу из списка получаемых;
- IGMP Membership Query — сообщение сервера клиенту, в котором сервер просит обновить подписку на получаемую клиентом группу (Speicific Query) или на все группы (General Query). Если подписка не будет обновлена в течении определенного времени, сервер перестанет вещать группу (или группы, в случае General Query) в данный сегмент сети.

Всего протокол IGMP имеет 3 версии. Основные типы сообщений для каждой из версий приведены в таблице:

Типы пакетов	Версия
IGMP Membership Report v1	IGMPv1
IGMP Membership Report v2	IGMPv2
IGMP Membership Report v3	IGMPv3
IGMP Membership Query	IGMPv1/v2/v3
IGMP Leave	IGMPv2

Сообщения "IGMP Membership Report" всех версий также иногда называют "IGMP Join".

IGMP Snooping на L2-коммутаторах доступа

IGMP Snooping — процесс отслеживания IGMP сообщений. Основная задача IGMP Snooping — предотвратить отсылку multicast-групп клиентам, которые явно не запросили данные группы.

Рассмотрим схему:

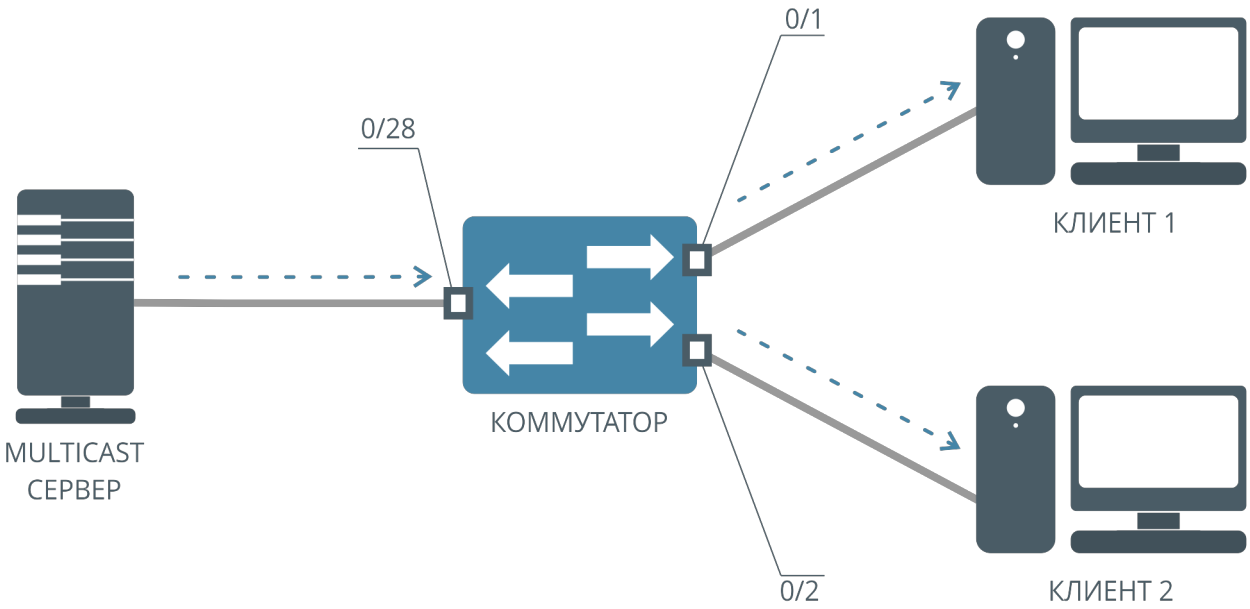

Рисунок 38. Схема передачи Multicast-трафика

Рассмотрим ситуацию, когда на коммутаторе отсутствует или отключен IGMP Snooping. Все группы multicast-трафика будут передаваться с uplink-интерфейса всем клиентам, вне зависимости от того, требуется клиентам этот трафик или нет. В итоге входящий клиентский канал окажется забит ненужным трафиком. Выходом из такой ситуации является включение IGMP Snooping на коммутаторе. В этом случае клиентам будут передаваться только те multicast-группы, которые явно запросил клиент.

Технически подписка на определенные группы на коммутаторах происходит с помощью таблицы подписок. Это специальная аппаратная таблица, в которой хранится адрес multicast-группы, а также интерфейсы, на которые следует передавать эту группу. Данная таблица формируется с помощью IGMP Snooping.

15.2. Настройка Multicast на коммутаторах АЛСиТЕК

Общие принципы конфигурирования

Служба IGMP Snooping

Служба IGMP Snooping настраивается в несколько шагов. Есть возможность настройки IGMP Snooping на интерфейсах и на VLAN.

Если настроить IGMP Snooping на интерфейсе, то будут обрабатываться все IGMP сообщения, приходящие на данный интерфейс во всех VLAN. Если настроить IGMP Snooping на VLAN, то будут обрабатываться все IGMP сообщения в данном VLAN.

При настройках на интерфейсе каждому интерфейсу коммутатора отводится одна из ролей:

- интерфейс не участвует в обработке IGMP-сообщений — IGMP-сообщения, которые приходят на данный интерфейс, не будут обрабатываться, но могут передаваться на другие интерфейсы в соответствии с таблицей подписок;
- интерфейс является клиентским — за данным интерфейсом располагается потребитель multicast-трафика. Сообщения "IGMP Membership Report", "IGMP Leave" будут передаваться с данного интерфейса на серверные интерфейсы, при этом будет соответствующим образом изменяться таблица подписок. Если на данный интерфейс придет "IGMP Query" пакет, то пакет будет отброшен, так как роль интерфейса не подразумевает вещание multicast-трафика;
- интерфейс является серверным — за данным интерфейсом располагается источник multicast-трафика. Сообщения "IGMP Query", приходящие на данный интерфейс, будут передаваться на клиентские интерфейсы. Сообщения "IGMP Membership Report" и "IGMP Leave", приходящие на данный интерфейс, будут отбрасываться;
- интерфейс является смешанным — комбинация из клиентского и серверного интерфейсов, все IGMP-сообщения будут обработаны.

При настройке на VLAN выбирается один или несколько VLAN, в которых будет происходить обработка сообщений IGMP. Каждый интерфейс, на котором будет разрешена обработка выбранных VLAN, автоматически станет клиентским. Затем настраивается один или несколько серверных интерфейсов для каждого из VLAN. После настройки IGMP Snooping начнет обрабатывать IGMP-сообщения и строить на их основе таблицу подписок.

Настройка передачи multicast-пакетов

Чтобы включить передачу multicast-пакетов только определенным клиентам, не достаточно только включить IGMP Snooping. IGMP Snooping формирует таблицу подписок. Передача multicast-пакетов может осуществляться как по записям в таблице, так и игнорируя записи в таблице подписок. За данное поведение отвечает режим передачи multicast во VLAN (Multicast VLAN Forwarding Mode) или сокращенно "mcast_vfm".

В коммутаторах АЛСиТЕК есть два режима передачи multicast во VLAN:

- передавать multicast-пакеты аналогично broadcast-пакетам, записи таблицы подписок игнорировать (команда: "mcast_vfm <VLAN> forward_all");
- передавать multicast-пакеты по таблице подписок. Если записи в таблице нет, то не передавать пакеты (команда: "mcast_vfm <VLAN> forward_registered").

По умолчанию для всех VLAN используется первая настройка, все multicast-пакеты передаются вне зависимости от таблицы подписок. Такой режим обычно применяется в случаях, когда отключен IGMP Snooping или же клиенты не могут послать IGMP-сообщения, а обеспечить доставку multicast-групп до клиентов требуется.

Второй режим, когда пакеты передаются в соответствии с таблицей подписок, обычно применяется совместно с IGMP Snooping, когда нужно обеспечить передачу только запрошенных клиентами групп.

Настройка IGMP Snooping на интерфейсах

Настройку IGMP Snooping будем производить на примере подключения клиента 1 и клиента 2 к multicast-серверу. Multicast-сервер вещает некоторые группы во VLAN 10 и 20 и подключен к коммутатору через интерфейс 0/28. Клиент 1 и клиент 2 получают multicast-трафик в 10 и 20 VLAN и подключены к интерфейсам коммутатора 0/1 и 0/2 соответственно.

Типовая схема подключения приведена ниже:

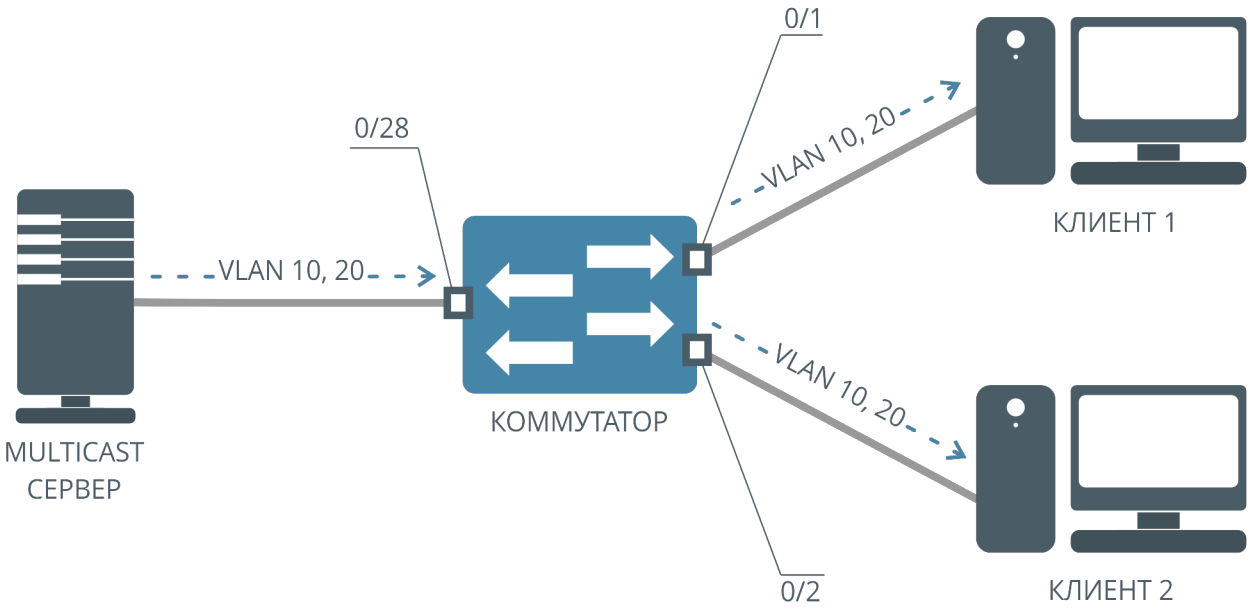

Рисунок 39. Схема подключения с двумя клиентами

Шаг 1. Настройка VLAN на устройстве

Для корректной работы IGMP Snooping на коммутаторе АЛСиТЕК согласно приведенной схеме требуется произвести следующие предварительные настройки.

Создаем VLAN 10 и 20, в которых будет осуществляться вещание:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 10,20
(als_sw) (Vlan) #exit
```

Настраиваем VLAN 10 и 20 на клиентских интерфейсах:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1,0/2
(als_sw) (configure) (interface 0/1-0/2) #vlan participation include 10,20
(als_sw) (configure) (interface 0/1-0/2) #vlan tagging 10,20
(als_sw) (configure) (interface 0/1-0/2) #exit
(als_sw) (configure) #exit
```

Настраиваем VLAN 10 и 20 на серверном интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #vlan participation include 10,20
(als_sw) (configure) (interface 0/28) #vlan tagging 10,20
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
```

Шаг 2. Включение службы IGMP Snooping на устройстве

Для глобального включения службы выполняем команду:

```
(als_sw) #configure
(als_sw) (configure) #set igmp
(als_sw) (configure) #exit
```

Шаг 3. Назначение клиентских интерфейсов IGMP Snooping

Для указания клиентских интерфейсов выполняем команду:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1,0/2
(als_sw) (configure) (interface 0/1-0/2) #set igmp
(als_sw) (configure) (interface 0/1-0/2) #exit
(als_sw) (configure) #exit
```

После данного шага интерфейсы 0/1 и 0/2 станут клиентскими интерфейсами, то есть сообщения "IGMP Membership Report" и "IGMP Leave" будут изменять таблицу подписок коммутатора. Однако сообщения "IGMP Membership Report" и "IGMP Leave" не будут передаваться на другие интерфейсы, так как в текущей конфигурации нет серверных интерфейсов.

Шаг 4. Назначение серверных интерфейсов IGMP Snooping

Для указания серверного интерфейса выполняем команду:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #set igmp mrouter interface
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
```

После данного шага сообщения "IGMP Membership Report" и "IGMP Leave", приходящие на 0/1 и 0/2 интерфейсы, будут не только обрабатываться, но и передаваться на 0/28 интерфейс. Сообщения "IGMP Query", приходящие на 0/28 интерфейс, будут передаваться на интерфейсы 0/1 и 0/2.

Шаг 5. Настройка передачи multicast-пакетов во VLAN

Настраиваем передачу multicast-пакетов в 10 и 20 VLAN таким образом, чтобы при передаче учитывались записи в таблице подписок коммутатора:

```
(als_sw) #configure
(als_sw) (configure) #mcast_vfm 10 forward_registered
(als_sw) (configure) #mcast_vfm 20 forward_registered
(als_sw) (configure) #exit
```

Просмотр клиентов

Просмотреть текущие подписки IGMP Snooping можно командой:

```
(als_sw) #show igmpsnooping groups all

  VlanId  Multicast Group  Version  Iface   Uptime         GMI (sec)  QRI (sec)

  ------  ---------------  -------  ------  -------------  ---------  ---------

  10      224.0.21.11      IGMPv2   0/1     0d, 00:00:48         212
  20      224.1.21.87      IGMPv2   0/1     0d, 00:00:48         212
  20      224.1.21.121     IGMPv2   0/2     0d, 00:00:48         212
```

Команда выводит следующую информацию:

- VlanId — номер VLAN, в котором была осуществлена подписка;
- Multicast Group — IPv4-адрес multicast группы;
- Version — версия протокола IGMP;
- Iface — интерфейс, за которым расположен клиент, запросивший данную группу;
- Uptime — время длительности подписки;
- GMI (sec) — значение таймера "group membership interval";
- QRI (sec) — значение таймера "query response interval".

Настройка IGMP Snooping на VLAN

Настройку IGMP Snooping на VLAN будем производить на примере подключения клиента 1 и клиента 2 к multicast-серверу. Multicast-сервер вещает некоторые группы в VLAN 10 и 20 и подключен к коммутатору через интерфейс 0/28. Клиент 1 и клиент 2 получают multicast-трафик во 10 и 20 VLAN и подключены к интерфейсам коммутатора 0/1 и 0/2 соответственно.

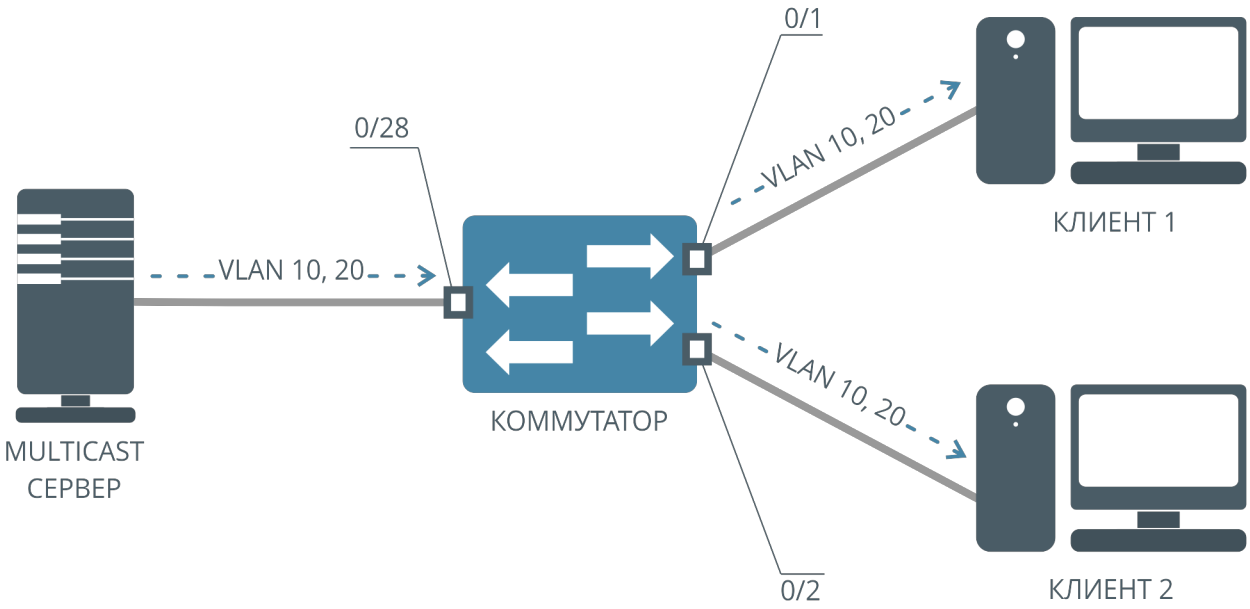

Рисунок 40. Схема подключения с двумя клиентами

Шаг 1. Настройка VLAN на устройстве

Смотрите пункт "Настройка IGMP Snooping на интерфейсах", шаг 1.

Шаг 2. Включение службы IGMP Snooping на устройстве

Смотрите пункт "Настройка IGMP Snooping на интерфейсах", шаг 2.

Шаг 3. Назначение IGMP Snooping на VLAN

Для включения службы на определенных VLAN выполняем команду:

```
(als_sw) #vlan database
(als_sw) (Vlan) #set igmp 10,20
(als_sw) (Vlan) #exit
```

После данного шага во VLAN 10 и 20 начнут обрабатываться сообщения "IGMP Membership Report" и "IGMP Leave". Соответственно таблица подписок будет заполняться только для 10 и 20 VLAN. IGMP-сообщения в других VLAN будут передаваться согласно режиму передачи multicast-трафика во VLAN и не будут обработаны службой IGMP Snooping.

Шаг 4. Назначение серверных интерфейсов IGMP Snooping

Для указания серверного интерфейса в 10 и 20 VLAN выполняем команду:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #set igmp mrouter 10,20
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
```

После данного шага во VLAN 10 и 20 сообщения "IGMP Membership Report" и "IGMP Leave" с клиентских интерфейсов будут передаваться на серверные. А сообщения "IGMP Query" в VLAN 10 и 20 с серверных интерфейсов будут передаваться на клиентские. IGMP-сообщения в других VLAN будут обрабатываться согласно режиму передачи multicast-трафика во VLAN и не будут обработаны службой IGMP Snooping.

Шаг 5. Настройка передачи multicast-пакетов во VLAN

Настраиваем продвижение multicast-пакетов в 10 и 20 VLAN таким образом, чтобы при передаче учитывались записи в таблице подписок:

```
(als_sw) #configure
(als_sw) (configure) #mcast_vfm 10 forward_registered
(als_sw) (configure) #mcast_vfm 20 forward_registered
(als_sw) (configure) #exit
```

Настройка MVR

IGMP Snooping позволяет отправлять управляющие IGMP-сообщения в определенном VLAN. Такая настройка может понадобиться в случае, если у каждого из клиентов, подключенных к коммутатору, свой собственный VLAN, а услуги IPTV должны оказываться всем клиентам.

Для реализации используется технология MVR (англ. Multicast VLAN Replication). Данная технология позволяет объединить запросы клиентов из разных VLAN в один, и передавать запросы клиентов на сервер в MVR VLAN.

Настройку IGMP MVR будем производить на примере подключения клиентов 1 и 2 к multicast-серверу. Multicast-сервер вещает multicast-группы в VLAN 10 и подключен к коммутатору через интерфейс 0/28. Клиенты получают multicast-трафик нетегированным и подключены к интерфейсам коммутатора 0/1 и 0/2 соответственно. Услуги Internet клиенты получают каждый в своем VLAN, 1208 и 1209 соответственно.

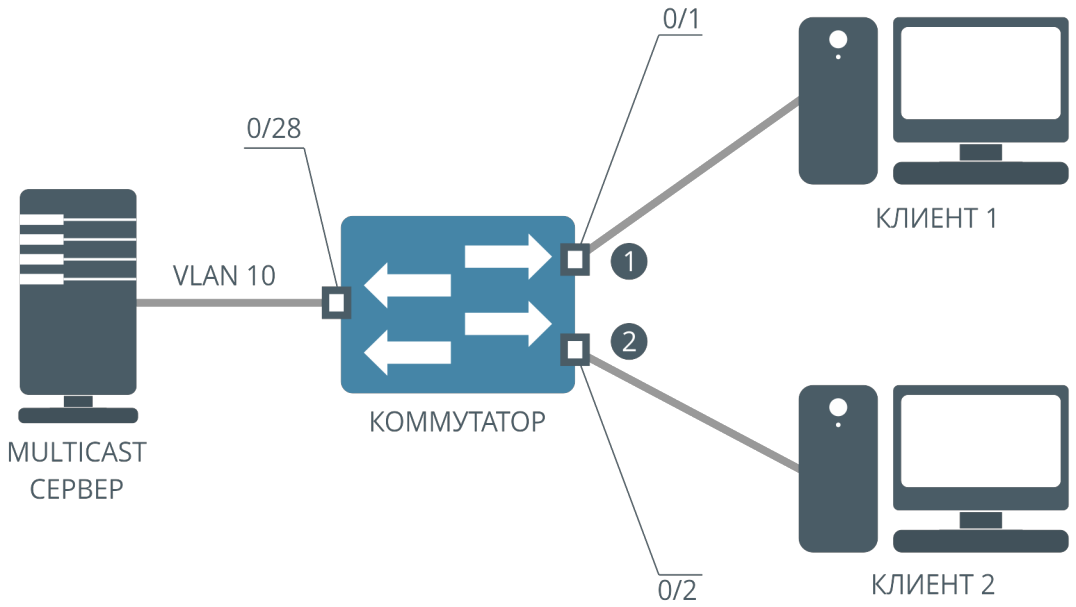

Рисунок 41. Схема работы MVR

1. Интерфейс первого клиента, на входе всему трафику назначается VLAN 1208.
2. Интерфейс второго клиента, на входе всему трафику назначается VLAN 1209.

Шаг 1. Настройка VLAN на устройстве

Для корректной работы IGMP Snooping на коммутаторе АЛСиТЕК согласно приведенной схеме требуется произвести следующие предварительные настройки.

Создаем VLAN 10, 1208 и 1209:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 10,1208,1209
(als_sw) (Vlan) #exit
```

Настраиваем VLAN на клиентских интерфейсах:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan participation include 10,1208
(als_sw) (configure) (interface 0/1) #vlan pvid 1208
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
(als_sw) (configure) #interface 0/2
(als_sw) (configure) (interface 0/2) #vlan participation include 10,1209
(als_sw) (configure) (interface 0/2) #vlan pvid 1209
(als_sw) (configure) (interface 0/2) #exit
(als_sw) (configure) #exit
```

Настраиваем VLAN 10 и клиентские VLAN 1208, 1209 на серверном интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #vlan participation include 10,1208,1209
(als_sw) (configure) (interface 0/28) #vlan tagging 10,1208,1209
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
```

Шаг 2. Включение службы IGMP Snooping на устройстве

Работа MVR возможна только при включенном IGMP Snooping. Выполним следующие команды:

```
(als_sw) #configure
(als_sw) (configure) #set igmp
(als_sw) (configure) #interface 0/1-0/2
(als_sw) (configure) (interface 0/1-0/2) #set igmp
(als_sw) (configure) (interface 0/1-0/2) #exit
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #set igmp mrouter interface
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
```

Шаг 3. Включение механизма IGMP MVR на устройстве

Для включения механизма IGMP MVR выполняем команды:

```
(als_sw) #configure
(als_sw) (configure) #set igmp mvr
(als_sw) (configure) #set igmp mvr 10
(als_sw) (configure) #exit
```

После данного шага включается механизм IGMP MVR, для MVR выбирается VLAN 10. Клиенты посылают сообщения "IGMP Membership Report" и "IGMP Leave" без тега. На клиентском интерфейсе сообщения получают 1208 или 1209 VLAN. Механизм IGMP MVR поменяет тег клиента на 10 MVR VLAN. После этого сообщения будут переданы на серверный интерфейс и отправятся на multicast-сервер с VLAN 10. Подписка будет сохранена во VLAN 10.

Шаг 4. Установка значения 802.1p для IGMP MVR трафика (опционально)

По умолчанию MVR заменяет тег клиента настроенным значением тега, и значение 802.1p берет из тега клиента. Однако это поведение можно изменить, установив значение метки с помощью команды:

```
(als_sw) #configure
(als_sw) (configure) #set igmp mvr cos 5
(als_sw) (configure) #exit
```

При данной настройке 802.1p метка в пакетах, обработанных MVR, всегда будет равна 5.

Просмотр клиентов

Контролировать текущие подписки IGMP Snooping можно с помощью команды "show igmpsnooping groups all". Обратите внимание, что в поле "VlanId" отображается MVR VLAN, а не клиентский VLAN.

Вид таблицы может быть следующим:

```
(als_sw) #show igmpsnooping groups all

  VlanId  Multicast Group  Version  Iface   Uptime         GMI (sec)  QRI (sec)

  ------  ---------------  -------  ------  -------------  ---------  ---------

  10      224.0.21.11      IGMPv2   0/1     0d, 00:00:42         210
  10      224.1.21.87      IGMPv2   0/1     0d, 00:00:48         209
  10      224.1.21.121     IGMPv2   0/2     0d, 00:00:31         212
```

Настройка Selective MVR

В предыдущем пункте был рассмотрен механизм MVR. Selective MVR может понадобиться, если клиент должен получать Multicast от двух разных источников в разных VLAN. При этом Multicast IP-адреса разных источников не должны совпадать в разных VLAN.

Для реализации используется технология Selective MVR (англ. Selective Multicast VLAN Replication).

Настройку IGMP Selective MVR будем производить на примере подключения клиентов 1 и 2 к Multicast-серверу 1 и 2. Multicast-сервер 1 вещает Multicast-группы 224.1.0.0/24 в VLAN 10 и подключен к коммутатору через интерфейс 0/27. Multicast-сервер 2 вещает Multicast-группы 224.2.1.1-224.2.1.10 в VLAN 20 и подключен к коммутатору через интерфейс 0/28. Клиенты получают Multicast-трафик нетегированным и подключены к интерфейсам коммутатора 0/1 и 0/2 соответственно. Услуги Internet клиенты получают каждый в своем VLAN (1208 и 1209 соответственно).

Шаг 1. Настройка VLAN на устройстве

Для корректной работы IGMP Snooping на коммутаторе АЛСиТЕК требуется произвести следующие предварительные настройки.

Создаем VLAN 10, 20, 1208 и 1209:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 10,20,1208,1209
(als_sw) (Vlan) #exit
```

Настраиваем VLAN на клиентских интерфейсах:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan participation include 10,20,1208
(als_sw) (configure) (interface 0/1) #vlan pvid 1208
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
(als_sw) (configure) #interface 0/2
(als_sw) (configure) (interface 0/2) #vlan participation include 10,20,1209
(als_sw) (configure) (interface 0/2) #vlan pvid 1209
(als_sw) (configure) (interface 0/2) #exit
(als_sw) (configure) #exit
```

Настраиваем VLAN 10 и клиентские VLAN 1208, 1209 на серверном интерфейсе 0/27:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #vlan participation include 10,1208,1209
(als_sw) (configure) (interface 0/28) #vlan tagging 10,1208,1209
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
```

Настраиваем VLAN 20 и клиентские VLAN 1208, 1209 на серверном интерфейсе 0/28:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #vlan participation include 20,1208,1209
(als_sw) (configure) (interface 0/28) #vlan tagging 20,1208,1209
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
```

Шаг 2. Включение службы IGMP Snooping на устройстве

Работа MVR возможна только при включенном IGMP Snooping. Выполним следующие команды:

```
(als_sw) #configure
(als_sw) (configure) #set igmp
(als_sw) (configure) #interface 0/1-0/2
(als_sw) (configure) (interface 0/1-0/2) #set igmp
(als_sw) (configure) (interface 0/1-0/2) #exit
(als_sw) (configure) #interface 0/27
(als_sw) (configure) (interface 0/27) #set igmp mrouter interface
(als_sw) (configure) (interface 0/27) #exit
(als_sw) (configure) #exit
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #set igmp mrouter interface
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
```

Шаг 3. Распределение Multicast IP-адресов между двумя Multicast-серверами

По условиям задачи Multicast-сервер 1 должен вещать в группах 224.1.0.0/24. Создадим профиль Selective MVR "IPTV_SERVER_1":

```
(als_sw) #configure
(als_sw) (configure) #set igmp mvr selective "IPTV_SERVER_1"
(als_sw) (configure) (selective mvr: "IPTV_SERVER_1") #group 224.1.0.0 mask 255
.255.0.0
(als_sw) (configure) (selective mvr: "IPTV_SERVER_1") #exit
```

Укажем VLAN 10 для Multicast-сервера 1:

```
(als_sw) #configure
(als_sw) (configure) #
(als_sw) (configure) #set igmp mvr selective "IPTV_SERVER_1" vlan 10
(als_sw) (configure) #exit
```

Multicast-сервер 2 должен вещать в группах 224.2.1.1-224.2.1.10. Создадим профиль Selective MVR "IPTV_SERVER_2":

```
(als_sw) #configure
(als_sw) (configure) #set igmp mvr selective "IPV_SERVER_2"
(als_sw) (configure) (selective mvr: "IPTV_SERVER_2") #group 224.2.1.1 to 224.2
.1.10
(als_sw) (configure) (selective mvr: "IPTV_SERVER_2") #exit
(als_sw) (configure) #exit
```

Укажем VLAN 20 для Multicast-сервера 2:

```
(als_sw) #configure
(als_sw) (configure) #
(als_sw) (configure) #set igmp mvr selective "IPTV_SERVER_2" vlan 20
(als_sw) (configure) #exit
```

Шаг 4. Включение механизма IGMP Selective MVR на устройстве

Для включения механизма IGMP Selective MVR выполняем команды:

```
(als_sw) #configure
(als_sw) (configure) #set igmp mvr
(als_sw) (configure) #set igmp mvr 1
(als_sw) (configure) #exit
```

После данного шага включается механизм IGMP MVR, для IGMP-пакетов VLAN выбирается в соответствии с профилями "IPTV_SERVER_1" и "IPTV_SERVER_2". Если Multicast-группа не попала ни в одно из этих правил, то срабатывает обычный механизм MVR.
Клиенты посылают сообщения "IGMP Membership Report" и "IGMP Leave" без тега. На клиентском интерфейсе сообщения получают 1208 или 1209 VLAN. Механизм IGMP MVR поменяет тег клиента на Selective MVR VLAN (в зависимости от группы). После этого сообщения будут переданы на серверный интерфейс.

Просмотр клиентов

Контролировать текущие подписки IGMP Snooping можно с помощью команды "show igmpsnooping groups all". Обратите внимание, что в поле "VlanId" отображается MVR VLAN, а не клиентский VLAN.

Вид таблицы может быть следующим:

```
(als_sw) #show igmpsnooping groups all

  VlanId  Multicast Group  Version  Iface   Uptime         GMI (sec)  QRI (sec)

  ------  ---------------  -------  ------  -------------  ---------  ---------

  1       224.3.1.1        IGMPv2   0/2     0d, 00:00:31         212
  10      224.1.21.11      IGMPv2   0/1     0d, 00:00:42         210
  20      224.2.1.3        IGMPv2   0/1     0d, 00:00:48         206
  20      224.2.1.4        IGMPv2   0/2     0d, 00:00:48         209
```

Группа 224.3.1.1 не попала в правила Selective MVR и получила VLAN 1.
Группа 224.1.21.11 попала в правила Selective MVR "IPTV_SERVER_1" (диапазон 224.1.0.0/24) и получила VLAN 10.
Группа 224.2.1.3 попала в правила Selective MVR "IPTV_SERVER_2" (диапазон 224.2.1.1-224.2.1.10) и получила VLAN 20.
Группа 224.2.1.4 попала в правила Selective MVR "IPTV_SERVER_2" (диапазон 224.2.1.1-224.2.1.10) и получила VLAN 20.

Настройка динамического назначения mrouter-интерфейсов

Настройка динамического назначения mrouter-интерфейсов производится в тех случаях, когда нельзя заранее определить, какой именно интерфейс должен играть роль серверного. Рассмотрим данную настройку на примере подключения клиента 1 к multicast-серверу через два коммутатора. Multicast-сервер подключен к коммутатору 2 через интерфейс 0/28. Коммутатор 2 соединен с коммутатором 1 через интерфейсы 0/26 и 0/27. Клиент 1 подключен к коммутатору 1 через интерфейс 0/1.

В данной схеме коммутаторы 1 и 2 соединены двумя линками. Такое соединение является избыточным для сети Ethernet, обычно в таких ситуациях прохождение трафика через один из интерфейсов будет заблокировано с помощью STP. Однако заранее не известно, какой из интерфейсов будет заблокирован. В таких случаях можно применить механизм IGMP dynamic mrouter.

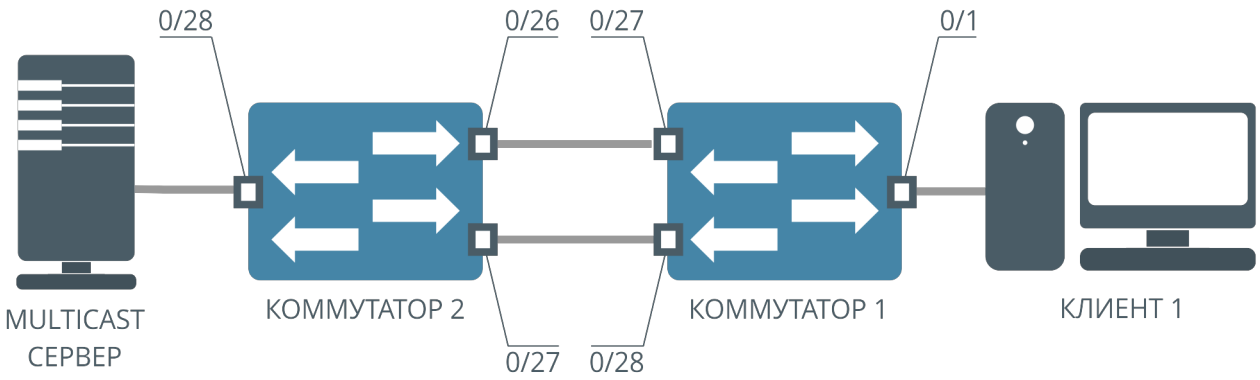

Рисунок 42. Схема работы IGMP dynamic mrouter

Далее будем рассматривать только конфигурацию коммутатора 1.

Шаг 1. Включение службы IGMP Snooping на устройстве

Для глобального включения службы выполняем команду:

```
(als_sw) #configure
(als_sw) (configure) #set igmp
(als_sw) (configure) #exit
```

Шаг 2. Назначение клиентских интерфейсов IGMP Snooping

Помечаем интерфейсы 0/1, 0/27, 0/28 как клиентские:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1,0/27,0/28
(als_sw) (configure) (interface 0/1,0/27-0/28) #set igmp
(als_sw) (configure) (interface 0/1,0/27-0/28) #exit
(als_sw) (configure) #exit
```

Шаг 3. Включение механизма IGMP dynamic mrouter на устройстве

Для включения механизма IGMP dynamic mrouter выполняем команду:

```
(als_sw) #configure
(als_sw) (configure) #set igmp mrouter dynamic
(als_sw) (configure) #exit
```

После данного шага включается механизм IGMP dynamic mrouter. Если на интерфейс, участвующий в IGMP Snooping ("set igmp" на интерфейсе или "set igmp <vlan>"), приходит сообщение "IGMP Query" от multicast-сервера, то интерфейс переходит в серверный режим.

Настройка фильтрации IGMP-групп

Механизм фильтрации multicast-групп позволяет ограничивать доступ к тем или иным multicast-группам. Фильтрация происходит с помощью профилей. Профиль может состоять из нескольких правил, которыми задаются multicast-группы. Есть возможность задать группы по маске или в виде диапазона. Профили могут быть двух типов:

- permit — разрешающий профиль, указанные в профиле multicast-группы будут разрешены, все остальные группы будут запрещены;
- deny — запрещающий профиль, указанные в профиле multicast-группы будут запрещены, все остальные группы будут разрешены.

На одном интерфейсе можно применять профили только одного типа. Профили влияют на возможность подписки клиентов на определенные группы и работают в момент подписки.

Схема для примера изображена ниже:

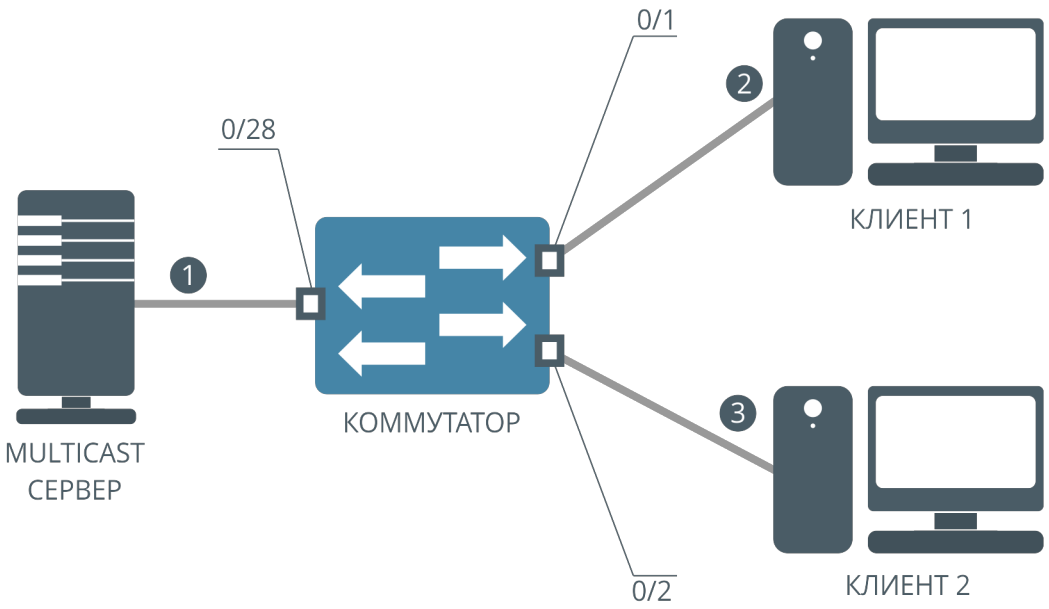

Рисунок 43. Схема работы профилей IGMP

Шаг 1. Включение службы IGMP Snooping на устройстве

Работа фильтрации IGMP-групп возможна только при включенном IGMP Snooping. Настроим 0/1 и 0/2 интерфейсы как клиентские, 0/28 интерфейс как серверный:

```
(als_sw) #configure
(als_sw) (configure) #set igmp
(als_sw) (configure) #interface 0/1-0/2
(als_sw) (configure) (interface 0/1-0/2) #set igmp
(als_sw) (configure) (interface 0/1-0/2) #exit
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #set igmp mrouter interface
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
```

Шаг 2. Создание профилей

Создаем разрешающий профиль "Client1", добавляем в профиль разрешенные клиенту 1 группы:

```
(als_sw) #configure
(als_sw) (configure) #set igmp profile "Client1" permit
(als_sw) (configure) (permit profile: "Client1") #group 224.1.1.1 to 224.1.1.4
(als_sw) (configure) (permit profile: "Client1") #group 224.1.20.0 mask 255.255
.255.0
(als_sw) (configure) (permit profile: "Client1") #exit
(als_sw) (configure) #exit
```

Создаем запрещающий профиль "Client2", добавляем в профиль запрещенные для клиента 2 группы:

```
(als_sw) #configure
(als_sw) (configure) #set igmp profile "Client2" deny
(als_sw) (configure) (deny profile: "Client2") #group 224.1.1.5 to 224.1.1.9
(als_sw) (configure) (deny profile: "Client2") #group 224.1.30.1 mask 255.255.2
55.255
(als_sw) (configure) (deny profile: "Client2") #exit
(als_sw) (configure) #exit
```

Шаг 3. Применение профиля на клиентском интерфейсе

Применяем разрешающий профиль для клиента 1:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #set igmp profile "Client1"
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
```

После данного шага на интерфейсе 0/1 будут разрешены только группы в диапазоне 224.1.1.1 — 224.1.1.4, 224.1.20.0 — 224.1.20.255. Остальные группы будут блокироваться.

Применяем запрещающий профиль для клиента 2:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/2
(als_sw) (configure) (interface 0/2) #set igmp profile "Client2"
(als_sw) (configure) (interface 0/2) #exit
(als_sw) (configure) #exit
```

После данного шага на интерфейсе 0/2 будут разрешен все группы, кроме групп в диапазоне 224.1.1.5 — 224.1.1.9, 224.1.30.1.

Ограничение количества multicast групп на клиентском интерфейсе

Данная настройка необходима в тех случаях, когда необходимо ограничить максимальное количество multicast групп, трафик с которых клиент может получать одновременно.

На схеме показан случай, когда клиент получает две группы: 224.1.1.1 и 224.1.1.2, затем запрашивает третью группу 224.1.1.3. Запрос на подключение к группе 224.1.1.3 будет блокироваться до тех пор, пока есть две любые активные подписки на группы.

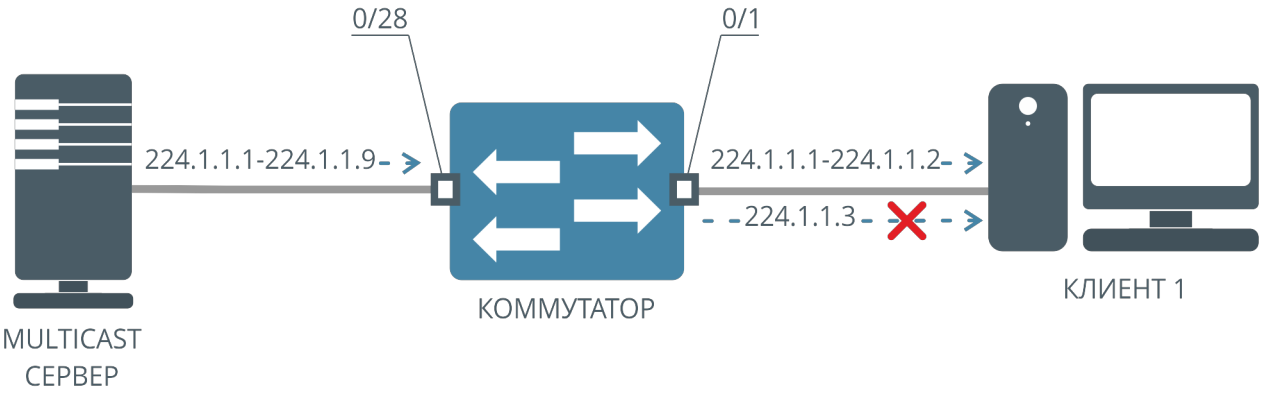

Рисунок 44. Схема работы ограничения количества multicast-групп на клиентском интерфейсе

Шаг 1. Базовая настройка службы IGMP Snooping на устройстве

Работа ограничения количества multicast-групп на интерфейсе возможна только при включенном IGMP Snooping. Это можно сделать с помощью команд:

```
(als_sw) #configure
(als_sw) (configure) #set igmp
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #set igmp
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #set igmp mrouter interface
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
```

Шаг 2. Включение ограничения количества multicast-групп на клиентском интерфейсе

Для задания ограничения количества multicast-групп на клиентском интерфейсе выполняем команду:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #set igmp mcastgrouplimit 2
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
```

После данного шага клиентский интерфейс сможет подписаться только на две группы одновременно.

Настройка статических групп

Настройка статических групп применяется в случае, когда клиент не может запросить получение multicast группы с помощью протокола IGMP. В этом случае на клиентский интерфейс статически прописывается определенный диапазон multicast групп. Клиент будет получать все прописанные multicast-группы. Если на интерфейсе вместе со статическими группами включен IGMP Snooping, то IGMP запросы на подключение к статическим группам будут блокироваться.

Настройку статических групп будем производить на примере подключения клиента 1 к multicast-серверу. Multicast-сервер вещает multicast-группы в 10 VLAN и подключен к коммутатору через интерфейс 0/28. Клиент 1 получает multicast-трафик в 10 VLAN, и подключен к интерфейсу коммутатора 0/1.

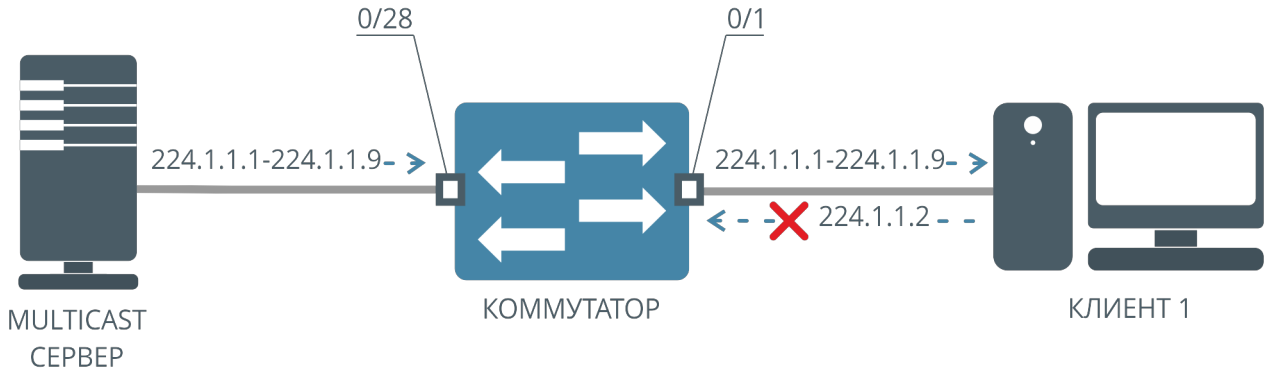

Рисунок 45. Схема работы статических групп

Шаг 1. Конфигурация VLAN

Создаем VLAN 10, в котором будет осуществляться вещание:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 10
(als_sw) (Vlan) #exit
```

Настраиваем VLAN 10 на клиентском интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan participation include 10
(als_sw) (configure) (interface 0/1) #vlan tagging 10
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
```

Настраиваем VLAN 10 на серверном интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #vlan participation include 10
(als_sw) (configure) (interface 0/28) #vlan tagging 10
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
```

Шаг 2. Создание статической группы

Адреса групп можно указывать двумя способами: адрес группы или диапазон адресов. Создаем статическую группу, добавляем в нее диапазон 224.1.1.1-224.1.1.9 в 10 VLAN:

```
(als_sw) #configure
(als_sw) (configure) #set igmp static-group StaticMulticast
(als_sw) (configure) (static-group: "StaticMulticast") #group 224.1.1.1 to 224.
1.1.9 vlan 10
(als_sw) (configure) (static-group: "StaticMulticast") #exit
(als_sw) (configure) #exit
```

Шаг 3. Применение группы на клиентском интерфейсе

Применяем статическую группу на интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #set igmp static-group StaticMulticast
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
```

После данного шага клиентский интерфейс будет получать multicast-трафик статически, вне зависимости от режима передачи multicast-трафика в 10 VLAN. Кроме того, клиенту не нужно подписываться на эти группы, чтобы получить их. Также эти группы не отображаются в списке подписок.

Настройка IGMP Snooping Proxy

IGMP Snooping Proxy — механизм, который заменяет MAC-адрес и IP-адрес источника сообщений "IGMP Membership Report" и "IGMP Leave" на MAC-адрес и IP-адрес коммутатора. IGMP Proxy настраивается только на серверном интерфейсе.

Настройку IGMP Snooping Proxy будем производить на примере подключения клиента 1 к multicast-серверу через два коммутатора. Multicast сервер вещает multicast-группы и подключен к коммутатору 2 через интерфейс 0/28.

Коммутатор 2 соединен с коммутатором 1 через интерфейс 0/27, а коммутатор 1 соединен с коммутатором 2 через интерфейс 0/28. Клиент 1 получает multicast-трафик и подключен к коммутатору 1 через интерфейс 0/1.

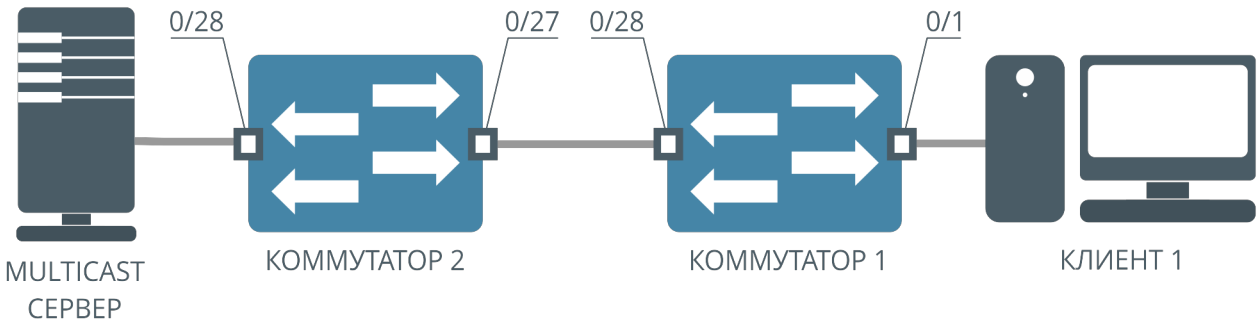

Рисунок 46. Схема работы IGMP Proxy

Шаг 1. Включение службы IGMP Snooping на устройстве

Настройка IGMP Snooping proxy возможна только при включенном IGMP Snooping. Включим IGMP Snooping на коммутаторе 1:

```
(als_sw) #configure
(als_sw) (configure) #set igmp
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #set igmp
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #set igmp mrouter interface
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
```

На коммутаторе 2 также включим IGMP Snooping:

```
(als_sw) #configure
(als_sw) (configure) #set igmp
(als_sw) (configure) #interface 0/27
(als_sw) (configure) (interface 0/27) #set igmp
(als_sw) (configure) (interface 0/27) #exit
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #set igmp mrouter interface
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
```

Шаг 2. Включение IGMP Snooping Proxy

Для включения IGMP Snooping Proxy выполняем команду на серверном интерфейсе коммутатора 1:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #set igmp proxy
(als_sw) (configure) (interface 0/28) #exit
(als_sw) (configure) #exit
```

После данного шага сообщения "IGMP Membership Report" и "IGMP Leave", выходящие с 0/28 интерфейса коммутатора 1, поменяют свои MAC и IP-адреса источника на MAC и IP-адрес коммутатора 1.

Шаг 3. Включение IGMP Snooping Proxy trust

Иногда необходимо, чтобы IGMP Porxy не менял MAC-адрес и IP-адрес в пакетах только с определенных устройств, данный интерфейс помечается как доверенный:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/2
(als_sw) (configure) (interface 0/2) #set igmp proxy trust
(als_sw) (configure) (interface 0/2) #exit
(als_sw) (configure) #exit
```

После данной команды IGMP-пакеты, приходящие на 0/2 интерфейс, не будут менять MAC-адрес и IP-адрес, поскольку этот интерфейс считается доверенным.

Настройка временных характеристик IGMP Snooping

У каждой подписки на multicast-группу в таблице подписок на коммутаторе есть два таймера: Group Membership Interval (GMI) и Query Response Interval (QRI).

Таймер GMI отсчитывает время, после которого подписка клиента, если он ее не обновил, удаляется. Стандартное значение таймера 260 секунд. Если от клиента за время действия таймера GMI приходит "IGMP Membership Report", то подписка обновляется, а значение таймера сбрасывается и отсчет начинается сначала.

Таймер QRI отсчитывает время, которое отводится клиенту на ответ на сообщение multicast-сервера "IGMP Group Specific Query". В момент прихода с сервера этого сообщения на коммутатор таймер запускается, а сообщение направляется клиенту. Если клиент не ответит на сообщение от сервера в течение времени работы таймера, то подписка клиента будет удалена. Стандартное значение таймера равно 10 секундам. Если клиент отвечает на сообщение сервера в установленное время, таймер сбрасывается, подписка остается.

Настройку временных характеристик будем производить на примере подключения клиента 1 к multicast-серверу. Multicast-сервер вещает multicast-группы и подключен к коммутатору через интерфейс 0/28. Клиент 1 получает multicast-трафик и подключен к интерфейсу коммутатора 0/1:

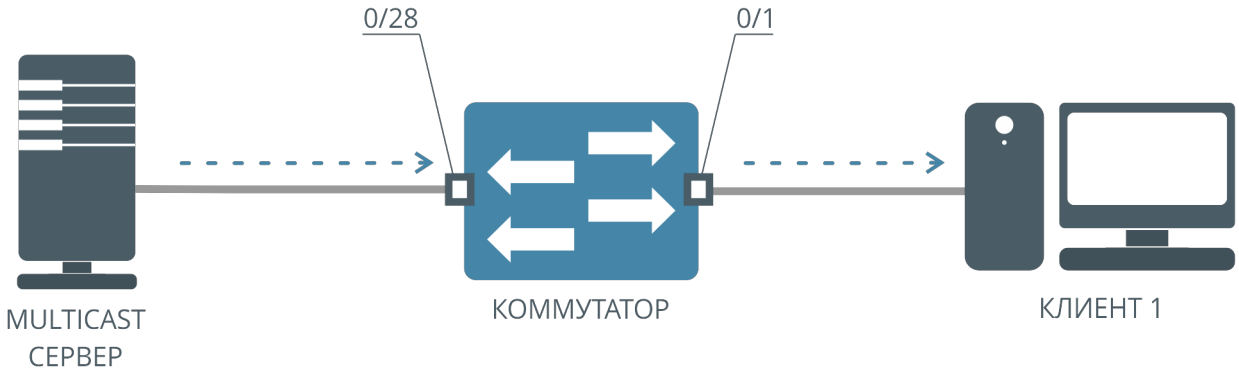

Рисунок 47. Схема подключения с одним клиентом

Шаг 1. Включение службы IGMP Snooping на устройстве

Настройка временных характеристик IGMP Snooping возможна только при включенном IGMP Snooping. Смотрите пункт "Настройка IGMP Snooping на интерфейсах", шаги 1-5.

Шаг 2. Настройка времени подписки клиента на multicast-группу

Для настройки времени подписки клиента на multicast-группу (GMI) выполняем команду:

```
(als_sw) #configure
(als_sw) (configure) #set igmp groupmembership-interval 250
(als_sw) (configure) #exit
```

После данного шага подписка клиента на multicast-группу будет удалена, если клиент не пришлет "IGMP Membership Report" за отведенный интервал в 250 секунд.

Шаг 3. Настройка времени ответа на запрос от multicast-сервера

Для настройки времени ответа клиента на запрос от multicast-сервера (QRI) выполняем команду:

```
(als_sw) #configure
(als_sw) (configure) #set igmp maxresponse 5
(als_sw) (configure) #exit
```

После данного шага подписка клиента на multicast-группу будет удалена, если клиент не пришлет "IGMP Membership Report" в ответ на "IGMP Specific Query" за отведенный интервал в 5 секунд.

Просмотр состояния таймеров

Просмотреть текущие значения таймеров для подписок можно командой:

```
(als_sw) #show igmpsnooping groups all

  VlanId  Multicast Group  Version  Iface   Uptime         GMI (sec)  QRI (sec)

  ------  ---------------  -------  ------  -------------  ---------  ---------

  10      224.0.21.11      IGMPv2   0/1     0d, 00:00:48         250          5

  20      224.1.21.87      IGMPv2   0/1     0d, 00:00:48         250          5

  20      224.1.21.121     IGMPv2   0/2     0d, 00:00:48         250          5
```

Настройка IGMP fast-leave на клиентском интерфейсе

При получении сообщения коммутатором "IGMP Leave" от клиента коммутатор не сразу удаляет подписку. Согласно протоколу IGMP, в ответ на сообщение клиента "IGMP Leave", multicast-сервер должен послать "IGMP Specific Query". Если клиент игнорирует "IGMP Specific Query" (не отправляет на него "IGMP Join"), то коммутатор удаляет подписку.

Существует возможность удалять подписку сразу после получения "IGMP Leave". Данный механизм получил название "fast-leave".

Шаг 1. Включение службы IGMP Snooping на устройстве

Включение "fast-leave" возможно только при включенном IGMP Snooping. Смотрите пункт "Настройка IGMP Snooping на интерфейсах", шаги 1-5.

Шаг 2. Включение настройки IGMP fast-leave на клиентском интерфейсе

Для включения настройки IGMP "fast-leave" на клиентском интерфейсе выполняем команду:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #set igmp fast-leave
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
```

После данного шага, если клиент за интерфейсом 0/1 пошлет "IGMP Leave", подписка клиента на группу будет удалена сразу.

Настройка проверки опции Router Alert

Router Alert — опция IPv4-заголовка, указывающая на то, что multicast-роутеры должны обрабатывать данный пакет. Данная опция должна присутствовать во всех сообщениях "IGMP Membership Report" и "IGMP Leave". По стандарту пакет без данной опции должен передаваться согласно таблице подписок и не должен обрабатываться как IGMP-пакет. Устаревшие сетевые устройства могут отсылать сообщения без опции Router Alert, поэтому есть возможность настроить поведение коммутатора при получении "IGMP Membership Report" и "IGMP Leave" без опции Router Alert.

Включение проверки опции Router Alert приведет к тому, что все "IGMP Membership Report" и "IGMP Leave" без опции будут передаваться согласно таблице подписок. При этом запрос на подключение или отключение multicast-группы, содержащейся в данном пакете, не будет обработан. При выключенной проверке опции Router Alert будут обработаны все "IGMP Membership Report" и "IGMP Leave" вне зависимости от наличия опции в пакете.

Шаг 1. Включение службы IGMP Snooping на устройстве

Настройка проверки опции Router Alert возможна только при включенном IGMP Snooping. Смотрите пункт "Настройка IGMP Snooping на интерфейсах", шаги 1-5.

Шаг 2. Включение проверки опции Router Alert

Для включения проверки опции Router Alert выполняем команду:

```
(als_sw) #configure
(als_sw) (configure) #set igmp router-alert
(als_sw) (configure) #exit
```

После данного шага IGMP-сообщения без этой опции обрабатываться не будут.

Базовые настройки IGMP Querier

IGMP Querier на коммутаторе может понадобиться в том случае, если multicast-сервер по каким-либо причинам не может отсылать в сеть сообщения "IGMP Query". В таком случае роль источника сообщений "IGMP Query" может взять на себя коммутатор. Для включения IGMP Querier включение службы IGMP Snooping не требуется.

Базовые настройки механизма IGMP Querier будем производить на примере подключения клиента 1 к multicast-серверу. Multicast-сервер вещает multicast-группы и подключен к коммутатору через интерфейс 0/28. Клиент 1 получает multicast-трафик и подключен к интерфейсу коммутатора 0/1:

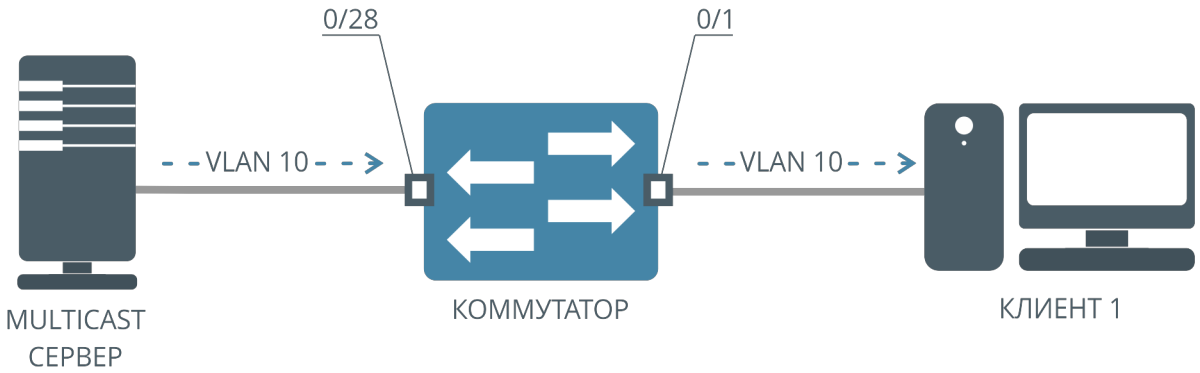

Рисунок 48. Схема работы IGMP Querier

Шаг 1. Настройка VLAN на устройстве

Создаем VLAN 10, в котором будет осуществляться вещание:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 10
(als_sw) (Vlan) #exit
```

Настраиваем VLAN 10 на серверном и клиентском интерфейсах:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1,0/28
(als_sw) (configure) (interface 0/1,0/28) #vlan participation include 10
(als_sw) (configure) (interface 0/1,0/28) #vlan tagging 10
(als_sw) (configure) (interface 0/1,0/28) #exit
(als_sw) (configure) #exit
```

Шаг 2. Включение механизма IGMP Querier на устройстве

Включаем IGMP Querier на устройстве:

```
(als_sw) #configure
(als_sw) (configure) #set igmp querier
(als_sw) (configure) #exit
```

Шаг 3. Настройка IGMP Querier на VLAN

Включаем IGMP Querier на 10 VLAN:

```
(als_sw) #vlan database
(als_sw) (Vlan) #set igmp querier 10
(als_sw) (Vlan) #exit
```

Шаг 4. Настройка адреса IGMP Querier

Настраиваем адрес IGMP Querier:

```
(als_sw) #configure
(als_sw) (configure) #set igmp querier address 172.17.1.1
(als_sw) (configure) #exit
```

После данного шага через каждые 60 секунд с клиентских интерфейсов будут отправляться сообщения "IGMP General Query" с меткой VLAN 10 и IP-адресом источника 172.17.1.1.

Шаг 5. Настройка временных характеристик IGMP Querier

В некоторых случаях может понадобиться смена интервала отправки сообщений службой IGMP Querier. Выбор значения зависит от абонентских устройств, установленных в сети доступа.

Для смены интервала отправки сообщений "IGMP Query" выполняем команду:

```
(als_sw) #configure
(als_sw) (configure) #set igmp querier query-interval 30
(als_sw) (configure) #exit
```

После данного шага IGMP Querier будет отправлять сообщения "IGMP Query" каждые 30 секунд. Значение по умолчанию для данного интервала равно 60 секундам.

Дополнительные настройки IGMP Querier

По стандарту IGMP Querier должен прекратить рассылку сообщений "IGMP Query", если эти сообщения приходят от вышестоящего оборудования. Коммутатор может находиться в двух состояниях:

- Querier — коммутатор осуществляет периодическую рассылку "IGMP General Query" сообщений и анализ входящих IGMP сообщений на серверных интерфейсах. Если коммутатор получит сообщение "IGMP Query" (general или specific), то он перейдет в состояние "Non-Querier";
- Non-Querier — коммутатор прекращает рассылку сообщений "IGMP Query" на некоторый период времени. Если в течении этого периода коммутатор не получит сообщений "IGMP Query", он переходит в состояние "Querier".

Дополнительные настройки механизма IGMP Querier будем производить на примере подключения клиента 1 к multicast-серверу. Multicast-сервер вещает multicast-группы и подключен к коммутатору через интерфейс 0/28. Клиент 1 получает multicast-трафик и подключен к интерфейсу коммутатора 0/1:

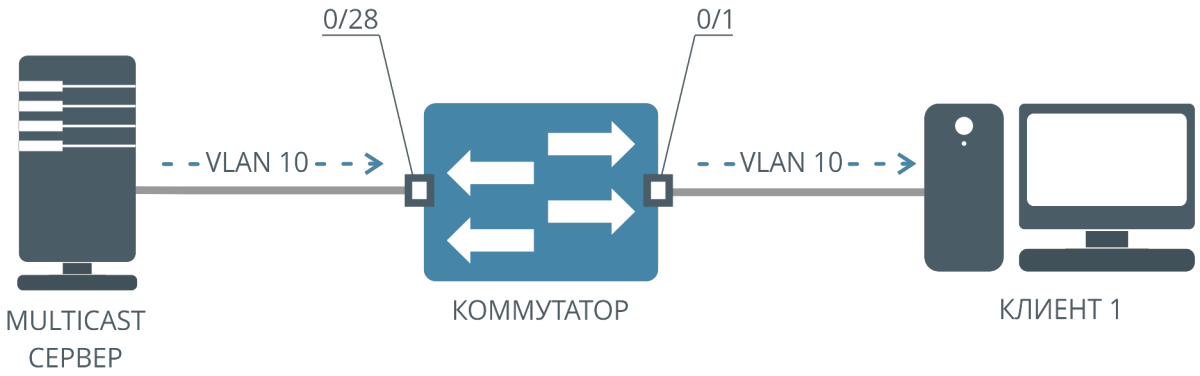

Рисунок 49. Схема работы IGMP Querier

Шаг 1. Базовые настройки IGMP Querier

Для добавления дополнительных настроек IGMP Querier необходимо сначала добавить базовые настройки IGMP Querier. Смотрите пункт "Базовые настройки IGMP Querier".

Шаг 2. Настройка дополнительных временных характеристик IGMP Querier

Для смены значения стандартного интервала времени, после которого коммутатор в состоянии "Non-Querier" перейдет в состояние "Querier", выполняем команду:

```
(als_sw) #configure
(als_sw) (configure) #set igmp querier timer expiry 70
(als_sw) (configure) #exit
```

После данного шага коммутатор в состоянии "Non-Querier" перейдет в состояние "Querier" через 70 секунд. Стандартное значение интервала — 60 секунд.

Шаг 3. Настройка отключения отправки General Query

В некоторых ситуациях требуется отключить отправку IGMP General Query, при этом сохранив функцию отправки Specific Query на абонентские интерфейсы. Ниже приведен пример отключения отправки IGMP General Query в VLAN 1.

```
(als_sw) #vlan database
(als_sw) (Vlan) #no set igmp querier 1 general
(als_sw) (Vlan) #exit
```

Шаг 4. Настройка отправки General Specific Query

По умолчанию Querier в ответ на IGMP Leave шлет один IGMP Specific Query.

Отправка нескольких IGMP Query производится командой. После выполнения команды, IGMP Query будет отправлять 2 IGMP Specific Query в ответ на IGMP Leave.

```
(als_sw) #configure
(als_sw) (configure) #set igmp querier last-member-query-count 2
(als_sw) (configure) #exit
```

Также возможно настроить интервал между отправками IGMP Specific Query. Интервал между отправками в примере 1 секунда.

```
(als_sw) #configure
(als_sw) (configure) #set igmp querier last-member-query-interval 1
(als_sw) (configure) #exit
```

Шаг 5. Настройка значения 802.1p

При генерации пакетов службы IGMP используют общий механизм отправки пакетов. Значение метки 802.1p в отправляемых пакетах можно установить с помощью команды:

```
(als_sw) #configure
(als_sw) (configure) #set igmp cos 6
(als_sw) (configure) #exit
```

Настроенное значение 802.1p будет записываться во все отправляемые службами IGMP пакеты, кроме пакетов, обрабатываемых службой MVR. Для MVR значение 802.1p настраивается отдельно другой командой, описанной в разделе настройки MVR.

Автоматическая подписка на каналы

В некоторых случаях необходимо организовать быстрое переключение на определенные каналы. Один из вариантов организации быстрого подключения заключается в постоянной трансляции группы каналов на оборудование уровня доступа. В этом случае при подписке клиента на канал он сразу же начинает передаваться клиенту, без задержки, связанной с передачей запроса клиента multicast-серверу.

Для организации постоянного вещания определенной группы каналов можно использовать специальную настройку коммутатора. При этой настройке коммутатор самостоятельно подписывается на указанные в конфигурации каналы и поддерживает эту подписку.

Схема работы коммутатора с настроенной автоматической подпиской на каналы:

Рисунок 50. Схема работы автоматической подписки на группы

Шаг 1. Создание группы подписок

Как правило, автоматическая подписка настраивается на несколько каналов сразу. Каналы можно указывать в виде их multicast-адресов, либо в виде диапазона адресов, указав начальный и конечный multicast-адрес.

Для создания группы используется команда:

```
(als_sw) #configure
(als_sw) (configure) #set igmp join-group "GroupName"
(als_sw) (configure) (join-group: "GroupName") #
```

После выполнения этой команды CLI перейдет в контекст настройки группы. В этом контексте можно добавить адреса каналов:

```
(als_sw) (configure) (join-group: "GroupName") #group 224.1.1.123 to 224.1.1.13
8 vlan 1
```

В данном примере в группу с именем "GroupName" добавлен диапазон адресов с 224.1.1.123 по 224.1.1.138, подписка будет осуществляться во VLAN 1.

Также можно добавить в группу отдельный адрес:

```
(als_sw) (configure) (join-group: "GroupName") #group 224.1.1.120 vlan 1
```

Все добавленные в группу адреса работают совместно. В случае, если один из адресов повторяется дважды (например, указаны перекрывающиеся интервалы адресов), в группе будет только один адрес.

Шаг 2. Включение статической подписки на интерфейсе

Для того, чтобы определенный интерфейс коммутатора поддерживал подписку на указанные в группе адреса, нужно включить на интерфейсе автоматическую подписку с указанием имени группы.

Для этого служит команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/8
(als_sw) (configure) (interface 0/8) #set igmp join-group "GroupName"
```

После выполнения этой команды коммутатор будет периодически отправлять на 0/8 интерфейс IGMP Membership Report (Join) на указанные в группе адреса, тем самым поддерживая подписку на каналы. Коммутатор также отвечает на приходящие IGMP Query с учетом настроенной автоматической подписки. При включенном IGMP Snooping при подписке клиента на определенный канал из группы multicast-трафик этого канала сразу же начинает передаваться клиенту, а сами пакеты Join и Leave от клиентов не передаются multicast-серверу. Это позволяет организовать быстрое переключение каналов у абонентов.

Режим IGMP Proxy Reporting

IGMP Proxy Reporting позволяет существенно уменьшить число управляющих IGMP-пакетов в сети провайдера. При подписке клиента на канал коммутатор проверяет, была ли создана подписка на этот канал раннее. Если нет, то коммутатор формирует IGMP Join пакет и от своего имени отсылает его на multicast-сервер. Если подписка уже имеется, то коммутатор не отправляет IGMP Join на multicast-сервер, а лишь начинает передачу multicast-трафика в соответствующий интерфейс.

Схема работы коммутатора в режиме IGMP Proxy Reporting при подписке двух клиентов на одну multicast-группу:

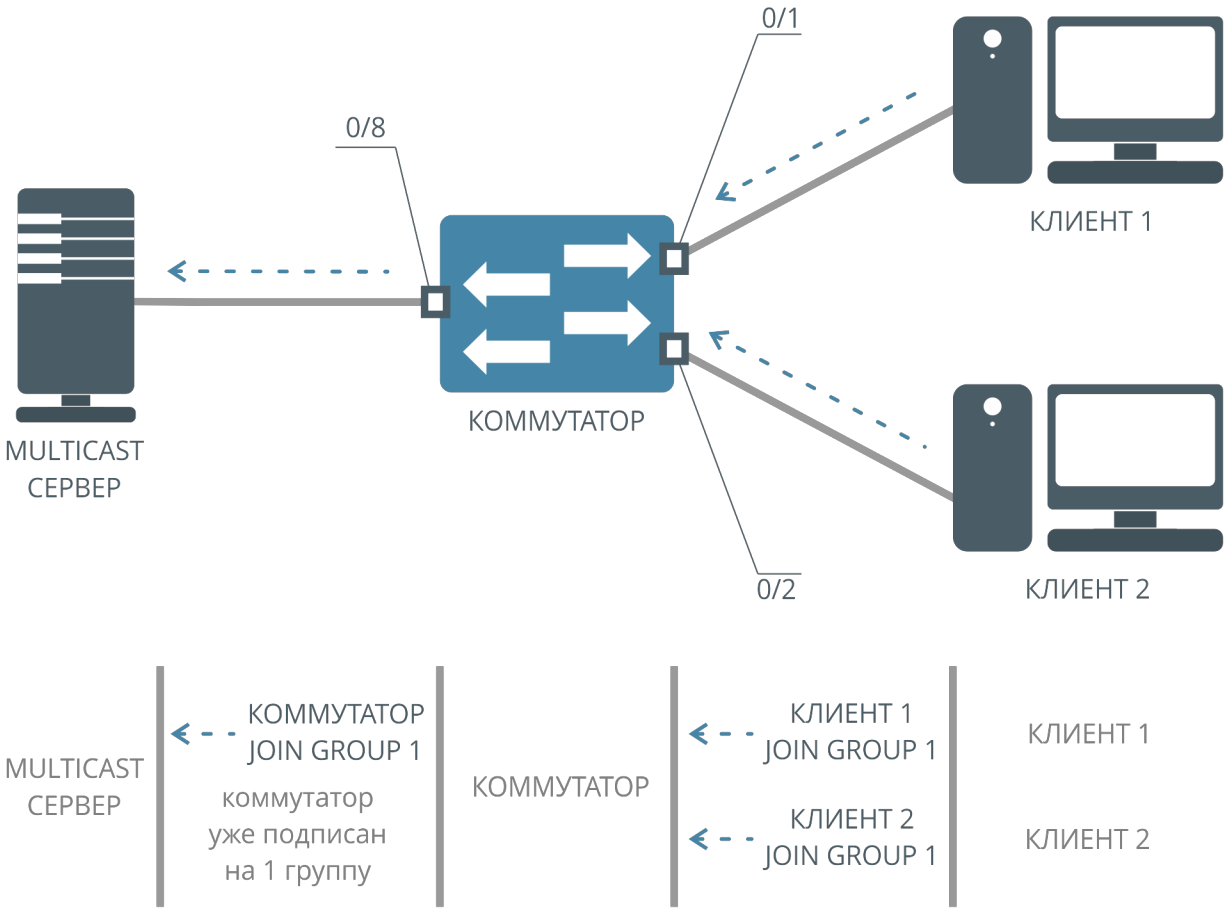

Рисунок 51. Режим IGMP Proxy Reporting, Join

При получении IGMP Leave от клиента коммутатор отключает пересылку Multicast-трафика клиенту (сразу же или спустя некоторое время, в зависимости от режима Fast Leave на интерфейсе клиента). Отправка IGMP Leave на сервер от лица коммутатора будет произведена только в том случае, если ни один из клиентов не подписан на этот канал.

Схема работы коммутатора в режиме IGMP Proxy Reporting при отписке клиентов:

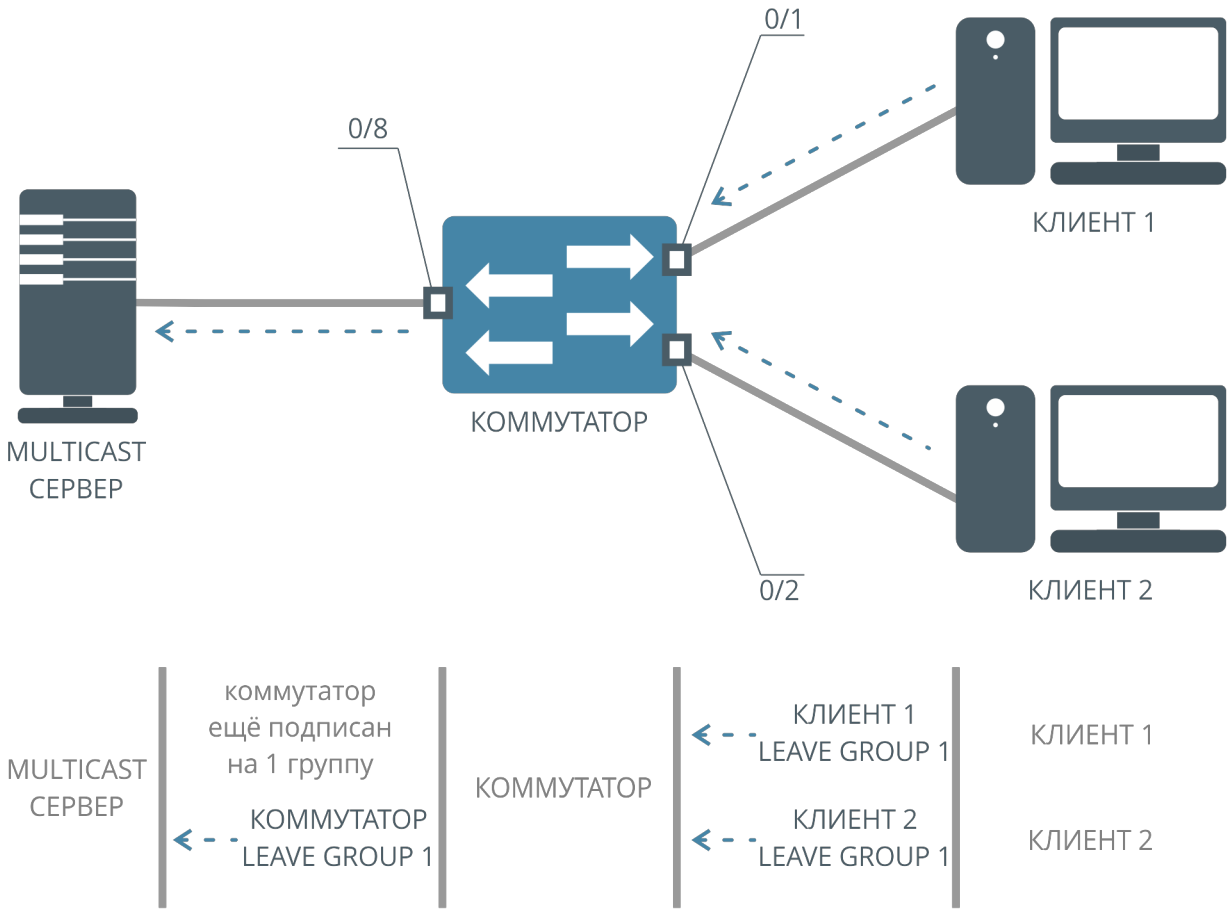

Рисунок 52. Режим IGMP Proxy Reporting, Leave

Также меняется реакция коммутатора на пакеты IGMP Query. При получении такого пакета с серверного интерфейса коммутатор самостоятельно отвечает на них на основании подписок клиентов, существующих в данный момент. Для совместимости со старыми приставками и оборудованием пакеты Query также пересылаются клиентам.

Схема работы коммутатора в режиме IGMP Proxy Reporting при получении IGMP Query с серверного интерфейса:

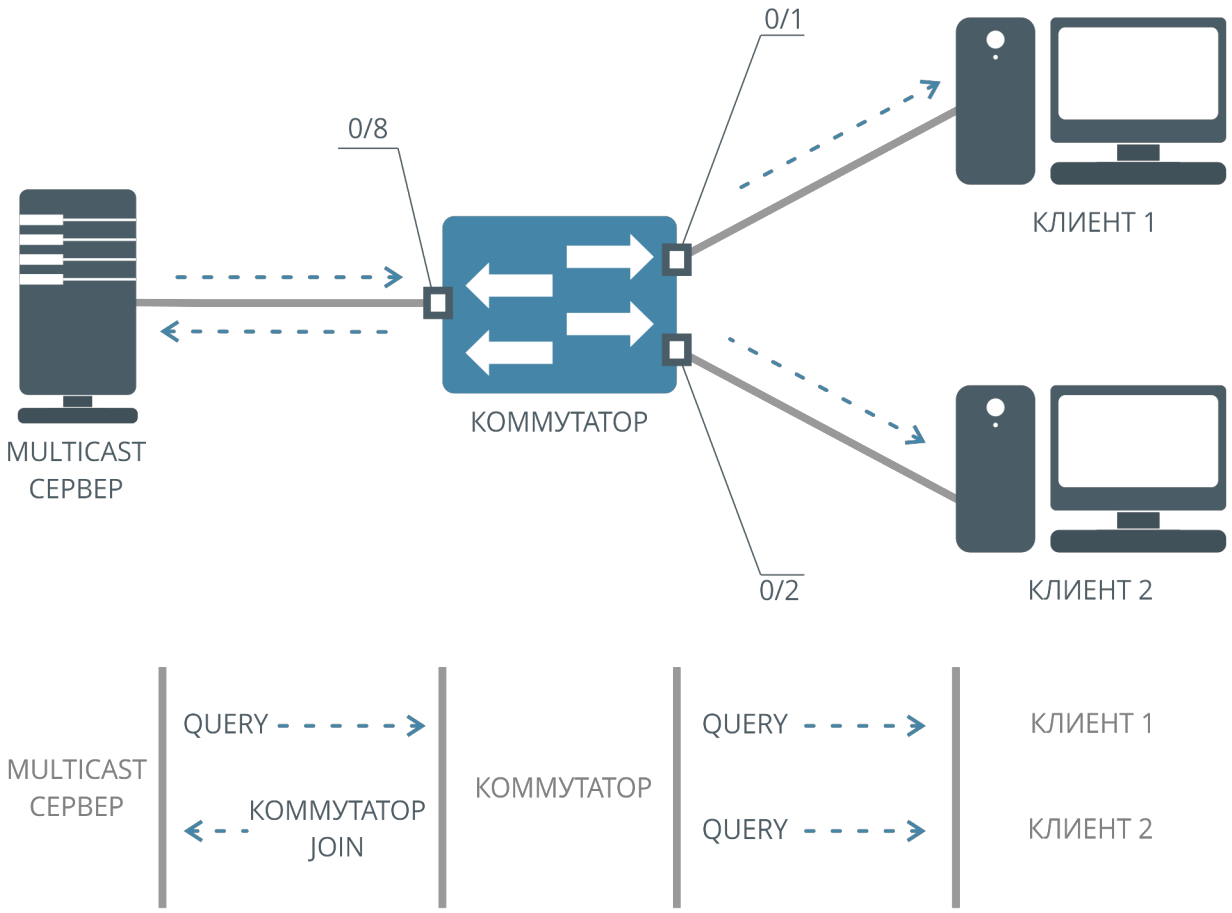

Рисунок 53. Режим IGMP Proxy Reporting, Query

Шаг 1. Базовая настройка IGMP Snooping

Для того, чтобы активировать режим IGMP Proxy Reporting на коммутаторе, необходимо настроить IGMP Snooping на интерфейсе или на VLAN.

В примере ниже IGMP Snooping включен на двух интерфейсах (0/1 и 0/2), также указан один серверный интерфейс (0/8):

```
(als_sw) #configure
(als_sw) (configure) #set igmp
(als_sw) (configure) #interface 0/1-0/2
(als_sw) (configure) (interface 0/1-0/2) #set igmp
(als_sw) (configure) (interface 0/1-0/2) #set igmp fast-leave
(als_sw) (configure) (interface 0/1-0/2) #exit
(als_sw) (configure) #interface 0/8
(als_sw) (configure) (interface 0/8) #set igmp mrouter interface
(als_sw) (configure) #exit
(als_sw) #exit
```

Шаг 2. Включение режима IGMP Proxy Reporting

Для включения режима используется команда:

```
(als_sw) #configure
(als_sw) (configure) #set igmp proxy-reporting
```

Шаг 3. Настройка Source IP для управляющих пакетов IGMP

В режиме IGMP Proxy Reporting коммутатор будет отправлять управляющие пакеты IGMP от своего имени, то есть со своим MAC-адресом и IP-адресом.

Назначить другой Source IP-адрес, отличный от собственного IP-адреса коммутатора, можно командой:

```
(als_sw) #configure
(als_sw) (configure) #set igmp proxy-reporting source-ip 172.17.1.10
```

В этом случае Source IP-адрес управляющих IGMP-пакетов, отправляемых коммутатором, будет равен 172.17.1.10.

15.3. Примеры типовых настроек

Приведем примеры настроек IGMP на 28-портовом коммутаторе АЛСиТЕК. Первые 24 интерфейса (0/1-0/24) будем считать абонентскими, то есть теми интерфейсами, за которыми расположены абоненты. Интерфейсы 0/25-0/28 будем считать серверными интерфейсами, за которыми находится вышестоящее оборудование.

Пример настройки IGMP Snooping на интерфейсах

В примере ниже демонстрируется настройка IGMP Snooping на портах. Multicast-трафик идет к абонентам в 10 и 20 VLAN и доступен только тем абонентам, которые явно запросили multicast-группы в этих VLAN по протоколу IGMP.

Конфигурация:

```
vlan database
vlan 10,20
exit
configure
mcast_vfm 10 forward_registered
mcast_vfm 20 forward_registered
set igmp
interface 0/1-0/24
set igmp
vlan participation include 10,20
vlan tagging 10,20
exit
interface 0/25-0/28
vlan participation include 10,20
vlan tagging 10,20
set igmp mrouter interface
exit
exit
```

Пример настройки IGMP Snooping на VLAN

В примере демонстрируется настройка IGMP Snooping на VLAN. Multicast-трафик идет к абонентам в 10 VLAN и доступен только тем абонентам, которые явно запросили multicast-группы в этом VLAN по протоколу IGMP.

Конфигурация:

```
vlan database
vlan 10
set igmp 10
exit
configure
mcast_vfm 10 forward_registered
set igmp
interface 0/1-0/24
vlan participation include 10
vlan tagging 10
exit
interface 0/25-0/28
vlan participation include 10
vlan tagging 10
set igmp mrouter 10
exit
exit
```

Пример настройки IGMP Querier

В примере приведены настройки IGMP Querier. Сообщения "IGMP General Query" отсылаются с интервалом 65 секунд на интерфейсы 0/1-0/24 в 10 VLAN. В качестве IP-адреса источника в сообщениях "IGMP General Query" указан адрес 172.17.1.2.

Конфигурация:

```
vlan database
vlan 10
set igmp querier 10
exit
configure
set igmp querier
set igmp querier address 172.17.1.2
set igmp querier timer expiry 65
interface 0/1-0/24
vlan participation include 10
vlan tagging 10
exit
exit
```

15.4. Типовые вопросы и ошибки

В: В конфигурации отображается команда "set igmp mvr 1". Включен ли механизм MVR?
О: Нет, не включен. Данная команда задает MVR VLAN, для включения механизма требуется добавить команду "set igmp mvr".

Пример: MVR отключен, в качестве MVR VLAN указан VLAN 1 (состояние по умолчанию).

Конфигурация будет такой:

```
configure
set igmp mvr 1
exit
```

Пример: MVR включен, в качестве MVR VLAN указан VLAN 1.

```
configure
set igmp mvr
set igmp mvr 1
exit
```

ГЛАВА 16. PPPOE INTERMEDIATE AGENT

16.1. Введение в PPPoE

PPPoE (Point-to-point protocol over Ethernet) — сетевой протокол канального уровня передачи кадров PPP через Ethernet. Стандарт PPPoE описан в документе RFC 2516.

В процессе установления PPPoE-сессии (PPPoE Discovery, PPPoED) клиент и сервер обмениваются следующими пакетами:

- PADI (PPPoE Active Discovery Initiation) — отправляется клиентом для обнаружения всех доступных PPPoE-серверов в рамках сети;
- PADO (PPPoE Active Discovery Offer) — отправляется сервером в ответ на PADI, является предложением установить сессию с данным сервером;
- PADR (PPPoE Active Discovery Request) — отправляется клиентом определенному PPPoE-серверу для установления сессии;
- PADS (PPPoE Active Discovery Session-confirmation) — отправляется сервером в ответ на PADR клиента, является подтверждением установления сессии;
- PADT (PPPoE Active Discovery Termination) — отправляется сервером или клиентом, информируя другую сторону о завершении сессии.

PPPoE Snooping

PPPoE Snooping — функция коммутатора, предназначенная для защиты от атак с использованием протокола PPPoE. PPPoE Snooping позволяет защитить клиента от получения адреса от ненадежного PPPoE-сервера. Для корректной работы PPPoE Snooping интерфейсы коммутатора делятся на две группы — доверенные (или trust) и недоверенные (untrust). При настройке коммутатора доверенными интерфейсами принято считать интерфейсы, за которыми находится доверенный PPPoE-сервер. Остальные интерфейсы считаются недоверенными.

PPPoE Snooping обрабатывает только PPPoED-пакеты, которыми обмениваются PPPoE-сервер и клиент.

Пример работы PPPoE Snooping изображен на схеме ниже:

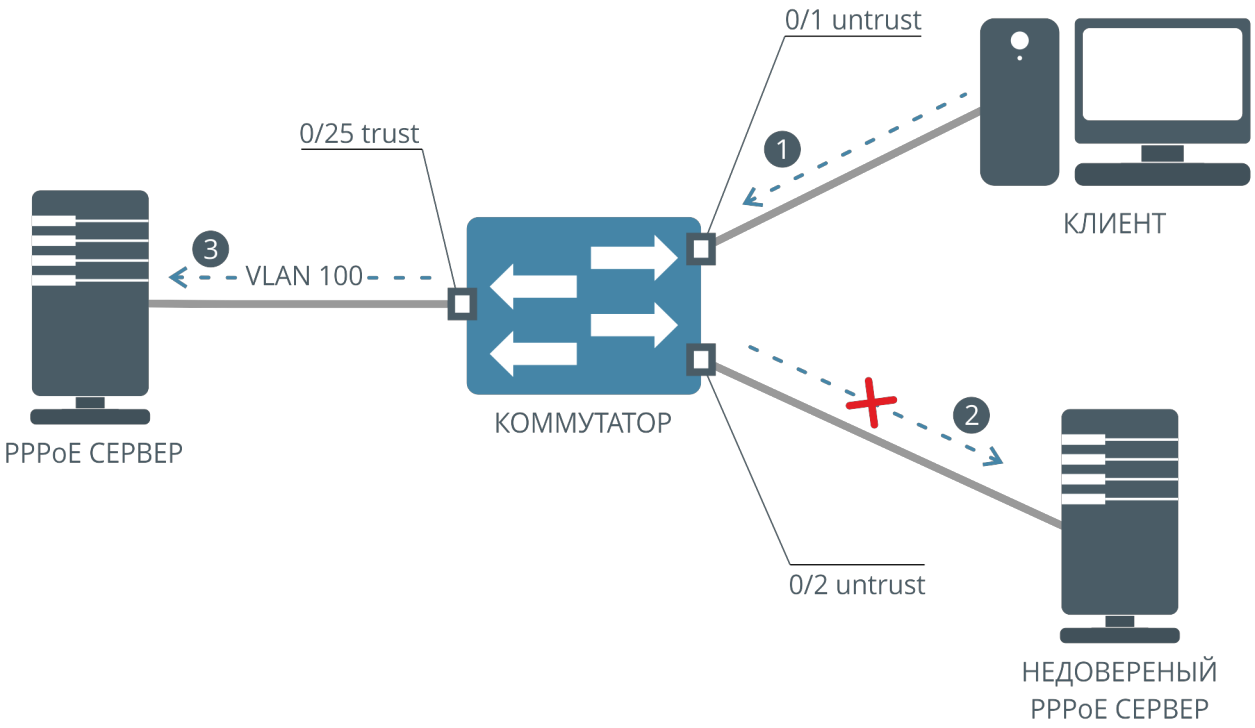

Рисунок 54. Схема работы PPPoE Snooping

1. Клиент отправляет запрос PADI для обнаружения доступных PPPoE-серверов.
2. Отправленный клиентом запрос PADI является широковещательным, но благодаря PPPoE Snooping он не попадает на недоверенный PPPoE-сервер за интерфейсом 0/2, а перенаправляется на доверенный интерфейс коммутатора 0/25.
3. Клиентский запрос PADI доходит до доверенного PPPoE-сервера. Сервер отвечает на него, после этого устанавливается PPPoE-сессия с доверенным сервером.

PPPoE Intermediate Agent

PPPoE Intermediate Agent предназначен для идентификации канала доступа абонента. Документ TR-101 предлагает использовать его для помощи BRAS в авторизации абонента при организации уровня доступа по протоколу PPPoE.

PPPoE IA перехватывает служебные пакеты, которыми клиент и сервер (BRAS) обмениваются в процессе установления PPPoE-сессии. Идентификация канала доступа осуществляется при помощи добавления определенного тега (опции) в пакеты, отправляемые клиентом (PADI, PADR), на базе которых BRAS может авторизовать или нет запрашивающего доступ клиента.

Ответы, отправляемые сервером (PADO, PADS), а также пакет, сообщающий о завершении сессии (PADT, отправляется и клиентом и сервером) по TR-101 не должны содержать данного тега. Но если тег по каким-то причинам присутствует, коммутатор удалит его из пакетов. Клиенту не должна попасть информация об организации уровня доступа провайдера.

TR-101 предлагает использовать тег 0x0105 (Vendor-Specific) с Vendor ID 0x00000DE9 (ID организации DSL Forum). Тег содержит 2 поля:

- Agent Circuit ID — произвольная строка, как правило, содержащая описание интерфейса, к которому подключен абонент;
- Agent Remote ID — произвольная строка, как правило, определяющая сам узел доступа.

16.2. Настройка PPPoE Snooping на коммутаторах АЛСиТЕК

Служба PPPoE Snooping настраивается в несколько шагов. Прежде всего необходимо включить глобально службу, затем назначить службе номера VLAN, в которых необходимо следить за PPPoE-трафиком. После этого необходимо указать доверенный интерфейс, за которым находится PPPoE-сервер. Доверенных интерфейсов может быть несколько. Все остальные интерфейсы будут считаться недоверенными (untrust), дополнительная настройка недоверенных интерфейсов на коммутаторе не требуется.

Через недоверенный интерфейс коммутатора могут проходить только клиентские PPPoE-пакеты (PADI, PADR, PADT). Эти пакеты будут направлены только на доверенные интерфейсы.

Пошаговая настройка PPPoE Snooping

Шаг 1. Предварительные настройки коммутатора

Создание VLAN 100:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 100
(als_sw) (Vlan) #exit
```

Настройка VLAN 100 на клиентских интерфейсах:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1-0/2
(als_sw) (configure) (interface 0/1-0/2) #vlan participation include 100
(als_sw) (configure) (interface 0/1-0/2) #vlan pvid 100
(als_sw) (configure) (interface 0/1-0/2) #exit
(als_sw) (configure) #exit
```

Настройка VLAN 100 на серверном интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/25
(als_sw) (configure) (interface 0/25) #vlan participation include 100
(als_sw) (configure) (interface 0/25) #vlan tagging 100
(als_sw) (configure) (interface 0/25) #exit
(als_sw) (configure) #exit
```

Шаг 2. Включение службы PPPoE Snooping на устройстве

Для включения службы используем команду:

```
(als_sw) #configure
(als_sw) (configure) #pppoe
(als_sw) (configure) #exit
```

Шаг 3. Задание VLAN, на которых включен PPPoE Snooping

Далее, указываем один или несколько VLAN, на которых должна быть включена служба PPPoE Snooping:

```
(als_sw) #configure
(als_sw) (configure) #pppoe 100
(als_sw) (configure) #exit
```

В примере PPPoE Snooping включается на VLAN 100. Может быть указано несколько VLAN.

Шаг 4. Назначение доверенных интерфейсов для PPPoE Snooping

Для указания доверенного интерфейса нужно выполнить команду:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/25
(als_sw) (configure) (interface 0/25) #pppoe trust
(als_sw) (configure) (interface 0/25) #exit
(als_sw) (configure) #exit
```

В примере интерфейс 0/25 переводится в состояние доверенного для PPPoE Snooping. Доверенных интерфейсов может быть несколько.

После описанных выше шагов конфигурация коммутатора будет следующей:

```
(als_sw) #show running-config
vlan database
vlan 100
exit
configure
pppoe
pppoe 100
interface 0/1
vlan pvid 100
vlan participation include 100
exit
interface 0/2
vlan pvid 100
vlan participation include 100
exit
interface 0/25
vlan participation include 100
vlan tagging 100
pppoe trust
exit
exit
```

Просмотр серверов

Коммутатор сохраняет данные о всех серверах PPPoE, с которыми клиенты обмениваются сообщениями. Список PPPoE-серверов можно посмотреть следующей командой:

```
(als_sw) #show pppoe servers

MAC Address          Interface    VLAN ID
-----------------    ---------    -------
00:13:AA:00:00:01    0/25         100
```

В данном примере за интерфейсом 0/25 находится сервер с MAC-адресом 00:13:AA:00:00:01.

16.3. Настройка PPPoE IA на коммутаторах АЛСиТЕК

Служба PPPoE IA работает совместно с PPPoE Snooping и настраивается в несколько шагов. Прежде всего необходимо включить глобально PPPoE Snooping, затем назначить службе PPPoE Snooping номера VLAN, в которых необходимо следить за PPPoE-трафиком. После этого необходимо указать доверенный интерфейс, за которыми находится PPPoE-сервер. Затем необходимо включить глобально PPPoE IA и настроить форматные строки для подопций circuit-id и remote-id.

Пошаговая настройка PPPoE IA

Рассмотрим настройку на примере простой схемы:

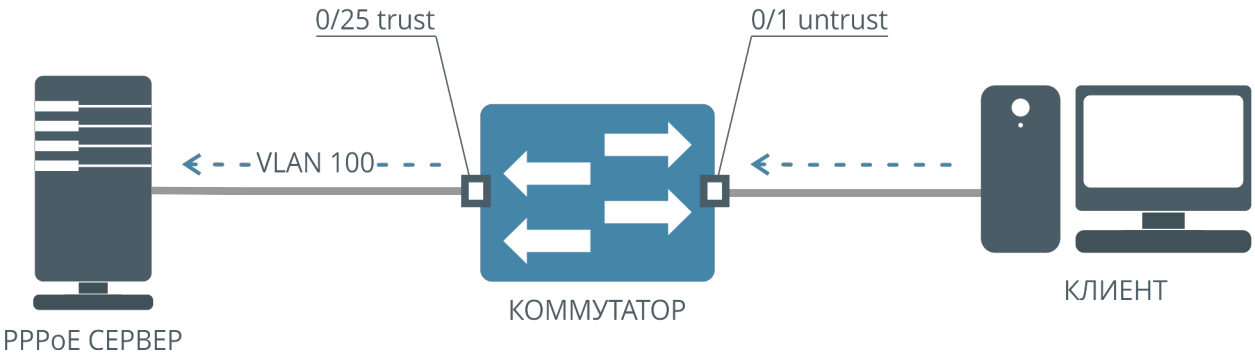

Рисунок 55. Упрощенная схема для иллюстрации работы PPPoE IA

Шаг 1. Предварительные настройки коммутатора

Создание VLAN 100:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 100
(als_sw) (Vlan) #exit
```

Настройка VLAN 100 на клиентском интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan participation include 100
(als_sw) (configure) (interface 0/1) #vlan pvid 100
(als_sw) (configure) (interface 0/1) #exit
(als_sw) (configure) #exit
```

Настройка VLAN 100 на серверном интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/25
(als_sw) (configure) (interface 0/25) #vlan participation include 100
(als_sw) (configure) (interface 0/25) #vlan tagging 100
(als_sw) (configure) (interface 0/25) #exit
(als_sw) (configure) #exit
```

Шаг 2. Настройка службы PPPoE Snooping на устройстве

Глобальное включение службы PPPoE:

```
(als_sw) #configure
(als_sw) (configure) #pppoe
(als_sw) (configure) #exit
```

Настройка работы службы PPPoE в определенных VLAN:

```
(als_sw) #configure
(als_sw) (configure) #pppoe 100
(als_sw) (configure) #exit
```

Указание доверенного интерфейса:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/25
(als_sw) (configure) (interface 0/25) #pppoe trust
(als_sw) (configure) (interface 0/25) #exit
(als_sw) (configure) #exit
```

Шаг 3. Глобальное включение PPPoE IA

Для глобального включения PPPoE IA необходимо выполнить команду:

```
(als_sw) #configure
(als_sw) (configure) #pppoe frmtstr enable
(als_sw) (configure) #exit
```

Шаг 4. Настройка форматных строк для полей PPPoE IA

Настройка подопции circuit-id выполняется командой:

```
(als_sw) #configure
(als_sw) (configure) #pppoe frmtstr "SomeStringWithLexems"
(als_sw) (configure) #exit
```

Настройка подопции remote-id выполняется командой:

```
(als_sw) #configure
(als_sw) (configure) #pppoe remoteid "SW1"
(als_sw) (configure) #exit
```

После описанных выше настроек конфигурация коммутатора будет следующей:

```
vlan database
vlan 100
exit
configure
pppoe
pppoe 100
pppoe frmtstr enable
pppoe frmtstr "SomeStringWithLexems"
pppoe remoteid "SW1"
interface 0/1
vlan pvid 100
vlan participation include 100
exit
interface 0/25
vlan participation include 100
vlan tagging 100
pppoe trust
exit
exit
```

Лексемы

Для возможности задания произвольных строк подопций используется механизм лексем. Лексема — это специальный набор символов, которые при вставке форматной строки в пакет будут заменены на некоторое реальное значение, специфичное для данного пакета или коммутатора. С помощью лексем в пакет можно поместить некоторую информацию для сервера, на основании которой он сможет принять решение по обработке данного пакета.

В форматных строках допускается использовать печатные символы ASCII (0x20-0x7e), за исключением символа двойных кавычек (") и обратного слеша (\):

```
 !#$%&#38;'()*+,-./
0123456789:;<=>?
@ABCDEFGHIJKLMNO
PQRSTUVWXYZ[]^_
`abcdefghijklmno
pqrstuvwxyz{|}~
```

Символу "$" при этом отводится роль служебного, он используется для обозначения лексем. Лексемы начинаются со служебного символа "$", далее может идти модификатор длины (необязательный параметр), после чего идет символ (буква), обозначающая лексему. Список лексем приведен ниже. Модификатор используется для задания разрядности значения. При этом недостающие разряды дополняются ведущими нулями.

Допустимо использование только 0 в качестве ведущего символа и длины от 1 до 9. Следовательно, допустимые модификаторы — от 01 до 09. Модификатор 01 приравнивается к отсутствию модификатора. Использование модификатора поддерживается не всеми лексемами, для каждой из них поддержка указана индивидуально.

В случае, если модификатор применен к неподдерживающей его лексеме, он игнорируется и лексема будет выведена без учета модификатора. В случае, если количество ведущих нулей, заданное модификатором, больше поддерживаемого лексемой, фактическое число задается верхним пределом, поддерживаемым лексемой.

Для добавления в форматную строку самого символа "$" используется его двойной вариант "$$".

При обработке лексемы анализатор идет последовательно от префикса вправо. Если на пути встречается неподдерживаемый символ, данная "ложная" лексема игнорируется и не попадает в строку результата, начиная с префикса и заканчивая самим неподдерживаемым символом.

Таким образом:

- $b — просто пропадет из результата;
- $\0 (символ 0x00) — просто пропадет из результата;
- $ (пробел) — просто пропадет из результата;
- $$b — в форматную строку добавятся символы $b;
- $$v — в форматную строку добавятся символы $v;
- abc$ — в форматную строку добавятся символы abc.

Ниже представлен список доступных лексем:

- $a — IP-адрес коммутатора (Relay), подставляется в форматную строку в виде ASCII (192.168.128.21), поддерживается модификатор от *01* до *03* ($01a..$03a). Например, адрес 192.168.10.1 и $03a дадут результат

"192.168.010.001";

- $A — IP-адрес коммутатора (Relay), подставляется в форматную строку в виде HEX (C0A88015), модификаторы не поддерживаются, размер всегда равен 4 байтам;
- $r — MAC-адрес коммутатора (Relay), подставляется в форматную строку в виде ASCII (00:13:AA:11:22:33), модификаторы не поддерживаются, разрядность элемента всегда дополняется ведущими нулями до 2 символов, размер всегда равен 17 символам;
- $R — MAC-адрес коммутатора (Relay), подставляется в форматную строку в виде HEX (0013AA112233), модификаторы не поддерживаются, размер всегда равен 6 байтам;
- $c — MAC-адрес клиента, подставляется в форматную строку в виде ASCII (00:13:AA:33:22:11), модификаторы не поддерживаются, разрядность элемента всегда дополняется ведущими нулями до 2 символов, размер всегда равен 17 символам;
- $C — MAC-адрес клиента, подставляется в форматную строку в виде HEX (0013AA332211), модификаторы не поддерживаются, размер всегда равен 6 байтам;
- $h — hostname коммутатора, подставляется в форматную строку в виде ASCII, модификаторы не поддерживаются;
- $v — VLAN ID, подставляется в форматную строку в виде ASCII, по умолчанию ведущие нули не используются;
- $V — VLAN ID, подставляется в форматную строку в виде HEX, модификаторы не поддерживаются, размер — всегда 2 байта (0000..0FFF);
- $i — <unit>/<slot>/<port>, подставляется в форматную строку в виде ASCII, модификаторы не поддерживаются, ведущие нули не используются;
- $u — unit id коммутатора, всегда равен 0, подставляется в форматную строку в виде ASCII, по умолчанию ведущие нули не используются;
- $U — unit id коммутатора, всегда равен 0x00, подставляется в форматную строку в виде HEX, модификаторы не поддерживаются, размер — всегда 1 байт;
- $s — slot id, подставляется в форматную строку в виде ASCII, по умолчанию ведущие нули не используются;
- $S — slot id, подставляется в форматную строку в виде HEX, модификаторы не поддерживаются, размер — всегда 1 байт;
- $p — номер интерфейса, подставляется в форматную строку в виде

ASCII, по умолчанию ведущие нули не используются;
- $P — номер интерфейса, подставляется в форматную строку в виде HEX, модификаторы не поддерживаются, размер — всегда 1 байт.

Подстановка лексемы в виде ASCII предполагает, что в пакет попадают шестнадцатеричные значения каждого символа из таблицы ASCII. Подстановка лексемы в виде HEX предполагает, что в пакет попадает значение как есть, в бинарном виде.

Например, VLAN 123 может быть представлен в двух видах. Обычный текстовый вид: $v — ASCII (значения символов "1", "2" и "3"), в форматную строку будет записано значение 0x313233, размер 3 байта. Специальный HEX-вид: $V — HEX (123 = 0x7B), в форматную строку будет записано значение 0x007B, размер 2 байта.

Допускается использовать обе формы лексем совместно в одной форматной строке. Допускается комбинировать лексемы в форматной строке произвольным образом, в том числе использовать обычные ASCII-символы вместе с лексемами в произвольном виде.

Обратите внимание, что длина форматной строки как правило меньше результата преобразования. Короткая лексема $i (2 символа) на выходе может быть преобразована в строку 0/0/12 (6 символов). Если достижение максимальной длины форматной строки происходит на участке между лексемами (среди обычных символов), дальнейшая обработка форматной строки прекращается. В любом случае, в пакет попадает только успешно обработанная часть форматной строки.

Максимальная длина форматной строки равна 64 символам.

ГЛАВА 17. DHCP SNOOPING

17.1. Введение в DHCP

DHCP (англ. Dynamic Host Configuration Protocol) — протокол, позволяющий хостам сети автоматически получать сетевые настройки, необходимые для работы в сети TCP/IP. Стандарт DHCP описан в RFC 2131.

Сообщения протокола DHCP

Обмен сообщениями между клиентом и сервером DHCP обычно инициируется клиентом. После получения сетевых настроек клиент и сервер продолжают обмениваться сообщениями, обновляя сетевые настройки, так как адрес по протоколу DHCP выдается сервером на определенное время.

- DHCPDISCOVER — отправляется клиентом для обнаружения всех доступных DHCP-серверов в сети;
- DHCPOFFER — отправляется сервером в ответ на DHCPDISCOVER, предлагая принять сетевые настройки от данного DHCP-сервера;
- DHCPREQUEST — отправляется клиентом определенному DHCP-серверу для запроса сетевых настроек;
- DHCPACK — отправляется сервером в ответ на DHCPREQUEST клиента, является подтверждением переданных сетевых настроек;
- DHCPDECLINE — отправляется в случае, если указанный сетевой адрес уже используется. После этого сообщения процесс запроса адреса должен быть повторен клиентом сначала;
- DHCPNAK — отправляется сервером в случае, если назначение переданных сетевых настроек невозможно. После этого сообщения процесс получения настроек должен быть повторен;
- DHCPRELEASE — отправляется клиентом в случае, если он освобождает занимаемый адрес. Процесс получения может быть повторен;
- DHCPINFORM — отправляется клиентом для получения от сервера дополнительных настроек сети.

Процесс получения сетевых настроек по протоколу DHCP:

Рисунок 56. Типовой процесс получения сетевых настроек по протоколу DHCP

DHCP Snooping

DHCP Snooping — служба коммутатора, предназначенная для защиты от атак с использованием протокола DHCP. DHCP Snooping позволяет защитить клиента от получения адреса от ненадежного DHCP-сервера. Для корректной работы DHCP Snooping интерфейсы коммутатора делятся на две группы — доверенные (trust) и недоверенные (untrust). При настройке коммутатора доверенными интерфейсами принято считать интерфейсы, за которыми находится доверенный DHCP-сервер. Остальные интерфейсы считаются недоверенными.

Пример работы DHCP Snooping изображен на схеме ниже:

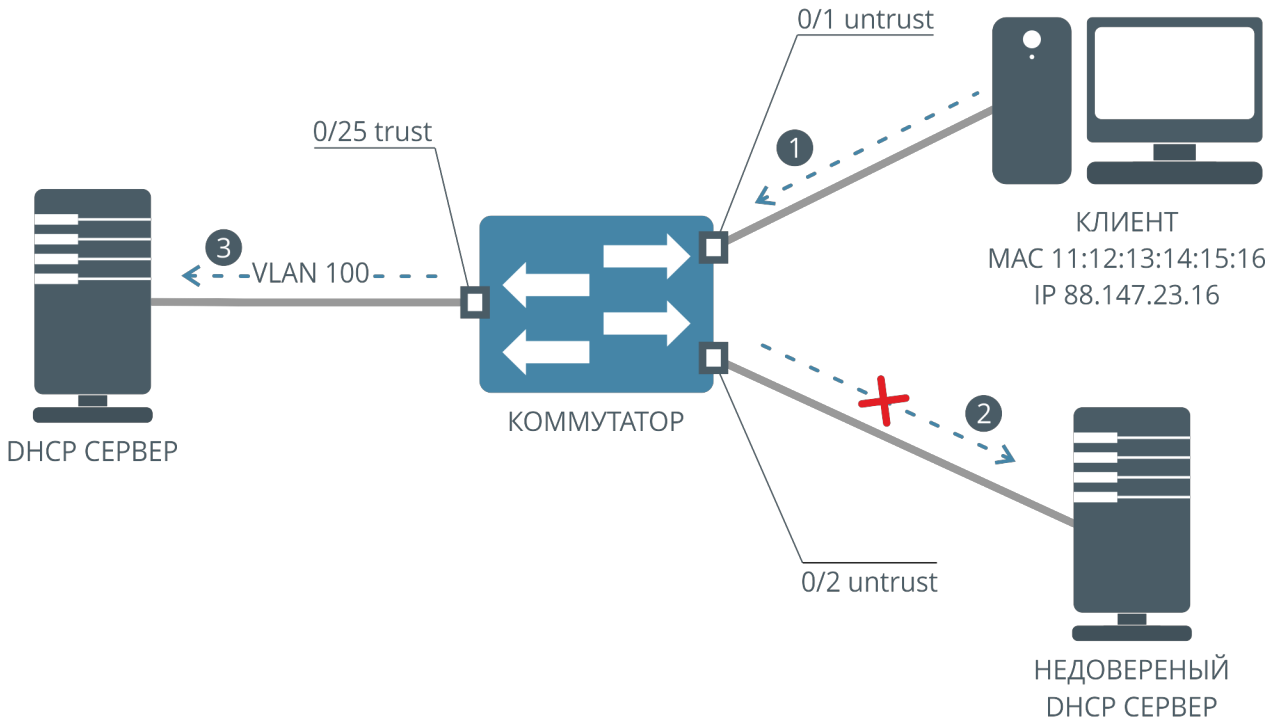

Рисунок 57. Схема работы DHCP Snooping

1. Клиент отправляет запрос DHCPDISCOVER для обнаружения доступных DHCP-серверов.
2. Отправленный клиентом запрос DHCPDISCOVER является широковещательным, но благодаря DHCP Snooping он не попадает на недоверенный DHCP-сервер за интерфейсом 0/2, а перенаправляется на доверенный интерфейс коммутатора 0/25.
3. Клиентский запрос DHCPDISCOVER доходит до доверенного DHCP-сервера. Сервер отвечает на него, после этого клиент получает сетевые настройки.

DHCP L2 Relay

DHCP L2 Relay — это пересылка DHCP-пакетов с добавлением информации о клиенте, запросившем сетевые настройки, и о самом коммутаторе, принявшего DHCP-пакеты, в виде специальной опции протокола DHCP, Option 82. Получив DHCP-запрос с опцией 82, DHCP-сервер сможет принять решение о выдаче сетевых настроек клиенту на основании дополнительной информации из полей опции 82 DHCP-пакета. DHCP L2 Relay может работать как независимо от DHCP Snooping, так и совместно. Стандарт описан в RFC 3046.

DHCP опция 82 состоит из двух подопций:

- Circuit ID — произвольная строка, содержит описание клиента, запросившего адрес;
- Remote ID — произвольная строка, описывающая DHCP-ретранслятор, принявший запрос клиента.

DHCP IP Source Guard

IP Source Guard — служба коммутатора, с помощью которой ограничивается IP-трафик на интерфейсах. Трафик фильтруется на основании таблицы клиентов DHCP Snooping и таблице статических привязок. Таблица привязок обычно устанавливает соответствие между IP-адресом, номером интерфейса, MAC-адресом и VLAN ID. IP Source Guard позволяет предотвратить ситуацию, когда злоумышленник выдает себя за клиента при помощи подмены IP-адреса, тем самым перехватывая трафик, предназначенный клиенту.

17.2. Настройка DHCP Snooping на коммутаторах АЛСиТЕК

Служба DHCP Snooping настраивается в несколько шагов. Прежде всего необходимо включить глобально службу, затем назначить службе номера VLAN, в которых необходимо следить за DHCP-трафиком. После этого необходимо указать доверенный интерфейс, за которым находится DHCP-сервер. Доверенных интерфейсов может быть несколько. Все остальные интерфейсы будут считаться недоверенными (untrust), дополнительная настройка недоверенных интерфейсов на коммутаторе не требуется.

Через недоверенный интерфейс коммутатора могут проходить только клиентские DHCP-пакеты. Эти пакеты будут направлены только на доверенные интерфейсы. После ответа сервера на клиентский запрос адреса клиент получит по DHCP все необходимые сетевые настройки.

Пошаговая настройка DHCP Snooping

Шаг 1. Предварительные настройки коммутатора

Для корректной работы DHCP-snooping на коммутаторе АЛСиТЕК согласно приложенной схеме требуется произвести следующие предварительные настройки.

Создание VLAN 100:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 100
(als_sw) (Vlan) #exit
```

Настройка VLAN 100 на клиентских интерфейсах:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1-0/2
(als_sw) (configure) (interface 0/1-0/2) #vlan participation include 100
(als_sw) (configure) (interface 0/1-0/2) #vlan pvid 100
```

Настройка VLAN 100 на сервеном интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/25
(als_sw) (configure) (interface 0/25) #vlan participation include 100
(als_sw) (configure) (interface 0/25) #vlan tagging 100
```

Шаг 2. Включение службы DHCP Snooping на устройстве

Для глобального включения службы необходимо выполнить команду:

```
(als_sw) #configure
(als_sw) (configure) #ip dhcp snooping
```

Шаг 3. Задание VLAN, на которых включен DHCP Snooping

Далее, указываем один или несколько VLAN, на которых должна быть включена служба DHCP Snooping:

```
(als_sw) #configure
(als_sw) (configure) #ip dhcp snooping vlan 100
```

В примере DHCP Snooping включается на VLAN 100. Может быть указано несколько VLAN.

Шаг 4. Назначение доверенных интерфейсов для DHCP Snooping

Для указания доверенного интерфейса нужно выполнить команду:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/25
(als_sw) (configure) (interface 0/25) #ip dhcp snooping trust
```

В примере интерфейс 0/25 переводится в состояние доверенного для DHCP Snooping. Доверенных интерфейсов может быть несколько.

Просмотр клиентов

После успешного получения адреса клиентами можно просмотреть таблицу клиентов следующей командой:

```
(als_sw) #show ip dhcp snooping binding

  MAC Address        IP Address       VLAN  Interface  Type     Lease (Secs)
  -----------------  ---------------  ----  ---------  -------  -----------
  11:12:13:14:15:16  88.147.23.16     100   0/1        DYNAMIC  285
```

На примере клиент за интерфейсом 0/1 получил адрес 88.147.23.16 по DHCP в 100 VLAN. Адрес выдается сервером на определенное время. По истечении этого времени (285 секунд в примере выше) клиент должен будет обновить время аренды адреса согласно протоколу DHCP. Если клиент не обновит адрес и время аренды истечет, запись будет удалена из таблицы коммутатора.

После описанных выше настроек конфигурация коммутатора будет следующей:

```
(als_sw) #show running-config
vlan database
vlan 100
exit
configure
ip dhcp snooping
ip dhcp snooping vlan 100
interface 0/1-0/2
vlan participation include 100
vlan pvid 100
exit
interface 0/25
vlan participation include 100
vlan tagging 100
ip dhcp snooping trust
exit
exit
```

17.3. Настройка DHCP L2 Relay на коммутаторах АЛСиТЕК

Служба DHCP L2 Relay настраивается в несколько шагов. Прежде всего необходимо включить глобально службу, затем назначить службе номера VLAN, в которых необходимо следить за DHCP-трафиком с опцией 82. После этого необходимо указать доверенный (l2relay trust) интерфейс, за которым находится DHCP-сервер. Доверенных интерфейсов может быть несколько. Также необходимо указать недоверенные (l2relay untrust) интерфейсы, обычно это интерфейсы, ведущие к абонентскому оборудованию.

Пошаговая настройка DHCP L2 Relay

Рассмотрим настройку на примере простой схемы:

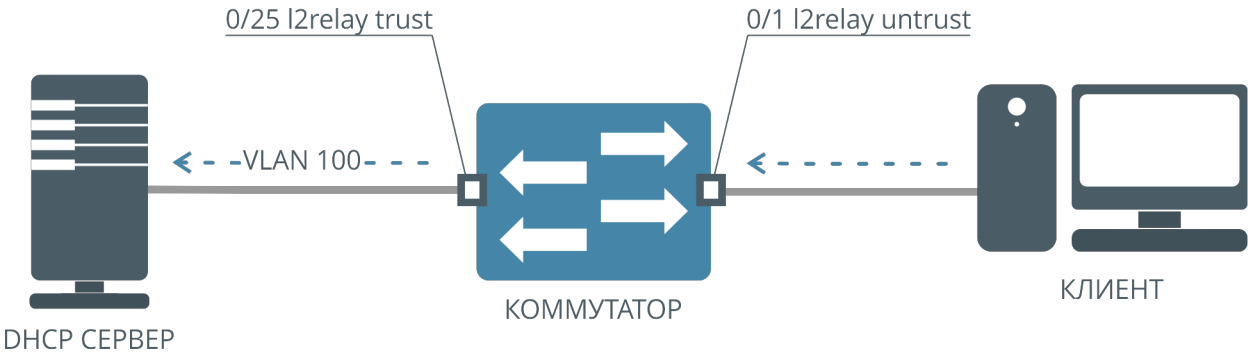

Рисунок 58. Схема работы DHCP L2 Relay

Шаг 1. Предварительные настройки коммутатора

Создание VLAN 100:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 100
(als_sw) (Vlan) #exit
```

Настройка VLAN 100 на клиентском интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan participation include 100
(als_sw) (configure) (interface 0/1) #vlan pvid 100
```

Настройка VLAN 100 на сервеном интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/25
(als_sw) (configure) (interface 0/25) #vlan participation include 100
(als_sw) (configure) (interface 0/25) #vlan tagging 100
```

Шаг 2. Включение службы DHCP L2 Relay на устройстве

Для глобального включения службы DHCP L2 Relay нужно выполнить команду:

```
(als_sw) #configure
(als_sw) (configure) #dhcp l2relay
```

Шаг 3. Задание VLAN, на которых включен DHCP L2 Relay

Для указания VLAN, на которых включен DHCP L2 Relay, нужно выполнить команду:

```
(als_sw) #configure
(als_sw) (configure) #dhcp l2relay vlan 100
```

Шаг 4. Настройка форматных строк для полей DHCP Option 82

Настройка подопции circuit-id:

```
(als_sw) #configure
(als_sw) (configure) #dhcp l2relay circuit-id frmtstr enable
(als_sw) (configure) #dhcp l2relay circuit-id frmtstr "SomeStringWithLexems"
(als_sw) (configure) #dhcp l2relay circuit-id vlan 100
```

Настройка подопции remote-id:

```
(als_sw) #configure
(als_sw) (configure) #dhcp l2relay remote-id "SW1" vlan 100
```

Шаг 5. Назначение доверенных (trust) и недоверенных (untrust) интерфейсов для DHCP L2 Relay

Настройка недоверенного (untrust) интерфейса

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #dhcp l2relay
```

Настройка доверенного (trust) интерфейса

```
(als_sw) #configure
(als_sw) (configure) #interface 0/25
(als_sw) (configure) (interface 0/25) #dhcp l2relay trust
```

После описанных выше настроек, конфигурация коммутатора будет следующей:

```
(als_sw) #show running-config
vlan database
vlan 100
exit
configure
dhcp l2relay
dhcp l2relay vlan 100
dhcp l2relay circuit-id frmtstr enable
dhcp l2relay circuit-id frmtstr "SomeStringWithLexems"
dhcp l2relay circuit-id vlan 100
dhcp l2relay remote-id "SW1" vlan 100
interface 0/1
vlan participation include 100
vlan pvid 100
dhcp l2relay
exit
interface 0/25
vlan participation include 100
vlan tagging 100
dhcp l2relay trust
exit
exit
```

Лексемы

Для возможности задания произвольных строк подопций используется механизм лексем. Лексема — это специальный набор символов, которые при вставке форматной строки в пакет будут заменены на некоторое реальное значение, специфичное для данного пакета или коммутатора. С помощью лексем в пакет можно поместить некоторую информацию для сервера, на основании которой он сможет принять решение по обработке данного пакета.

В форматных строках допускается использовать печатные символы ASCII (0x20-0x7e), за исключением символа двойных кавычек (") и обратного слеша (\):

```
 !#$%&#38;'()*+,-./
0123456789:;<=>?
@ABCDEFGHIJKLMNO
PQRSTUVWXYZ[]^_
```

```
`abcdefghijklmno
pqrstuvwxyz{|}~
```

Символу "$" при этом отводится роль служебного, он используется для обозначения лексем. Лексемы начинаются со служебного символа "$", далее может идти модификатор длины (необязательный параметр), после чего идет символ (буква), обозначающая лексему. Список лексем приведен ниже. Модификатор используется для задания разрядности значения. При этом недостающие разряды дополняются ведущими нулями.

Допустимо использование только 0 в качестве ведущего символа и длины от 1 до 9. Следовательно, допустимые модификаторы — от 01 до 09. Модификатор 01 приравнивается к отсутствию модификатора. Использование модификатора поддерживается не всеми лексемами, для каждой из них поддержка указана индивидуально.

В случае, если модификатор применен к неподдерживающей его лексеме, он игнорируется и лексема будет выведена без учета модификатора. В случае, если количество ведущих нулей, заданное модификатором, больше поддерживаемого лексемой, фактическое число задается верхним пределом, поддерживаемым лексемой.

Для добавления в форматную строку самого символа "$" используется его двойной вариант "$$".

При обработке лексемы анализатор идет последовательно от префикса вправо. Если на пути встречается неподдерживаемый символ, данная "ложная" лексема игнорируется и не попадает в строку результата, начиная с префикса и заканчивая самим неподдерживаемым символом.

Таким образом:

- $b — просто пропадет из результата;
- $\0 (символ 0x00) — просто пропадет из результата;
- $ (пробел) — просто пропадет из результата;
- $$b — в форматную строку добавятся символы $b;
- $$v — в форматную строку добавятся символы $v;
- abc$ — в форматную строку добавятся символы abc.

Ниже представлен список доступных лексем:

- $a — IP-адрес коммутатора (Relay), подставляется в форматную строку в виде ASCII (192.168.128.21), поддерживается модификатор от *01* до *03* ($01a..$03a). Например, адрес 192.168.10.1 и $03a дадут результат "192.168.010.001";
- $A — IP-адрес коммутатора (Relay), подставляется в форматную строку в виде HEX (C0A88015), модификаторы не поддерживаются, размер всегда равен 4 байтам;
- $r — MAC-адрес коммутатора (Relay), подставляется в форматную строку в виде ASCII (00:13:AA:11:22:33), модификаторы не поддерживаются, разрядность элемента всегда дополняется ведущими нулями до 2 символов, размер всегда равен 17 символам;
- $R — MAC-адрес коммутатора (Relay), подставляется в форматную строку в виде HEX (0013AA112233), модификаторы не поддерживаются, размер всегда равен 6 байтам;
- $c — MAC-адрес клиента, подставляется в форматную строку в виде ASCII (00:13:AA:33:22:11), модификаторы не поддерживаются, разрядность элемента всегда дополняется ведущими нулями до 2 символов, размер всегда равен 17 символам;
- $C — MAC-адрес клиента, подставляется в форматную строку в виде HEX (0013AA332211), модификаторы не поддерживаются, размер всегда равен 6 байтам;
- $h — hostname коммутатора, подставляется в форматную строку в виде ASCII, модификаторы не поддерживаются;
- $v — VLAN ID, подставляется в форматную строку в виде ASCII, по умолчанию ведущие нули не используются;
- $V — VLAN ID, подставляется в форматную строку в виде HEX, модификаторы не поддерживаются, размер — всегда 2 байта (0000..0FFF);
- $i — <unit>/<slot>/<port>, подставляется в форматную строку в виде ASCII, модификаторы не поддерживаются, ведущие нули не используются;
- $u — unit id коммутатора, всегда равен 0, подставляется в форматную строку в виде ASCII, по умолчанию ведущие нули не используются;
- $U — unit id коммутатора, всегда равен 0x00, подставляется в форматную строку в виде HEX, модификаторы не поддерживаются, размер — всегда 1 байт;
- $s — slot id, подставляется в форматную строку в виде ASCII, по умолчанию ведущие нули не используются;
- $S — slot id, подставляется в форматную строку в виде HEX, модификаторы не поддерживаются, размер — всегда 1 байт;
- $p — номер интерфейса, подставляется в форматную строку в виде

ASCII, по умолчанию ведущие нули не используются;
- $P — номер интерфейса, подставляется в форматную строку в виде HEX, модификаторы не поддерживаются, размер — всегда 1 байт.

Подстановка лексемы в виде ASCII предполагает, что в пакет попадают шестнадцатеричные значения каждого символа из таблицы ASCII. Подстановка лексемы в виде HEX предполагает, что в пакет попадает значение как есть, в бинарном виде.

Например, VLAN 123 может быть представлен в двух видах. Обычный текстовый вид: $v — ASCII (значения символов "1", "2" и "3"), в форматную строку будет записано значение 0x313233, размер 3 байта. Специальный HEX-вид: $V — HEX (123 = 0x7B), в форматную строку будет записано значение 0x007B, размер 2 байта.

Допускается использовать обе формы лексем совместно в одной форматной строке. Допускается комбинировать лексемы в форматной строке произвольным образом, в том числе использовать обычные ASCII-символы вместе с лексемами в произвольном виде.

Обратите внимание, что длина форматной строки как правило меньше результата преобразования. Короткая лексема $i (2 символа) на выходе может быть преобразована в строку 0/0/12 (6 символов). Если достижение максимальной длины форматной строки происходит на участке между лексемами (среди обычных символов), дальнейшая обработка форматной строки прекращается. В любом случае, в пакет попадает только успешно обработанная часть форматной строки.

Максимальная длина форматной строки равна 64 символам.

Просмотр счетчиков

В ходе работы служба DHCP L2 Relay ведет подсчет обработанных пакетов. Счетчики пакетов можно просмотреть командой:

```
(als_sw) #show dhcp l2relay stats trust

 Interface   Server Packets  Server Packets     Client Packets  Client Packets
             With Option 82  Without Option 82  With Option 82  Without Option
82
-----------  --------------  -----------------  --------------  ---------------
--
 0/25          2               0                  0               0
```

17.4. Настройка DHCP IP Source Guard на коммутаторах АЛСиТЕК

Служба DHCP IP Source Guard настраивается в несколько шагов. Прежде всего необходимо включить глобально службу DHCP Snooping, затем назначить службе DHCP Snooping номера VLAN, в которых необходимо следить за DHCP трафиком. После этого необходимо включить IP Source Guard на клиентских интерфейсах. Опционально можно задать статическую привязку клиента по MAC, IP и VLAN к конкретному интерфейсу.

На коммутаторах АЛСиТЕК есть два режима работы IP Source Guard: режим проверки соответствия только IP-адреса, выданного доверенным DHCP-сервером клиенту, и режим проверки соответствия IP-адреса и MAC-адреса (одновременно) клиента.

Пошаговая настройка DHCP IP Source Guard

Рассмотрим настройку на примере простой схемы:

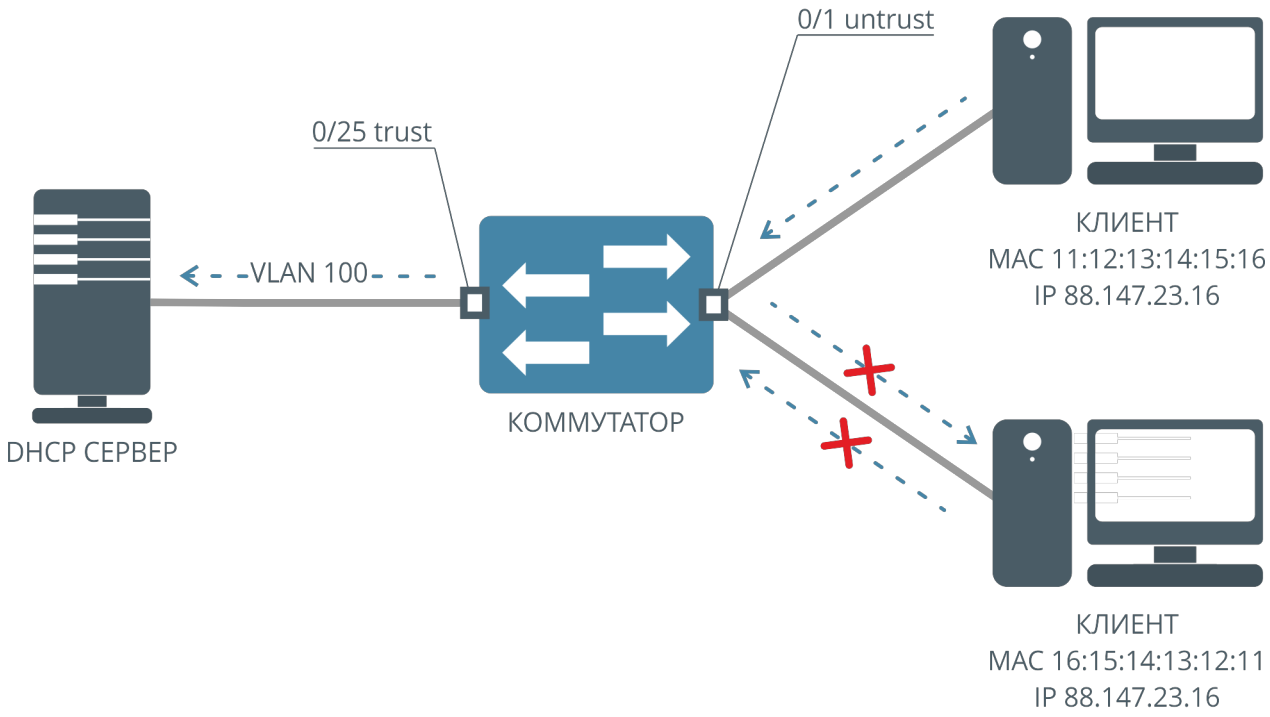

Рисунок 59. Схема работы DHCP IP Source Guard

Шаг 1. Предварительные настройки коммутатора

Создание VLAN 100:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 100
(als_sw) (Vlan) #exit
```

Настройка VLAN 100 на клиентском интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #vlan participation include 100
(als_sw) (configure) (interface 0/1) #vlan pvid 100
```

Настройка VLAN 100 на серверном интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/25
(als_sw) (configure) (interface 0/25) #vlan participation include 100
(als_sw) (configure) (interface 0/25) #vlan tagging 100
```

Шаг 2. Включение службы DHCP Snooping на устройстве

Для глобального включения службы необходимо выполнить команду:

```
(als_sw) #configure
(als_sw) (configure) #ip dhcp snooping
```

Шаг 3. Задание VLAN, на которых включен DHCP Snooping

Далее, указываем один или несколько VLAN, на которых должна быть включена служба DHCP Snooping:

```
(als_sw) #configure
(als_sw) (configure) #ip dhcp snooping vlan 100
```

В примере DHCP Snooping включается на VLAN 100. Может быть указано несколько VLAN.

Шаг 4. Включение IP Source Guard на клиентских интерфейсах

Включение проверки IP-адреса на интерфейсе 0/1:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #ip verify source
```

Включить проверку IP-адреса и MAC-адреса на интерфейсе 0/1 (опционально):

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #ip verify source port-security
```

Шаг 5. Назначение доверенного интерфейса

В примере ниже назначается доверенным интерфейс 0/25:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/25
(als_sw) (configure) (interface 0/25) #ip dhcp snooping trust
```

Шаг 6. Статическое назначение адресов (опционально)

Для статической привязки к определенному интерфейсу MAC и IP-адреса в определенном VLAN используется команда:

```
(als_sw) #configure
(als_sw) (configure) #ip dhcp snooping binding 11:12:13:14:15:16 vlan 100 88.14
7.23.16 interface 0/1
```

В примере для интерфейса 0/1 разрешен трафик с MAC-адреса 11:12:13:14:15:16 во VLAN 100, с IP-адресом источника 88.147.23.16. Весь остальной трафик на этом интерфейсе будет блокирован, в том числе и DHCP.

После описанных выше настроек конфигурация коммутатора будет следующей:

```
(als_sw) #show running-config
vlan database
vlan 100
exit
configure
ip dhcp snooping
ip dhcp snooping vlan 100
ip dhcp snooping binding 11:12:13:14:15:16 vlan 100 88.147.23.16 interface 0/1
interface 0/1
vlan participation include 100
vlan pvid 100
ip verify source port-security
exit
interface 0/25
vlan participation include 100
vlan tagging 100
ip dhcp snooping trust
exit
exit
```

ГЛАВА 18. DYNAMIC ARP INSPECTION (DAI)

Dynamic ARP Inspection (также используется название Dynamic ARP Protection) — функция коммутатора, предназначенная для предотвращения атак с использованием протокола ARP. Примером таких атак является ARP Spoofing, в результате которой может быть перехвачен трафик между узлами в пределах одного широковещательного домена. Функция работает только с пакетами протокола ARP и не влияет напрямую на трафик пользователей и другие протоколы.

18.1. Введение в Dynamic ARP Inspection

Данная функция коммутатора условно делит интерфейсы коммутатора на две группы:

- доверенные (trust). Пакеты ARP, приходящие на эти интерфейсы, не обрабатываются и пересылаются как обычно;
- недоверенные (untrust). Каждый пакет ARP, приходящий на эти интерфейсы, проверяется по группе условий, о которых подробно сказано ниже.

Проверки, которые производит коммутатор, можно разделить на три группы:

- соответствие общим правилам для ARP-пакета. В этом пункте проверяется соответствие заголовка ARP-пакета и его содержимого во избежание передачи заведомо неправильных пакетов в сеть;
- разрешение согласно статическим спискам доступа, перечисленным в конфигурации коммутатора;
- разрешение согласно динамической информации о клиентах, получивших сетевые настройки по протоколу DHCP. В этом пункте также проверяется соответствие статическим привязкам DHCP, если они есть.

18.2. Dynamic ARP Inspection на коммутаторах АЛСиТЕК

Настройка DAI производится в несколько этапов. Прежде всего нужно глобально включить службу Dynamic ARP Inspection на определенном VLAN. VLAN, в которых будут перехватываться ARP-пакеты, может быть несколько. Далее необходимо указать какие интерфейсы коммутатора нужно считать доверенными (trust), а какие — недоверенными (untrust).

Также можно дополнительно указать особые опции валидации перехваченных в указанных VLAN пакетов ARP. Валидацию пакетов можно проводить по трем пунктам:

- Source MAC — проверяется совпадение поля Source MAC в заголовке Ethernet и значение поля Sender MAC в теле ARP-пакета в **запросах и ответах** ARP;
- Destination MAC — проверяется совпадение поля Destination MAC в заголовке Ethernet и значение поля Target MAC в теле ARP-пакета в **ответах** ARP;
- IP — проверяется поле Sender IP в ARP-пакетах **запросов и ответов** и поле Target IP в **ответах**. Адреса в этих полях не могут быть равны 0.0.0.0, 255.255.255.255, не могут быть мультикаст-адресами, адресами из подсети 240.0.0.0/4 и адресами из подсети 127.0.0.0/8.

По умолчанию все типы валидации отключены. Их можно включать в любых комбинациях, независимо друг от друга, глобально для всего коммутатора.

Настройка Dynamic ARP Inspection

В ходе настройки мы будем использовать типовую схему, изображенную на рисунке ниже:

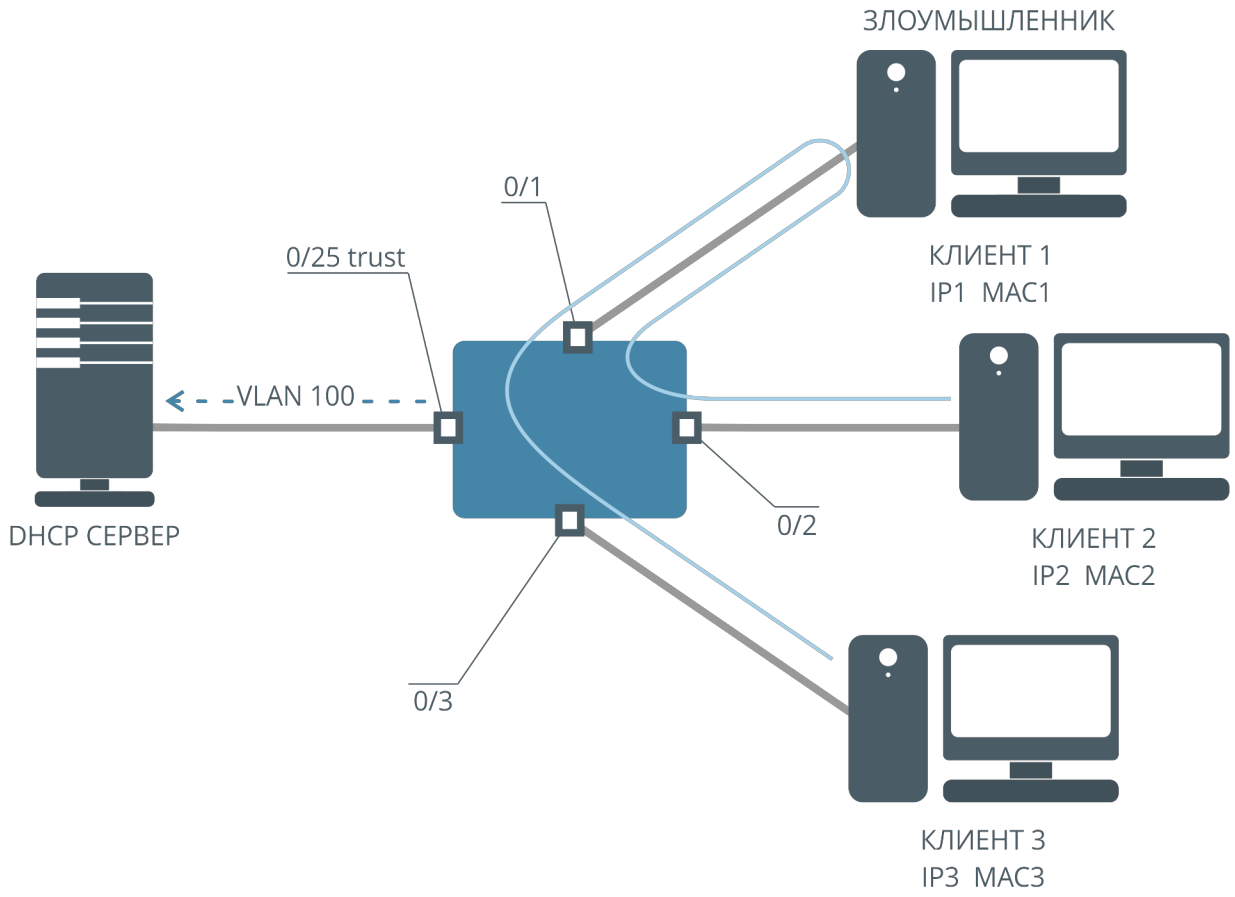

Рисунок 60. Упрощенная схема атаки с подменой MAC-адресов

На рисунке изображен коммутатор, к которому через интерфейсы 0/1, 0/2 и 0/3 подключены клиенты. Эти клиенты получают сетевые настройки от DHCP-сервера, подключенного к интерфейсу 0/25.

В обычной ситуации может быть проведена атака с подменой пакетов ARP следующим образом:

- клиент 3 высылает broadcast ARP-запрос на поиск IP2. Поскольку запрос широковещательный, его увидят все клиенты, а именно клиенты 1 и 2;
- клиент 2 на запрос ARP отвечает валидным ARP-ответом, поскольку IP 2 принадлежит ему. В ответе указывается MAC 2. Ответ направлен на MAC 3;
- злоумышленник на клиенте 1 отвечает на запрос ARP невалидным ARP-ответом, в котором указан IP 2 и MAC 1. Ответ направляется на IP 3, MAC 3;
- может сложиться ситуация, при которой ответ ARP от злоумышленника будет получен ранее на клиенте 3, и он будет считать, что IP 2 имеет хост с MAC-адресом MAC 1. В этом случае трафик, предназначенный клиенту 2, будет на самом деле направлен клиенту 1;
- далее злоумышленник на клиенте 1 аналогичным образом может подменить ARP-ответ для клиента 2, который будет искать MAC-адрес клиента 3.

Таким образом, после успешной подмены ARP-ответов, трафик между клиентами 2 и 3 будет проходить через клиента 1, при этом узнать об этом клиенты 2 и 3 не смогут. Служба Dynamic ARP Inspection позволяет предотвратить подобные атаки, заблокировав ARP-пакеты, которые содержат в своем теле IP-адрес или MAC-адрес, которые не могут быть приняты с данного интерфейса.

Шаг 1. Предварительная настройка

Для корректной работы Dynamic ARP Inspection на коммутаторе АЛСиТЕК согласно приложенной схеме требуется произвести следующие предварительные настройки:

Создание VLAN 100:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 100
(als_sw) (Vlan) #exit
```

Настройка VLAN 100 на абонентских интерфейсах:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1-0/3
(als_sw) (configure) (interface 0/1-0/3) #vlan participation include 100
(als_sw) (configure) (interface 0/1-0/3) #vlan pvid 100
(als_sw) (configure) (interface 0/1-0/3) #exit
(als_sw) (configure) #exit
```

Настройка VLAN 100 на uplink:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/25
(als_sw) (configure) (interface 0/25) #vlan participation include 100
(als_sw) (configure) (interface 0/25) #vlan tagging 100
(als_sw) (configure) (interface 0/25) #exit
(als_sw) (configure) #exit
```

Шаг 2. Включение Dynamic ARP Inspection на VLAN

Служба включается на определенном VLAN, которых может быть несколько. Все ARP-пакеты, входящие на коммутатор в этих VLAN, будут перехвачены и проверены перед отправкой. Для включения используется команда:

```
(als_sw) #configure
(als_sw) (configure) #ip arp inspection vlan 100
(als_sw) (configure) #exit
```

После выполнения этой команды все ARP-пакеты, входящие на коммутатор во VLAN 100, будут перехватываться и обрабатываться службой Dynamic ARP Inspection. Поскольку ни DHCP, ни разрешающие правила ARP ACL еще не настроены, пакеты ARP будут отбрасываться. Проверка ARP-пакетов службой Dynamic ARP Inspection производится после применения входных правил (port-based VLAN в нашем примере).

Шаг 3. Назначение доверенных интерфейсов

На доверенных интерфейсах проверка ARP-пакетов не будет производиться. Как правило, это интерфейс или несколько интерфейсов, которыми коммутатор соединяется с вышестоящей сетью провайдера.

```
(als_sw) #configure
(als_sw) (configure) #interface 0/25
(als_sw) (configure) (interface 0/25) #ip arp inspection trust
(als_sw) (configure) (interface 0/25) #exit
(als_sw) (configure) #exit
```

Все остальные интерфейсы будут считаться недоверенными (untrust), дополнительная настройка для них не требуется. После выполнения этой команды ARP-пакеты, приходящие с интерфейса 0/25, не будут отбрасываться и проверяться службой Dynamic ARP Inspection, поскольку этот порт считается доверенным.

Шаг 4. Настройка DHCP Snooping на коммутаторе (опционально)

Поскольку разрешение ARP-пакетов может быть получено либо на основании списков доступа ARP, либо на основании таблицы клиентов DHCP, необходимо настроить хотя бы одну из этих служб. Если не будет настроена ни одна из служб, то весь трафик ARP на недоверенных (untrust) интерфейсах будет отброшен.

Подробнее про настройку DHCP Snooping можно прочитать в соответствующей главе. Приведем конфигурацию для нашего примера:

```
configure
ip dhcp snooping
ip dhcp snooping vlan 100
interface 0/25
ip dhcp snooping trust
exit
exit
```

Из конфигурации видно, что DHCP Snooping включен глобально, включен на VLAN 100, а также 0/25 интерфейс принят доверенным (trust) для службы DHCP Snooping. Теперь при получении адреса клиентом 1 (на схеме), полученный IP-адрес и MAC-адрес клиента будет записан в таблицу DHCP-клиентов, а Dynamic ARP Inspection разрешит прохождение пакетов протокола ARP на этом интерфейсе от данного клиента по его IP и MAC адресам.

Шаг 5. Создание списка доступа ARP (опционально)

В некоторых случаях необходимо настроить Dynamic ARP Inspection без включения DHCP Snooping, задав список разрешенных MAC и IP-адресов вручную. Для этого используются специальные списки доступа ARP.

Ниже приведен пример разрешения отправки ARP-пакетов с MAC-адресом 08:60:6e:6f:5b:6c и IP-адресом 88.147.23.16:

```
(als_sw) #configure
(als_sw) (configure) #arp access-list "LIST1"
(als_sw) (configure) (arp-access-list "LIST1") #permit ip host 88.147.23.16 mac
 host 08:60:6e:6f:5b:6c
(als_sw) (configure) (arp-access-list "LIST1") #exit
(als_sw) (configure) #exit
```

После создания список доступа не влияет на прохождение ARP-пакетов. Для того, чтобы он применился, нужно явно настроить привязку к определенному VLAN явно. Правил в списке доступа может быть несколько, допустимо создавать только разрешающие. Настраивать списки доступа ARP не обязательно, поскольку Dynamic ARP Inspection может использовать данные из таблицы DHCP-клиентов для проверки ARP-пакетов, как уже было сказано в предыдущем шаге. Тем не менее необходимо, чтобы хотя бы один источник разрешения был настроен — либо DHCP, либо списки доступа ARP. В противном случае весь трафик ARP во VLAN, для которых включена служба Dynamic ARP Inspection, будет блокирован на недоверенных (untrust) интерфейсах.

Шаг 6. Применение списка доступа ARP на VLAN (опционально)

```
(als_sw) #configure
(als_sw) (configure) #ip arp inspection filter "LIST1" vlan 100
(als_sw) (configure) #exit
```

После этой настройки все ARP-пакеты, перехваченные на недоверенных (untrust) портах во VLAN 100, будут проходить проверку соответствия правилам списка доступа с именем "LIST1". К одному VLAN можно привязать только один список доступа. Если пакет попадет под одно из правил списка доступа, он будет считаться разрешенным и проверка завершится. Если пакет не подойдет ни под одно из правил (включая DHCP), пакет будет считаться запрещенным и будет отброшен.

Есть два режима привязки списка доступа к VLAN. Первый, приведенный в примере выше, разрешает продолжение проверки ARP-пакета по таблице DHCP-клиентов. Второй режим, static, такую проверку запрещает:

```
(als_sw) #configure
(als_sw) (configure) #ip arp inspection filter "LIST1" vlan 100 static
(als_sw) (configure) #exit
```

После такой настройки все запрещенные ARP ACL пакеты будут отброшены, вне зависимости от состояния DHCP-таблицы.

Задание режима валидации ARP-пакетов

Валидация ARP-пакетов производится до проверок по спискам доступа ARP и DHCP. Режим валидации ARP-пакетов задается глобально для всей службы и состоит из трех флагов: SRC MAC, DST MAC, IP. Флаги можно устанавливать независимо друг от друга, в любой комбинации. По умолчанию валидация пакетов отключена. Если перехваченный пакет не пройдет валидацию по любому из включенных пунктов, он будет отброшен, дальнейшая проверка (по спискам доступа ARP или по таблице клиентов DHCP) не производится.

Примеры настройки всех трех пунктов:

```
(als_sw) #configure
(als_sw) (configure) #ip arp inspection validate src-mac dst-mac ip
(als_sw) (configure) #exit
```

Пример настройки проверки только SRC MAC:

```
(als_sw) #configure
(als_sw) (configure) #ip arp inspection validate src-mac
(als_sw) (configure) #exit
```

Просмотр настройки:

```
(als_sw) #show ip arp inspection

Source MAC Validation.......................... Enabled
Destination MAC Validation..................... Disabled
IP Address Validation.......................... Disabled
```

Изменение режима логирования отброшенных пакетов

Служба Dynamic ARP Inspection при обработке пакетов может выводить в системный лог коммутатора дамп отброшенных пакетов. По этим дампам можно впоследствии определить источник атаки. По умолчанию все отброшенные пакеты заносятся в системный лог коммутатора. Чтобы выключить эту функцию, выполните команду:

```
(als_sw) #configure
(als_sw) (configure) #no ip arp inspection vlan 100 logging
(als_sw) (configure) #exit
```

Сообщение службы может выглядеть следующим образом:

```
<5> arpinspection_control.cpp:384 [2000.01.01 00:14:30] MSG(18): 'ArpInspection
: Got a invalid ARP packet from 0/16 in VLAN 999, Ethernet II: DST ff:ff:ff:ff:f
f:ff SRC 08:60:6e:6f:5b:6c, packet is: ARP REQUEST from 224.17.1.100 (08:60:6e:6
f:5b:6c) to 172.17.1.1 (ff:ff:ff:ff:ff:ff)'
```

Задание порога отключения интерфейса

У службы Dynamic ARP Inspection есть дополнительная возможность, при превышении определенного количества ARP-пакетов на интерфейсе служба может заблокировать интерфейс, тем самым прервав весь поток трафика. По истечении определенного времени служба восстановит интерфейс, и он продолжит работу. Интерфейс отключается физически (shutdown).

Для включения этой функции используется команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #ip arp inspection limit rate <packets> bu
rst interval <sec>
```

Команда устанавливает порог в packets пакетов ARP, которые могут пройти за seconds секунд. Если количество пакетов за seconds секунд превысит указанное число, то интерфейс будет заблокирован. В момент блокирования интерфейса начинается отсчет времени восстановления. Когда это время истечет, интерфейс будет восстановлен.

По умолчанию ограничение на количество пакетов установлено в 15 ARP-пакетов в секунду. Его также можно отключить командой:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #ip arp inspection limit none
```

В случае отключения, при превышении количества пакетов в секунду, интерфейс не будет блокироваться.

Для управления интервалом восстановления используется команда:

```
(als_sw) #configure
(als_sw) (configure) #ip arp inspection recovery interval <seconds>
```

Время восстановления указывается в секундах, по умолчанию оно равно 300 секундам, то есть через 5 минут заблокированный интерфейс вновь продолжит свою работу.

Просмотр состояния и счетчиков

У службы Dynamic ARP Inspection есть несколько команд для получения состояния и счетчиков. Команда ниже выводит состояние VLAN, привязанных к ним списков доступа ARP, режима логирования отброшенных пакетов, а также режим привязки:

```
(als_sw) #show ip arp inspection

Source MAC Validation.......................... Disabled
Destination MAC Validation..................... Disabled
IP Address Validation.......................... Disabled

VLAN  Configuration  Log Invalid  ACL Name                    Static
----  -------------  -----------  --------------------------  --------
100   Enabled        Enabled      LIST1                       Disabled
```

Следующая команда выводит счетчики обработанных пакетов:

```
(als_sw) #show ip arp inspection statistics

  VLAN  Forwarded  Dropped
  ----  ---------  ---------
  100   4          2
```

Эта команда показывает количество пропущенных и отброшенных пакетов для всех VLAN, на которых настроена служба Dynamic ARP Inspection.

Команда ниже выводит счетчики обработанных пакетов для определенного VLAN:

```
(als_sw) #show ip arp inspection statistics vlan 999

  VLAN  DHCP    ACL     DHCP    ACL     Bad     Bad     Invalid
        Drop    Drop    Permit  Permit  SrcMAC  DstMAC  IP
  ----  ------  ------  ------  ------  ------  ------  -------
  100   1       0       3       0       0       0       0
```

Эта команда показывает количество отброшенных пакетов согласно правилам DHCP (DHCP Drop) и ARP ACL (ACL Drop), пропущенных согласно правилам DHCP (DHCP Permit) и ARP ACL (ACL Permit), а также количество отброшенных пакетов, не прошедших валидацию по каждому из типов (Bad Src MAC, Bad Dest MAC, Invalid IP).

Очистить счетчики службы Dynamic ARP Inspection можно командой:

```
(als_sw) #clear ip arp inspection statistics
```

Настройки службы и время до восстановления заблокированных интерфейсов после очистки статистики не меняются.

ГЛАВА 19. QOS (QUALITY OF SERVICE)

19.1. Введение в QoS

Термином QoS обозначают комплекс мер для повышения качества предоставления услуг в цифровых сетях. Механизм работы QoS базируется на общих соглашениях между производителями сетевого оборудования и описывается в нескольких стандартах на различных уровнях организации сети. Как правило, коммутаторы L2+ для повышения качества услуг используют информацию из тега 802.1q и IP-заголовка пакета.

В классическом варианте результатом работы механизмов QoS является полное прохождение высокоприоритетного трафика (например трафика IPTV), в том числе за счет потери низкоприоритетного трафика:

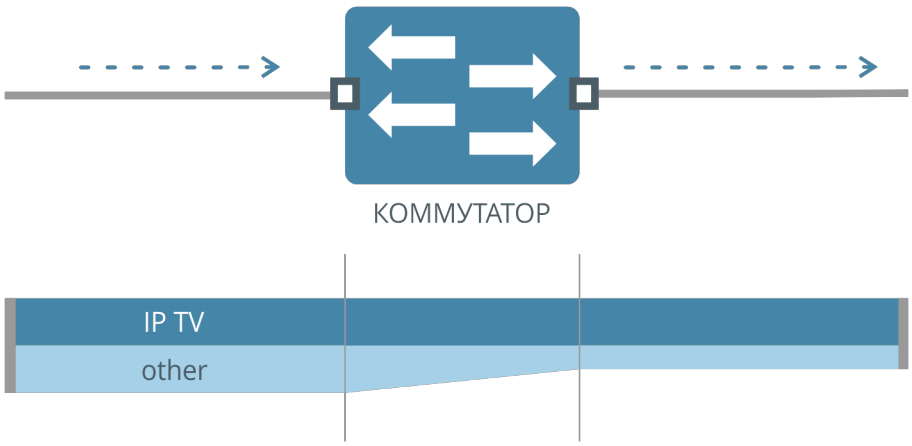

Рисунок 61. Общий принцип работы QoS

Назначение приоритета трафика по полям пакета

Назначение приоритета определенному трафику по полям пакета может быть произведено с помощью создания политик QoS. Как правило, при описании политики указывается, к какому трафика она применяется, и какие действия необходимо произвести с трафиком. Поля и действия могут различаться у разных производителей.

Доверие меткам 802.1p (CoS)

Режим доверия меткам 802.1p (поле PRI в метке 802.1q пакета) производит приоритизацию входящего трафика на основании значения метки CoS. Значение CoS в пакетах может быть выставлено на абонентском оборудовании или на оборудовании уровня доступа. Поле CoS имеет длину 3 бита и может принимать значения от 0 до 7.

Положение метки 802.1p (CoS) в пакете изображено на схеме ниже:

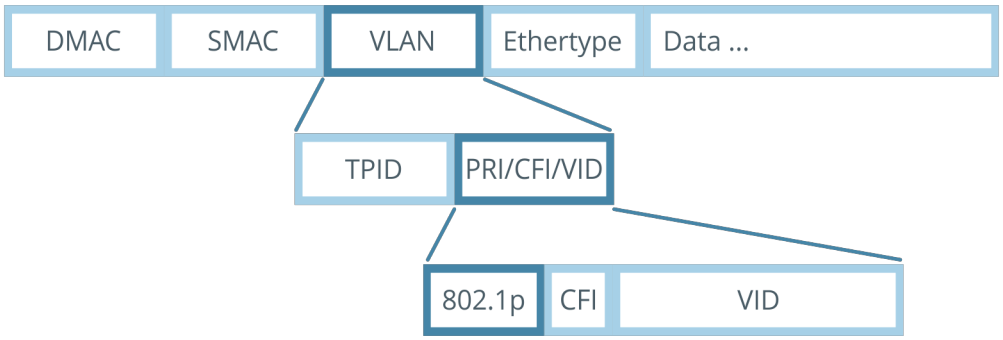

Рисунок 62. Положение метки 802.1p в теге 802.1q

Доверие меткам DSCP

Режим доверия меткам DSCP (часть поля DS в IP-заголовке, заменившего старое поле ToS в последних редакциях IP-заголовка) производит приоритизацию входящего трафика на основании значения метки DSCP. Значение поля DSCP может быть выставлено приложениями клиента, некоторым абонентскими устройствами или оборудованием уровня доступа. Поле DSCP имеет длину 6 бит и может принимать значения от 0 до 63.

Положение метки DSCP в заголовке IP-пакета изображено на схеме ниже:

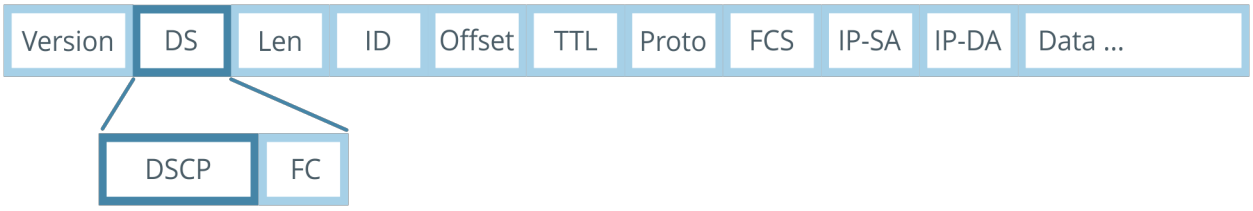

Рисунок 63. Положение метки DSCP в заголовке IP-пакета

19.2. QoS на коммутаторах АЛСиТЕК

На коммутаторах АЛСиТЕК имеется 8 выходных очередей для каждого порта.

Этапы обработки пакета в коммутаторах АЛСиТЕК можно описать следующим образом:

- Traffic Classifier. При входе пакета он классифицируется в соответствии с настроенными политиками QoS, режиму доверия меткам CoS и DSCP;
- Traffic Policer. По данным классификации пакета он обрабатывается настроенными политиками QoS и механизмами доверия меткам, и ему назначается определенный приоритет;
- В соответствии с приоритетом, пакет передается на выходной интерфейс (их может быть несколько), в соответствии с приоритетом пакету назначается определенная выходная очередь;
- Scheduler. При отправке пакеты выбираются из очередей согласно настройкам очереди и алгоритму планирования (Scheduling Algorithm). Подробнее алгоритмы планирования описаны ниже.

Графически это можно изобразить следующей схемой:

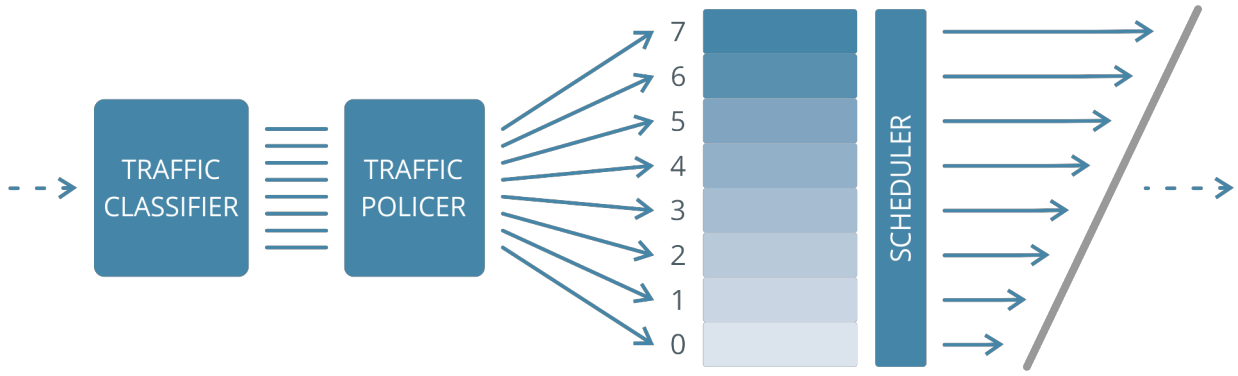

Рисунок 64. Механизм работы QoS на коммутаторах АЛСиТЕК

19.3. Очереди и алгоритмы планировщика

Выборка пакетов из очередей может производиться двумя алгоритмами планировщика: Strict Priority (SP) и Weighted Round Robin (WRR).

При использовании алгоритма SP пакеты отправляются строго по приоритету очереди. Чем больше номер очереди — тем выше приоритет, то есть сначала отправляются все пакеты из очереди 7, если они отправлены — производится отправка из очереди 6. Если все пакеты из очередей с 7 по 1 отправлены, начинается отправка из очереди 0 (наименее приоритетная очередь). Если трафик более приоритетных очередей занял всю пропускную способность интерфейса, трафик менее приоритетных очередей не отправляется и будет отброшен.

Графическое представление алгоритма SP:

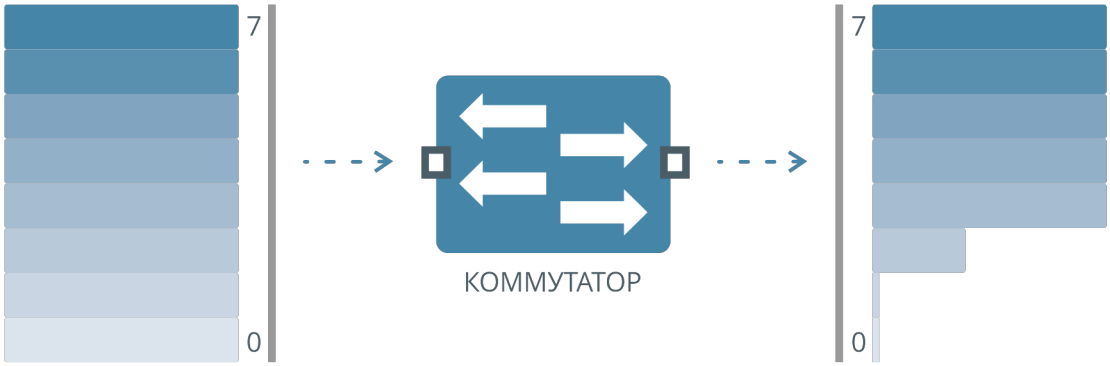

Рисунок 65. Алгоритм Strict Priority

Алгоритм WRR позволяет мягче распределить потери пакетов. Отправка пакетов осуществляется согласно назначенным весам очередей. Вес определяет, в каком соотношении будут отправляться пакеты из очередей.

Графическое представление алгоритма WRR:

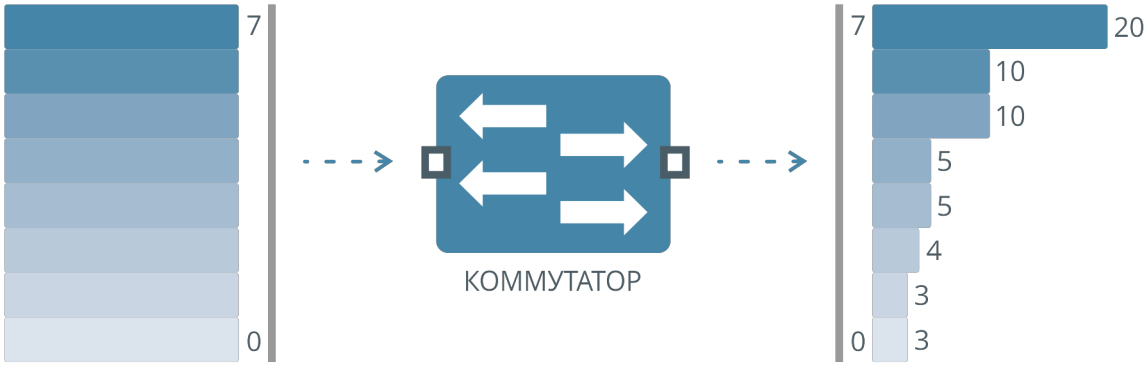

Рисунок 66. Алгоритм Weighted Round Robin

Алгоритм настраивается применительно к отдельной очереди — допустимо настроить часть очередей с алгоритмом SP и часть — с алгоритмом WRR. На схеме ниже на очередях с 7 по 4 настроен алгоритм SP, на очередях с 3 по 0 — алгоритм WRR):

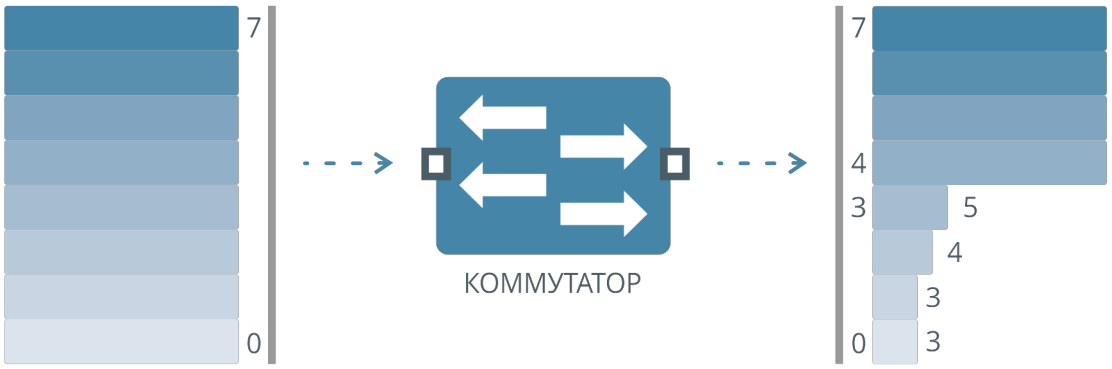

Рисунок 67. Алгоритмы Strict Priority и Weighted Round Robin одновременно

Настройка очередей и алгоритмов планировщика отправки пакетов

По умолчанию на коммутаторе на всех очередях включен алгоритм SP, команда включения отображается в конфигурации по умолчанию.

Включение алгоритма SP

Настроить на очередях алгоритм SP можно командой (включен по умолчанию):

```
(als_sw) #configure
(als_sw) (configure) #cos-queue strict 0 1 2 3 4 5 6 7
```

В параметрах эта команда принимает номера очередей, на которых нужно включить алгоритм SP. Последовательность номеров не имеет значения.

Включение алгоритма WRR

Для включения на очередях алгоритма WRR необходимо сначала отключить на них алгоритм SP:

```
(als_sw) #configure
(als_sw) (configure) #no cos-queue strict 0 1 2 3 4 5 6 7
```

Затем установить для этих очередей весовые коэффициенты для алгоритма WRR:

```
(als_sw) (configure) #cos-queue wrr weights 3 3 4 5 5 10 10 20
```

Последняя команда задает весовые коэффициенты для очередей с 0 по 7 последовательно. Значения задаются в диапазоне от 0 до 100. Эти значения являются весовыми коэффициентами для очередей и определяют, в каком соотношении пакеты из каждой очереди будут выходить с коммутатора. При вводе значения веса 0 для определенной очереди на этой очереди неявно включается режим SP.

Совместная настройка SP и WRR

Для совместной работы необходимо, аналогично предыдущим пунктам, настроить на части очередей алгоритм SP, на части — WRR:

```
(als_sw) #configure
(als_sw) (configure) #cos-queue strict 4 5 6 7
(als_sw) (configure) #no cos-queue strict 0 1 2 3
(als_sw) (configure) #cos-queue wrr weights 3 3 4 5 0 0 0 0
```

После выполнения этих команд очереди с 0 по 3 будут работать по алгоритму WRR с весовыми коэффициентами 3, 3, 4, 5. Остальные очереди, с 4 по 7, будут работать по алгоритму SP. Вводить вес 0 для очередей SP при настройке весов WRR не обязательно, независимо от введенных значений весов для очередей с включенным алгоритмом SP значения весов не учитываются.

В настройке существуют следующие особенности:

- если для очереди включен алгоритм SP, то весовой коэффициент не учитывается, даже если он задан;
- если для очереди включен алгоритм WRR, но весовой коэффициент

равен 0, то для очереди неявно включается алгоритм Strict;
- если для очереди включен алгоритм WRR, но весовые коэффициенты не установлены, очереди присваивается весовой коэффициент по умолчанию исходя из номера очереди: для очередей с 0 по 7 — веса 1, 2, 3, 4, 5, 6, 7 и 8 соответственно.

19.4. Настройка буфера пакетов

Буфер предназначен для временного хранения пакетов между их приемом и передачей, а также для сглаживания кратковременных всплесков трафика.

На коммутаторах АЛСиТЕК существует статический буфер, закрепленный за каждой очередью на каждый порт, а также общий динамический буфер, который может использоваться любой очередью дополнительно. Есть возможность отключения динамического буфера. Также есть возможность настройки буферов очередей.

Настройка статического режима

Шаг 1. Отключение динамического буфера

Для отключения динамического буфера необходимо выполнить команду:

```
(als_sw) (configure) #

(als_sw) (configure) #no cos-queue dynamic
```

Шаг 2. Настройка буферов очередей

Для каждой очереди указывается 2 лимита ячеек: лимит, при котором пакеты начинают отбрасываться, и лимит, при котором пакеты вновь начинают добавляться в очередь:

```
(als_sw) (configure) #cos-queue cell-limit 0 20 10
(als_sw) (configure) #cos-queue cell-limit 1 30 15
(als_sw) (configure) #cos-queue cell-limit 2 40 20
(als_sw) (configure) #cos-queue cell-limit 3 50 25
(als_sw) (configure) #cos-queue cell-limit 4 60 30
(als_sw) (configure) #cos-queue cell-limit 5 70 35
(als_sw) (configure) #cos-queue cell-limit 6 80 40
(als_sw) (configure) #cos-queue cell-limit 7 90 45
(als_sw) (configure) #exit
```

19.5. Настройка политик QoS

Данный механизм позволяет гибко настроить приоритизацию трафика по полям пакета. Работа производится в несколько этапов.

Сначала пакет классифицируется путем сравнения настроенных на коммутаторе признаков с полями пакета. За это отвечает класс трафика (class-map). Если для пакета найден подходящий класс, над ним выполняются связанные с этим классом действия. Действия описываются в политике (policy-map) применительно к каждому классу трафика.

Настройка подразумевает создание классов для нужного трафика, создание политик с действиями для выбранных классов и применение политик на интерфейсах.

На одном интерфейсе можно применить только одну политику. Одна и та же политика может быть применена на нескольких интерфейсах. Трафик, который не соответствует ни одному классу примененной на интерфейсе политики, будет направлен на обработку механизму доверия меткам. В то же время трафик, который попал под определенный класс примененной на интерфейсе политики QoS, не будет обработан механизмом доверия меткам.

Обратите внимание, что политики IPv4 и IPv6 QoS имеют разное предназначение. Политики IPv4 QoS предназначены для маркировки, перемаркировки, классификации IPv4 трафика и не IP протоколов: ARP, STP, LLDP и т.д. Политики IPv6 QoS предназначены для маркировки, перемаркировки, классификации только IPv6 трафика. Следовательно, с помощью политик IPv6 QoS невозможно классифицировать не IP протоколы ARP, STP, LLDP и т. д.

Создание классов трафика и политик IPv4 QoS

Шаг 1. Создание класса трафика

Класс трафика создается с помощью команды:

```
(als_sw) #configure
(als_sw) (configure) #class-map match-all "class1" ipv4
(als_sw) (configure) (class-map "class1") #
```

После создания класса трафика консольный интерфейс переходит в контекст настройки этого класса. В нем можно задать для класса признаки, которые будут проверяться у входящих пакетов. Строка "class1" — имя, идентифицирующее данный класс.

В качестве признаков могут быть выбраны следующие поля:

- Значение тега 802.1q (match vlan <1..4095>);
- * Значение приоритета 802.1p (match cos <0..7>);
- MAC-адреса источника и назначения (match source-address <mac>) и (match destination-address <mac>);
- Поле EtherType заголовка Ethernet (match ethertype <0x0600..0xffff>);
- IP-адреса источника и назначения (match srcip <ip>) и (match dstip <ip>);
- Поле protocol заголовка IPv4 (match protocol <0..255>);

- Поле DSCP байта TOS заголовка IPv4 (match ip dscp <0..63>);
- Поле TOS заголовка IPv4 значение и маска (match ip tos <0x00..0xFF> <0x00..0xFF>);
- Порты TCP или UDP источника и назначения (match srcl4port <0..65535>) и (match dstl4port <0..65535>);

В примере ниже класс описывает пакеты со значением 802.1q VLAN ID, равным 100 (VLAN 100 должен быть создан на коммутаторе до того, как будет указан в классе):

```
(als_sw) (configure) (class-map "class1") #match vlan 100
(als_sw) (configure) (class-map "class1") #exit
```

Пример использования в качестве признака 802.1p CoS:

```
(als_sw) (configure) #class-map match-all "class2" ipv4
(als_sw) (configure) (class-map "class2") #match cos 2
(als_sw) (configure) (class-map "class2") #exit
```

Существует признак, под который подходит любой трафик:

```
(als_sw) (configure) #class-map match-all "class3" ipv4
(als_sw) (configure) (class-map "class3") #match any
(als_sw) (configure) (class-map "class3") #exit
```

Может быть задана комбинация признаков (под класс в примере попадает трафик с VLAN ID 100 и CoS 5 одновременно):

```
(als_sw) (configure) #class-map match-all "class4" ipv4
(als_sw) (configure) (class-map "class4") #match vlan 100
(als_sw) (configure) (class-map "class4") #match cos 5
(als_sw) (configure) (class-map "class4") #exit
```

Шаг 2. Создание политики обработки трафика

Для создания политики обработки трафика применяется команда:

```
(als_sw) #configure
(als_sw) (configure) #policy-map "policy1" in
(als_sw) (configure) (policy-map "policy1") #
```

После выполнения этой команды будет создана политика с именем "policy1". Политика предполагает привязку действий к классам:

```
(als_sw) (configure) (policy-map "policy1") #class "class4"
(als_sw) (configure) (policy-map "policy1") (class "class4") #
```

После выполнения этой команды консольный интерфейс переходит в контекст настройки действий политики с именем "policy1" для определенного класса трафика с именем "class4". В этом контексте можно задать действия, которые в рамках этой политики будут производиться над пакетами, подпавшими под указанный класс.

Допустимые действия:

- assign-queue — назначить пакетам выходную очередь;
- mark cos — заменить значение 802.1p CoS в пакетах на указанное и назначить пакетам соответствующую выходную очередь;
- mark ip-dscp — изменение поля DSCP байта TOS заголовка IP на указанное;
- police-simple — ограничить на входе скорость прохождения трафика данного класса.

Действия **assign-queue** и **mark cos** являются взаимоисключающими в пределах действий для одного класса. При попытке настроить любое из этих действий при ранее настроенном другом будет выведена ошибка и новое действие не будет добавлено. В пределах одной политики, но для разных классов их настройка допустима.

Ограничение скорости **police-simple** подразумевает ввод верхнего порога скорости в кбит/с для трафика выбранного класса, ввод значения burst для ограничения скорости и указания двух действий: для трафика ниже ограничения и для трафика, превысившего ограничение. Всего доступен один вариант: трафик, не превысивший ограничение, пропускается (transmit), а трафик, превысивший ограничение — отбрасывается (drop).

Ниже приведены примеры настройки действий:

```
(als_sw) (configure) (policy-map "policy1") #class "class1"
(als_sw) (configure) (policy-map "policy1") (class "class1") #assign-queue 6
(als_sw) (configure) (policy-map "policy1") (class "class1") #exit
(als_sw) (configure) (policy-map "policy1") #class "class2"
(als_sw) (configure) (policy-map "policy1") (class "class2") #mark cos 3
(als_sw) (configure) (policy-map "policy1") (class "class2") #police-simple 500
0 64 conform-action transmit violate-action drop
(als_sw) (configure) (policy-map "policy1") (class "class2") #exit
(als_sw) (configure) (policy-map "policy1") #class "class3"
(als_sw) (configure) (policy-map "policy1") (class "class3") #
(als_sw) (configure) (policy-map "policy1") (class "class3") #mark ip-dscp 10
(als_sw) (configure) (policy-map "policy1") (class "class3") #exit
```

Если трафик подпадает под несколько классов, будут выполнены действия только для одного класса — для того, который был раньше добавлен в политику. Допустимо использовать одни и те же классы в разных политиках.

Шаг 3. Применение политики обработки трафика на интерфейсе

Для того, чтобы настроенная политика заработала, ее необходимо применить на интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #service-policy in "policy1"
```

Теперь весь трафик, входящий на интерфейс 0/1, будет обработан службой QoS по политике "policy1".

После описанных выше настроек конфигурация коммутатора примет следующий вид:

```
vlan database
vlan 100
exit

configure

class-map match-all class1 ipv4
match vlan 100
exit

class-map match-all class2 ipv4
match cos 2
exit

class-map match-all class3 ipv4
match any
exit

class-map match-all class4 ipv4
match vlan 100
match cos 5
exit

policy-map policy1 in

class class1
assign-queue 6
exit

class class2
police-simple 5000 64 conform-action transmit violate-action drop
mark cos 3
exit

class class3
mark ip-dscp 10
exit

exit

interface 0/1
service-policy in policy1
exit

exit
```

С данными настройками механизм будет работать следующим образом:

1. Трафику класса "class1" (VLAN ID 100) будет назначена выходная очередь с номером 6.
2. Трафику класса "class2" (cos 2) будет назначена выходная очередь номер 3, а метка cos будет изменена на 3. Кроме того, прохождение этого трафика будет ограничено скоростью 5 Мбит/с.
3. Трафик класса "class3" получит метку IP DSCP 10.

Создание классов трафика и политик IPv6 QoS

Шаг 1. Создание класса трафика

Класс трафика создается с помощью команды:

```
(als_sw) #configure
(als_sw) (configure) #class-map match-all "class1" ipv6
(als_sw) (configure) (class-map "class1") #
```

После создания класса трафика консольный интерфейс переходит в контекст настройки этого класса. В нем можно задать для класса признаки, которые будут проверяться у входящих пакетов. Строка "class1" — имя, идентифицирующее данный класс.

В качестве признаков могут быть выбраны следующие поля:

- Значение тега 802.1q (match vlan <1..4095>);
- Значение приоритета 802.1p (match cos <0..7>);
- IPv6-адреса источника и назначения (match ipv6 source-address <ipv6>) и (match ipv6 destination-address<ipv6>);
- Поле next-header заголовка IPv6 (match ipv6 next-header <0..255>);
- Поле DSCP байта Traffic Class заголовка IPv6 (match ipv6 dscp <0..63>);
- Поле Traffic Class заголовка IPv6 значение и маска (match ip tc <0x00..0xFF> <0x00..0xFF>);
- Порты TCP или UDP источника и назначения (match l4 source-port <0..65535>) и (match l4 destination-port <0..65535>);

В примере ниже класс описывает пакеты со значением 802.1q VLAN ID, равным 100 (VLAN 100 должен быть создан на коммутаторе до того, как будет указан в классе):

```
(als_sw) (configure) (class-map "class1") #match vlan 100
(als_sw) (configure) (class-map "class1") #exit
```

Пример использования в качестве признака IPv6 адрес источника.

```
(als_sw) (configure) #class-map match-all "class2" ipv6
(als_sw) (configure) (class-map "class2") #match ipv6 source-address fe80::/10
(als_sw) (configure) (class-map "class2") #exit
```

Существует признак, под который подходит любой трафик:

```
(als_sw) (configure) #class-map match-all "class3" ipv6
(als_sw) (configure) (class-map "class3") #match any
(als_sw) (configure) (class-map "class3") #exit
```

Может быть задана комбинация признаков (под класс в примере попадает трафик с VLAN ID 100 и Link Local адресом источника, ICMPv6 (Next header 58)):

```
(als_sw) (configure) #class-map match-all "class4" ipv6
(als_sw) (configure) (class-map "class4") #match vlan 100
(als_sw) (configure) (class-map "class4") #match ipv6 source-address fe80::/10
(als_sw) (configure) (class-map "class4") #match ipv6 next-header 58
(als_sw) (configure) (class-map "class4") #exit
```

Шаг 2. Создание политики обработки трафика

Для создания политики обработки трафика применяется команда:

```
(als_sw) #configure
(als_sw) (configure) #policy-map "policy1" in
(als_sw) (configure) (policy-map "policy1") #
```

После выполнения этой команды будет создана политика с именем "policy1". Политика предполагает привязку действий к классам:

```
(als_sw) (configure) (policy-map "policy1") #class "class4"
(als_sw) (configure) (policy-map "policy1") (class "class4") #
```

После выполнения этой команды консольный интерфейс переходит в контекст настройки действий политики с именем "policy1" для определенного класса трафика с именем "class4". В этом контексте можно задать действия, которые в рамках этой политики будут производиться над пакетами, подпавшими под указанный класс.

Допустимые действия:

- assign-queue — назначить пакетам выходную очередь;
- mark cos — заменить значение 802.1p CoS в пакетах на указанное и назначить пакетам соответствующую выходную очередь;
- mark ip-dscp — изменение поля DSCP байта TOS заголовка IP на указанное;
- police-simple — ограничить на входе скорость прохождения трафика данного класса;

Действия **assign-queue** и **mark cos** являются взаимоисключающими в пределах действий для одного класса. При попытке настроить любое из этих действий при ранее настроенном другом будет выведена ошибка и новое действие не будет добавлено. В пределах одной политики, но для разных классов их настройка допустима.

Ограничение скорости **police-simple** подразумевает ввод верхнего порога скорости в килобитах в секунду для трафика выбранного класса, ввод значения burst для ограничения скорости и указания двух действий: для трафика ниже ограничения и для трафика, превысившего ограничение. Всего доступен один вариант: трафик, не превысивший ограничение, пропускается (transmit), а трафик, превысивший ограничение — отбрасывается (drop).

Ниже приведены примеры настройки действий:

```
(als_sw) (configure) (policy-map "policy1") #class "class1"
(als_sw) (configure) (policy-map "policy1") (class "class1") #assign-queue 6
(als_sw) (configure) (policy-map "policy1") (class "class1") #exit
(als_sw) (configure) (policy-map "policy1") #class "class2"
(als_sw) (configure) (policy-map "policy1") (class "class2") #mark cos 3
(als_sw) (configure) (policy-map "policy1") (class "class2") #police-simple 500
0 64 conform-action transmit violate-action drop
(als_sw) (configure) (policy-map "policy1") (class "class2") #exit
(als_sw) (configure) (policy-map "policy1") #class "class3"
(als_sw) (configure) (policy-map "policy1") (class "class3") #
(als_sw) (configure) (policy-map "policy1") (class "class3") #drop
(als_sw) (configure) (policy-map "policy1") (class "class3") #exit
```

Если трафик подпадает под несколько классов, будут выполнены действия только для одного класса — для того, который был раньше добавлен в политику. Допустимо использовать одни и те же классы в разных политиках.

Шаг 3. Применение политики обработки трафика на интерфейсе

Для того, чтобы настроенная политика заработала, ее необходимо применить на интерфейсе:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #service-policy in "policy1"
```

Теперь весь трафик, входящий на интерфейс 0/1, будет обработан службой QoS по политике "policy1".

После описанных выше настроек конфигурация коммутатора примет следующий вид:

```
vlan database
vlan 100
exit

configure

class-map match-all class1 ipv6
match vlan 100
exit

class-map match-all class2 ipv6
match ipv6 source-address fe80::/10
exit

class-map match-all class3 ipv6
match any
exit

class-map match-all class4 ipv6
match vlan 100
match ipv6 source-address fe80::/10
match ipv6 next-header 58
exit

policy-map policy1 in

class class1
assign-queue 6
exit

class class2
police-simple 5000 64 conform-action transmit violate-action drop
mark cos 3
exit

class class3
mark ip-dscp 10
exit

exit

interface 0/1
service-policy in policy1
exit

exit
```

С данными настройками механизм будет работать следующим образом:

1. Трафику класса "class1" (VLAN ID 100) будет назначена выходная очередь с номером 6.
2. Трафику класса "class2" (Link Local адрес источника) будет назначена выходная очередь номер 3, а метка cos будет изменена на 3. Кроме того, прохождение этого трафика будет ограничено скоростью 5 Мбит/с.
3. Трафик класса "class3" получит метку IP DSCP 10.

19.6. Доверие меткам CoS

Механизм задействуется в случае, если пакету не была ранее назначена очередь механизмом классификации пакетов. Интерфейс коммутатора может находиться в трех режимах: доверие меткам 802.1p (CoS) в пакетах, доверие меткам DSCP в IP-заголовке пакетов и отсутствие доверия меткам.

По умолчанию доверие меткам отключено. При отключенном режиме доверия всем входящим пакетам независимо от меток 802.1p или DSCP назначается очередь, указанная в таблице соответствия для метки 802.1p, равной 0. По умолчанию это очередь номер 1.

При включении режима доверия меткам 802.1p при получении пакета на базе его метки 802.1p по таблице соответствия выбирается выходная очередь. Если пакет не имеет тега 802.1p/802.1q, ему назначается очередь, указанная в таблице соответствия для метки 802.1p, равной 0. По умолчанию это очередь номер 1.

При включении режима доверия меткам DSCP в IP-заголовке пакетов по таблице соответствия выбирается выходная очередь. Если пакет не имеет IP-заголовка, ему назначается очередь, указанная в таблице соответствия для метки 802.1p, равной 0. По умолчанию это очередь номер 1.

Настройка доверия меткам CoS

Настройка заключается в назначении общего режима доверия и редактировании таблицы соответствия 802.1p (CoS) и выходных очередей. CoS может принимать значения от 0 до 7.

Шаг 1. Просмотр текущего состояния

Посмотреть текущее состояние режима доверия можно командой:

```
(als_sw) #show classofservice trust

 Interface  Trust Mode
 ---------  ------------------------------
  0/1        Untrusted. Traffic Class: 1
  0/2        Untrusted. Traffic Class: 1
  0/3        Untrusted. Traffic Class: 1
  ...
  0/28       Untrusted. Traffic Class: 1
```

В этой таблице перечислены все интерфейсы и текущий режим доверия на этих интерфейсах.

Посмотреть таблицу соответствия 802.1p и выходных очередей можно командой:

```
(als_sw) #show classofservice dot1p-mapping

 802.1p tag  Egress queue
 ----------  -------------
     0          1
     1          0
     2          0
     3          1
     4          2
     5          2
     6          3
     7          3
```

В примере приведено состояние по умолчанию. Например, пакеты с CoS 0 попадут в очередь с номером 1, а пакеты с CoS 6 попадут в очередь 3. Распределение может быть настроено произвольное.

Шаг 2. Задание режима доверия

Для смены режима доверия используется команда:

```
(als_sw) #configure
(als_sw) (configure) #classofservice trust (dot1p|ip-dscp|untrusted)
```

Режимы:

- dot1p — режим доверия меткам 802.1p. Очередь каждого пакета определяется значением из таблицы соответствия CoS и номера очереди;
- ip-dscp — режим доверия меткам DSCP. Очередь пакета определяется по таблице соответствия DSCP и номера очереди;
- untrusted — отключение режима доверия меткам 802.1p. Все пакеты помещаются в одну очередь и имеют одинаковый приоритет. Номер очереди определяется значением соответствия для CoS 0. По умолчанию CoS 0 соответствует номер очереди 1, поэтому все пакеты при установке этого режима будут помещены в очередь 1.

Обратите внимание, что режим untrusted не выключает службу. В этом режиме все пакеты имеют один приоритет, помещаются в одну очередь. Чтобы изменить номер очереди для всего трафика в режиме untrusted, нужно в таблице соответствия изменить номер очереди, соответствующей CoS 0.

Шаг 3. Задание режима доверия на интерфейсе (опционально)

Для интерфейса можно задать режим доверия, отличный от заданного глобально. Настройки на интерфейсе необязательны и являются более приоритетными, чем глобальная настройка.

Задать режим доверия для интерфейса можно командой:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #classofservice trust (dot1p|ip-dscp|untru
sted)
```

Шаг 4. Задание соответствия значения CoS номеру определенной очереди

Таблица соответствия определенного значения CoS и номера очереди может быть настроена произвольно. Изменения, сделанные в контексте глобальной конфигурации, коснутся всех интерфейсов.

Для редактирования глобальной таблицы соответствия используется команда вида:

```
(als_sw) #configure
(als_sw) (configure) #classofservice dot1p-mapping <cos> <queue>
```

В этой команде нужно указать желаемое соответствие значения CoS и номера очереди, в которую будут помещены пакеты с данным CoS.

Для примера установим полное соответствие значений CoS номерам очередей:

```
(als_sw) #configure
(als_sw) (configure) #classofservice dot1p-mapping 0 0
(als_sw) (configure) #classofservice dot1p-mapping 1 1
(als_sw) (configure) #classofservice dot1p-mapping 2 2
(als_sw) (configure) #classofservice dot1p-mapping 3 3
(als_sw) (configure) #classofservice dot1p-mapping 4 4
(als_sw) (configure) #classofservice dot1p-mapping 5 5
(als_sw) (configure) #classofservice dot1p-mapping 6 6
(als_sw) (configure) #classofservice dot1p-mapping 7 7
(als_sw) (configure) #exit
```

Теперь таблица соответствия примет следующий вид:

```
(als_sw) #show classofservice dot1p-mapping

 802.1p tag  Egress queue
 ----------  -------------
     0           0
     1           1
     2           2
     3           3
     4           4
     5           5
     6           6
     7           7
```

Шаг 5. Задание соответствия значения CoS номеру определенной очереди на интерфейсе (опционально)

Для интерфейса можно задать соответствие значения CoS номеру определенной очереди, которое будет отлично от глобальной настройки. Настройка на интерфейсе не обязательна и является более приоритетной, чем глобальная настройка. Если на интерфейсе указаны не все соответствия, а только часть (например, для cos 2 и 3), то остальная часть таблицы будет взята из глобальной настройки.

Задать соответствие можно командой:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #classofservice dot1p-mapping <cos> <queue
>
```

19.7. Доверие меткам DSCP

Механизм аналогичен доверию меткам 802.1p, только в данном случае за назначение выходной очереди отвечает метка DSCP из IP заголовка. DSCP (differentiated services code point) — это старшие 6 бит 8-ми битного поля Differentiated services Field в IP-заголовке. Поле DSCP может принимать числовые значения от 0 до 63.

Настройка доверия меткам DSCP

- включение режима доверия DSCP меткам (по умолчанию действует режим доверия 802.1p)
- назначение соответствия значений DSCP и номеров выходных очередей

Шаг 1. Просмотр текущего состояния

Посмотреть текущее состояние режима доверия можно командой:

```
(als_sw) #show classofservice trust

 Interface  Trust Mode
 ---------  ------------------------------
  0/1        Dot1P
  0/2        Dot1P
  0/3        Dot1P
...
  0/28       Dot1P
```

По умолчанию включен режим доверия меткам 802.1p.

Посмотреть таблицу соответствия значения DSCP и номера выходной очереди можно командой:

```
(als_sw) #show classofservice ip-dscp-mapping

  IP DSCP    Egress queue
  ---------  ------------
  0(be/cs0)  1
...
  6          1
  7          0
...
  22(af23)   0
  23         1
...
  29         1
  30(af33)   2
...
  44         2
  45         3
...
  60         3
  61         0
  62         0
  63         0
```

Шаг 2. Задание режима доверия

Для установки режима доверия меткам DSCP используется команда:

```
(als_sw) #configure
(als_sw) (configure) #classofservice trust ip-dscp
```

Данная команда устанавливает для всех интерфейсов коммутатора режим доверия меткам DSCP. При получении пакета по его метке DSCP пакету назначается определенная выходная очередь из таблицы соответствия, приведенной выше. В случае, если пакет не имеет метки DSCP (например, пакет не имеет заголовка IP), ему назначается выходная очередь, которая соответствует значению 0 из таблицы соответствия 802.1p (аналогично режиму untrusted).

Шаг 3. Задание режима доверия на интерфейсе (опционально)

Для установки режима доверия меткам DSCP на интерфейсе используется команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #classofservice trust ip-dscp
```

Эта команда может быть полезна в случае, если необходимо глобально включить один режим доверия (например, доверие меткам 802.1p), а на определенном интерфейсе — режим доверия меткам DSCP.

Шаг 4. Задание соответствия значения DSCP номеру определенной очереди

Для редактирования таблицы соответствия используется команда вида:

```
(als_sw) #configure
(als_sw) (configure) #classofservice ip-dscp-mapping <dscp> <queue>
```

Параметры:

- <dscp> — значение метки DSCP. Допустимо вводить числовые значения от 0 до 63, а также символьные имена (приведены в таблице ниже);
- <queue> — значение очереди для данного DSCP.

Некоторые значения DSCP имеют собственное название и являются сокращениями от английских слов Assured Forwarding (AF) и Class Selector (CS). Подробнее про них можно почитать в RFC 2474 и связанных документах.

Символьные константы для назначения DSCP:

Символьная константа	Числовое значение
af11	10
af12	12
af13	14
af21	18
af22	20
af23	22
af31	26
af32	28
af33	30
af41	34
af42	36
af43	38
cs0	0

cs1	8
cs2	16
cs3	24
cs4	32
cs5	40
cs6	48
cs7	56

Для примера изменим таблицу для нескольких DSCP:

```
(als_sw) #configure
(als_sw) (configure) #classofservice ip-dscp-mapping 0 0
(als_sw) (configure) #classofservice ip-dscp-mapping 37 4
(als_sw) (configure) #classofservice ip-dscp-mapping cs6 6
(als_sw) (configure) #classofservice ip-dscp-mapping 32 3
```

Шаг 5. Задание номера очереди (внутреннего приоритета) для пакетов без IP заголовка

Поскольку при доверии меткам 802.1p и DSCP некоторые виды трафика не попадают под условия правил доверия, такие виды трафика будут направлены в очередь по умолчанию. Это касается трафика без тега при режиме доверия меткам 802.1p и трафику без IP-заголовку, то есть без метки DSCP. Такой трафик помещается в очередь, с которой связано значение 0 в таблице соответствия метки 802.1p и внутренней очереди. Обратите внимание, что по умолчанию значение очереди для 0 метки 802.1p равно 1.

Изменить это значение можно следующей командой:

```
(als_sw) #configure
(als_sw) (configure) #classofservice dot1p-mapping 0 <queue>
```

19.8. Ограничение скорости порта на выходе

В некоторых случаях может понадобиться ограничить скорость порта на выходе. Для этой цели служит следующая команда:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (configure) (interface 0/1) #traffic-shape 5000
```

В данном примере скорость интерфейса 0/1 ограничена 5 Мбит/с. Минимальное значение ограничения равно 64 кбит/с, максимальное равно скорости порта. Алгоритмы планировщика и политики QoS будут учитывать выходную пропускную способность порта при отправке трафика, в том числе и установленное этой командой ограничение скорости.

19.9. Примеры типовых настроек

Как правило, на коммутаторе доступа существует не менее 3 VLAN: для управления, для интернета и для цифрового телевидения (IPTV). При этом обычно максимальный приоритет дается управляющему трафику (по VLAN) — устройство должно оставаться доступным при любой нагрузке. Для этого он направляется в очередь номер 7. Следующим по приоритету настраивается IPTV — на изображение не должно влиять использование интернета. Для него обычно используется очередь 5 или 6. Для интернета обычно оставляется очередь по умолчанию (1), как самая низкоприоритетная из трех используемых. Реализуется это созданием двух классов для выделения управляющего трафика и трафика IPTV по VLAN ID и политики, в которой для этих классов трафика назначается очередь. Политика с приоритизацией трафика применяется на uplink-коммутатора.

Пример конфигурации:

```
network mgmt_vlan 1000

vlan database
vlan 222,640,1000
vlan name 222 "IPTV"
vlan name 640 "INET"
vlan name 1000 "MGMT"
exit

configure

mcast_vfm 222 forward_registered

cos-queue strict 0 1 2 3 4 5 6 7

class-map match-all mgmt ipv4
match vlan 1000
exit
class-map match-all iptv ipv4
match vlan 222
exit

policy-map qos_policy in
class mgmt
assign-queue 7
exit
class iptv
assign-queue 5
exit
exit

interface 0/1
description "CLIENT"
vlan pvid 640
vlan participation include 222,640
vlan tagging 222
exit

interface 0/28
description "UPLINK"
vlan participation include 222,640,1000
vlan tagging 222,640,1000
vlan participation exclude 1
service-policy in qos_policy
exit

exit
```

ГЛАВА 20. IPV6 SNOOPING

20.1. Введение в IPv6 Snooping

В IPv6-сетях важную роль играет протокол ICMPv6. Данный протокол используется для следующих целей:

- получение динамического адреса при помощи SLAAC (IPv6 Stateless Address AutoConfiguration);
- получение динамического адреса по DHCPv6 (совместно с протоколом DHCPv6);
- разрешение MAC-адреса устройства (аналог ARP в IPv4-сетях);
- получение информации о маршрутизаторе, маршрутах и других сетевых параметрах.

Типы адресов IPv6

В IPv6-сетях различают следующие типы IPv6-адресов:

- unicast — данные адреса используются для соединений двух устройств между собой;
    - global — адреса используются в глобальной сети Интернет, в настоящий момент это префикс 2000::/3;
    - link-local — адреса из подсети fe80::/10 для связи устройств внутри IPv6-сегмента (до маршрутизатора);
- anycast — используются для отправки пакета хотя бы одному устройству из группы;
- multicast — используются для отправки пакета множеству адресатов.

Типы пакетов ICMPv6

IPv6 Snooping отслеживает следующие пакеты ICMPv6:

- Neighbor Solicitation;
- Neighbor Advertisement;
- Router Solicitation;
- Router Advertisement.

ICMPv6 Neighbor Solicitation

Используется клиентом для разрешения MAC-адреса устройства (аналог ARP в IPv4-сетях). В этом случае IPv6-адрес источника будет ненулевым. В случае если IPv6-адрес источника нулевой (::), данный пакет используется для получения первоначального адреса (SLAAC) и подтверждения его уникальности (DAD, Duplicate Address Detection). Обычно IPv6 Snooping не блокирует пакеты ICMPv6 Neighbor Solicitation, так как если запретить их прохождение, в сети могут появиться узлы с одинаковыми IPv6-адресами. Также невозможно будет определить MAC-адрес устройства для начала обмена сообщениями с ним.

ICMPv6 Neighbor Advertisement

Используется клиентом в качестве ответа на ICMPv6 Neighbor Solicitation. Данный пакет отправляет клиент, адрес которого указан в ICMPv6 Neighbor Solicitation. В пакете также содержится MAC-адрес клиента.

ICMPv6 Router Solicitation

Отправляется клиентом сразу после настройки link-local адреса после подключения к сети. Оповещает маршрутизатор о том, что в сети появился новый клиент, и является предложением маршрутизатору ускорить отправку пакета ICMPv6 Router Advertisement. Маршрутизатор имеет право проигнорировать данный пакет.

ICMPv6 Router Advertisement

Используется маршрутизатором для рассылки информации о шлюзе, маршрутах, допустимых IPv6-префиксах. Отправляется с определенной периодичностью, например, раз в 30 секунд. Маршрутизатор может сделать внеочередную отправку ICMPv6 Router Advertisement при получении пакета ICMPv6 Router Solicitation (например, при подключении нового клиента).

Получение IPv6-адреса при помощи SLAAC

Рассмотрим ситуацию, когда IPv6-клиент без статических IPv6-адресов и с включенной автонастройкой был подключен к некоторому IPv6-сегменту сети с IPv6-маршрутизатором.

Работа механизма подразумевает автоматическую настройку IPv6-адреса в какой-либо из IPv6-подсетей. Обычно адрес генерируется на основе IPv6-префикса и MAC-адреса сетевого интерфейса.

В общем случае получение адреса можно разбить на следующие этапы:

- получение префикса от маршрутизатора (при получении глобального адреса, в случае link-local данный этап пропускается);
- генерация IPv6-адреса с полученным префиксом (или link-local) на основе MAC-адреса сетевого интерфейса с использованием некоторой случайной величины;
- отправка ICMPv6 Neighbor Solicitation с нулевым адресом источника и сгенерированным IPv6-адресом в качестве target-адреса для проверки занятости адреса;
- ожидание ICMPv6 Neighbor Advertisement в течение некоторого времени (обычно не более 5-10 секунд);
    - установка адреса на интерфейс, если ICMPv6 Neighbor Advertisement в течение заданного времени не был получен;
    - повторение алгоритма, если был получен ICMPv6 Neighbor Advertisement. Это означает, что данный адрес уже используется другим устройством.

Получение link-local адреса

Вначале клиент генерирует link-local IPv6-адрес на основе своего MAC-адреса. Затем отправляет пакет ICMPv6 Neighbor Solicitation с нулевым адресом источника и сгенерированным IPv6-адресом в качестве target-адреса. Если за определенное время ни один клиент в данной подсети не ответит на данный пакет, считается, что адрес свободен, и клиент применяет его на интерфейсе. Если же клиент получает ICMPv6 Neighbor Advertisement, то генерируется другой адрес на основе MAC и алгоритм повторяется.

Когда у клиента появляется link-local адрес, клиент может обмениваться данными с другими клиентами в рамках IPv6-сегмента (участка сети до маршрутизатора). Чтобы обмениваться данными с клиентами, расположенными за маршрутизатором, требуется получить информацию о шлюзе и используемых префиксах. Кроме того, для обмена данными с глобальной сетью необходим global-адрес — в IPv6 не предусмотрен аналог NAT в IPv4-сетях.

Получение информации о маршрутизаторе и используемых маршрутах

Клиент отправляет пакет ICMPv6 Router Solicitation, при этом в качестве адреса источника используется link-local адрес, полученный на предыдущем шаге. Данный пакет отправляется как попытка ускорить отправку пакета ICMPv6 Router Advertisement маршрутизатором.

После получения ICMPv6 Router Advertisement клиент считывает информацию о шлюзе и префиксах.

Для каждого префикса клиент получает адрес по процедуре, описанной в предыдущем пункте, но уже не в подсети link-local, а в подсетях, полученных от маршрутизатора. Важно отметить, что маршрутизатор может раздавать адреса в любых unicast-подсетях, это может быть как global-адрес, так и адрес link-local с другим префиксом, например внутренним префиксом организации или офиса.

Также механизм SLAAC не обязывает маршрутизатор контролировать получение адреса клиентом и его уникальность в пределах подсети — этим должен заниматься сам клиент.

Информация о времени аренды адреса

Адреса, сгенерированные на основе префиксов маршрутизатора, даются клиенту в аренду. Время аренды обычно указывается для каждого префикса внутри пакета ICMPv6 Router Advertisement. Существует Valid lifetime и Preferred lifetime.

- **Valid lifetime** — время, в течение которого адрес может быть использован устройством. После его истечения использовать адрес нельзя;
- **Preferred lifetime** — предпочтительное время использования адреса — время, в течение которого адрес может быть использован для установления новых соединений. После его истечения адрес не следует использовать для новых соединений.

20.2. IPv6 Snooping на коммутаторах АЛСиТЕК

IPv6 Snooping на коммутаторах АЛСиТЕК выполняет следующие функции:

- отслеживание получения адреса с помощью механизма SLAAC;
- отслеживание информации о маршрутизаторах;
- отслеживание информации о префиксах;
- блокировка пакетов ICMPv6 Router Advertisement с недоверенных интерфейсов.

Информацию о текущих маршрутизаторах и префиксах коммутатор получает в ходе анализа пакетов ICMPv6 Router Advertisement. Пакеты ICMPv6 Router Advertisement с недоверенных интерфейсов будут заблокированы коммутатором и информация из них не будет добавлена в таблицу IPv6 Snooping.

Информация о получении адреса с помощью механизма SLAAC может быть использована службой IPv6 Source Guard для предотвращения выхода в сеть с IPv6-адресом без предварительного получения адреса с использованием SLAAC.

Информация о префиксах используется для получения времени жизни адресов в префиксах. Также информация о префиксах может быть использована IPv6 Source Guard для разрешения только тех префиксов, которые раздает IPv6-маршрутизатор.

IPv6 Snooping не блокирует пакеты ICMPv6 Neighbor Solicitation, ICMPv6 Neighbor Advertisement, так как блокировка данных пакетов помешает работе механизма IPv6 DAD.

IPv6 Snooping не блокирует пакеты ICMPv6 Router Solicitation, так как они не влияют напрямую на состояние сети, а лишь являются рекомендацией для маршрутизаторов отправить ICMPv6 Router Advertisement быстрее.

Настройка IPv6 Snooping

Шаг 1. Настройка VLAN на устройстве

Для корректной работы IPv6 Snooping на коммутаторах АЛСиТЕК требуется произвести предварительные настройки.

Создаем VLAN 10, в котором будет осуществляться передача данных по протоколу IPv6:

```
(als_sw) #vlan database
(als_sw) (Vlan) #vlan 10
(als_sw) (Vlan) #exit
```

Настраиваем VLAN 10 на клиентских интерфейсах:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1,0/2
(als_sw) (configure) (interface 0/1-0/2) #vlan participation include 10
```

```
(als_sw) (configure) (interface 0/1-0/2) #vlan tagging 10
```

Настраиваем VLAN 10 на серверных интерфейсах:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #vlan participation include 10
(als_sw) (configure) (interface 0/28) #vlan tagging 10
```

Шаг 2. Включение службы IPv6 Snooping на устройстве

Для глобального включения службы выполняем команду:

```
(als_sw) #configure
(als_sw) (configure) #ipv6 snooping
```

Шаг 3. Включение IPv6 Snooping на VLAN

Для включения службы на созданном VLAN выполняем команду:

```
(als_sw) #configure
(als_sw) (configure) #ipv6 snooping vlan 10
```

Шаг 4. Назначение доверенных интерфейсов IPv6 Snooping

Назначаем доверенные интерфейсы, за которыми расположены IPv6-маршрутизаторы, следующей командой:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/28
(als_sw) (configure) (interface 0/28) #ipv6 snooping trust
```

Все интерфейсы, которые не настроены как доверенные, считаются клиентскими интерфейсами.

Просмотр IPv6-соседей

IPv6-соседей можно посмотреть с помощью следующей команды:

```
(als_sw) #show ipv6 snooping neighbors

  VLAN  Neighbor                          Iface  MAC               Valid   Pref
                                                                   Life    Life
  ----  --------------------------------  -----  ----------------  ------  -----
-
  10    2001:470:1f15:2db:fedd:efec:      0/1    00:13:aa:00:00:01  60      30
        5fe9:9dad
  10    fe80::4f3c:ae9:d3d1:b639          0/1    00:13:aa:00:00:01  7182    7182
```

Поля:

- VLAN — VLAN, в котором обнаружен сосед;
- Neighbor — IPv6-адрес соседа;
- Iface — интерфейс, за которым обнаружен сосед;
- Valid Life — время аренды адреса, в течение которого адрес будет валиден, в секундах;
- Pref Life — предпочтительное время использования адреса в секундах.

Для команды доступен фильтр по VLAN, интерфейсу и IPv6-префиксу.

Просмотр IPv6-маршрутизаторов

Список обнаруженных IPv6-маршрутизаторов можно посмотреть при помощи команды:

```
(als_sw) #show ipv6 snooping routers

  VLAN  Router                                Iface  MAC               Route
r
                                                                       Life
  ----  ------------------------------------  -----  ----------------  -----
-
  10    fe80::0213:aaff:fe00:01               0/28   00:13:aa:00:00:01  30
```

Поля:

- VLAN — VLAN, в котором был обнаружен IPv6-маршрутизатор;
- Router — IPv6-адрес маршрутизатора;
- Iface — интерфейс, за которым расположен маршрутизатор;
- MAC — MAC-адрес маршрутизатора;
- Router Life — время жизни маршрутизатора в секундах. По истечении времени жизни маршрутизатор будет удален из таблицы.

Также можно посмотреть информацию о префиксах, которые анонсирует маршрутизатор:

```
(als_sw) #show ipv6 snooping routers detail fe80::0213:aaff:fe00:01 vlan 10

  Prefix                                    Valid     Prefered
                                            Life      Life
  ----------------------------------------  --------  --------
  2001:470:1f15:2db::/64                    10000     14400
```

Поля:

- Prefix — префиксы, которые анонсирует маршрутизатор;
- Valid Life — время аренды адреса, в течение которого адрес будет валиден, в секундах;
- Pref Life — предпочтительное время использования адреса в секундах.

Просмотр IPv6-префиксов

Все префиксы сегмента IPv6-сети можно посмотреть с помощью следующей команды:

```
(als_sw) #show ipv6 snooping prefixes

  VLAN  Router                      Prefixes                    Valid   Pref

                                                                Life    Life

  ----  --------------------------  --------------------------  ------  -----
-
  10    fe80::0213:aaff:fe00:01     2001:db8:1::/64             86400   14400

                                    2001:dead:1::/64            86400   14400
```

Поля:

- VLAN — VLAN, в котором маршрутизатор анонсирует данные IPv6-префиксы;
- Router — IPv6-адрес маршрутизатора;
- Prefixes — IPv6-префиксы, которые анонсирует маршрутизатор;
- Valid Life — время аренды адреса, в течение которого адрес будет валиден, в секундах;
- Pref Life — предпочтительное время использования адреса в секундах.

Для команды доступен фильтр по VLAN и интерфейсу.

ГЛАВА 21. РАБОТА С ДАТЧИКАМИ

21.1. Температурный датчик

Настройка температурного датчика

Коммутатор может отправлять SNMP-trap сообщения о выходе текущего значения температурного датчика за нижнюю и верхнюю границы установленного температурного диапазона. По умолчанию на коммутаторе отключена отправка SNMP-trap сообщений об изменении состояния температуры.

Шаг 1. Просмотр текущего состояния

Для просмотра текущего значения температуры на плате используется команда:

```
(als_sw) #show box temperature

Temperature sensor       : 32 C
SNMP Trap                : disable
Lower threshold          : 0 C
Upper threshold          : 45 C
```

- Temperature sensor — текущая температура в градусах Цельсия;
- SNMP Trap — настройка отправки SNMP-trap сообщений;
- Lower threshold — нижняя граница температурного диапазона;
- Upper threshold — верхняя граница температурного диапазона.

Шаг 2. Изменение нижней границы температурного диапазона

Для изменения используется команда:

```
(als_sw) #configure
(als_sw) (configure) #box
(als_sw) (configure) (box) #temperature threshold min 15
```

Шаг 3. Изменение верхней границы температурного диапазона

Для изменения используется команда:

```
(als_sw) #configure
(als_sw) (configure) #box
(als_sw) (configure) (box) #temperature threshold max 37
```

Шаг 4. Включение отправки SNMP-trap сообщений

Для включения отправки SNMP-trap сообщений используется команда:

```
(als_sw) #configure
(als_sw) (configure) #box
(als_sw) (configure) (box) #temperature snmp-trap
```

Шаг 5. Просмотр изменений

Проверим изменения с помощью команды:

```
(als_sw) #show box temperature

Temperature sensor       : 32 C
SNMP Trap                : enable
Lower threshold          : 15 C
Upper threshold          : 37 C
```

При данной конфигурации SNMP-trap сообщение будет отправлено в случае, если температура опустится ниже 15 градусов Цельсия, либо возрастет выше 37 градусов. Текущее значение температуры и значение порога записывается в SNMP-trap.

ГЛАВА 22. ETHERNET OAM

22.1. Введение в Ethernet OAM

OAM (operations, administration and management) — протокол, предназначенный для управления, администрирования и контроля функционирования оборудования. Определен стандартом IEEE 802.3ah-2004. Включает в себя следующие технологии:

- Discover — обнаружение соседних устройств с включенным OAM;
- Unidirection — технология оповещение удаленной стороны об однонаправленном линке;
- Link Fault Monitor — обнаружение проблем с линком, без прерывания пользовательских сервисов;
- Remote Loopback — инструмент для замыкания петель на удаленном оборудовании с целью локализации проблемы или тестирования производительности;
- MIB retrieval — технология инкапсуляции SNMP протокола в OAM пакеты;
- Dying gasp — оповещение соседей о неисправимой проблеме, которая приведет к прекращению предоставления услуг.

Важно отметить, что в стандарте не все перечисленные технологии являются обязательными для реализации в конечных устройствах.

Технология Discover

Discover — технология, позволяющая обнаружить и установить связь с соседями, находящимися в пределах одного сегмента линка от устройства. Данная технология позволяет обнаружить соседние устройства, а также узнать о поддержке этими устройствами технологий OAM.

По стандарту устройства должны иметь поддержку двух режимов работы функции discover: активный и пассивный. В активном режиме работы устройство будет периодически посылать тестовые OAM пакеты для обнаружения соседей. В пассивном режиме работы OAM пакеты будут отсылаться только в ответ на пришедшие OAM пакеты.

Технология Link Fault Monitor

Link Fault Monitor — технология, позволяющая обнаруживать проблемы с линком, без прерывания пользовательских сервисов.

Данная технология начинает работать после успешного завершения обнаружения устройств с включенным OAM (технология Discover). Устройства периодически отправляют небольшое количество тестовых данных через линк и контролируют целостность данных, задержку передачи данных. В случае если наблюдаются потери данных или критическое увеличение задержки при прохождении данных, линк помечается как нерабочий. Об этом уведомляется удаленное устройство с помощью посылки OAM пакета со специальным флагом.

Информация о нерабочем линке может быть использована для:

- Перезапуска автоопределения скорости. Например если возникли проблемы с согласованием дуплекса (duplex mismatch);
- Перестроения дерева Spanning Tree, для исключения сбойного сегмента из топологии;
- Исключения интерфейса из агрегированного LAG интерфейса;
- Оповещения администратора о проблеме.

В подавляющем большинстве случаев данная технология требует аппаратной поддержки. Реализация данной технологии опциональна для устройств, поддерживающих Ethernet OAM.

Технология Remote Loopback

Remote Loopback — технология позволяет отправлять команды на замыкание или размыкание петель на удаленном оборудовании, для тестирования и локализации проблемы.

Данная технология начинает работать после успешного завершения обнаружения устройств с включенным OAM (технология Discover). После этого администратор может отправить с вышестоящего оборудования команду на замыканию петли на нижестоящем оборудовании. В этом случае все приходящие на нижестоящее оборудование пакеты, будут отправлены обратно вышестоящему оборудованию, при этом порт будет изолирован от других портов нижестоящего коммутатора.

В основном данная технология используется для:

- Локализации проблем с оборудованием;
- Тестировании производительности;
- Определения задержки на линии передачи данных.

Технология Dying Gasp

Dying Gasp — технология, позволяющая оповестить соседние устройства в случае возникновения фатального сбоя, который приведет к остановке работы устройства. В случае возникновения фатального сбоя соседним устройствам отправляется специальный пакет.

Устройства с питанием от 220 В могут поддерживать быструю отправку пакета Dying Gasp при пропадании питания 220 В.

Случаи, в которых может срабатывать Dying gasp:

- Перезагрузка устройства (по любым причинам);
- Пропадание питания устройства;
- Выключение OAM на порту.

Информация о фатальном сбое может быть использована соседними устройствами для:

- Перестроения дерева Spanning Tree, для исключения сбойного сегмента из топологии;
- Исключения интерфейса из агрегированного LAG интерфейса;
- Оповещения администратора о проблеме.

22.2. Настройка Ethernet OAM на коммутаторах АЛСиТЕК

Общие принципы конфигурирования

Ethernet OAM на коммутаторах АЛСиТЕК позволяет включить службу глобально, а также настроить список интерфейсов, на которых будет работать данная служба.

Поддерживается пассивный режим работы Discover, в которым пакеты отправляются только в ответ на пришедшие OAM пакеты.

Link Fault Monitor включается автоматически, если служба работает на интерфейсе.

Базовая настройка Ethernet OAM

В ходе данной настройки осуществляется включение службы Ethernet OAM на устройстве, а также настройка службы на интерфейсах.

Шаг 1. Включение службы Ethernet OAM на устройстве

Глобально включить службу можно командой:

```
(als_sw) #configure
(als_sw) (configure) #ethernet oam
```

Шаг 2. Настройка интерфейсов

Разрешить работу службы на интерфейсах можно командой:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1-0/8
(als_sw) (interface 0/1-0/8) #ethernet oam
```

Шаг 3. Настройка пассивного режима работы (опционально)

В некоторых случаях требуется, чтобы устройство отвечало на OAM пакеты только при наличии партнера, поддерживающего OAM. Если такого партнера нет, то устройство не посылает в сеть пробные OAM пакеты. Данный режим называется пассивным режимом работы.

Активируем пассивный режим работы. По умолчанию устройство работает в активном режиме.

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1-0/8
(als_sw) (interface 0/1-0/8) #ethernet oam mode passive
```

Просмотр информации OAM Discover

Просмотреть информацию OAM Discover можно с помощью следующей команды:

```
(SW1) #show ethernet oam discover 0/1

Local client:
  Mode..................................... Active
  Unidirection............................. Not supported
  Link monitor............................. Supported
  Remote loopback.......................... Not supported
  MIB retrieval............................ Not supported
  Mtu size................................. 1500
  Port status.............................. Not operational
  Parser action............................ Forward
  Multiplexor action....................... Forward
  PDU revision............................. 0

Remote client:
  MAC address.............................. 00:00:00:00:00:00
  Vendor (oui)............................. 00:00:00
  Mode..................................... Passive
  Unidirection............................. Not supported
  Link monitor............................. Not supported
  Remote loopback.......................... Not supported
  MIB retrieval............................ Not supported
  Mtu size................................. 0
  Parser action............................ Forward
  Multiplexor action....................... Forward
  PDU revision............................. 0
  Dying gasp received...................... No
```

Команда выводит следующую информацию:

- Local client — состояние Ethernet OAM на локальном устройстве;
    - Mode — режим работы обнаружения: "Active" — активный режим, "Passive" — пассивный режим;
    - Unidirection — поддержка технологии оповещение удаленной стороны об однонаправленном линке: "Supported" — поддерживается; "Not supported" — не поддерживается;
    - Link monitor — поддержка технологии Link Fault Monitor: "Supported" — поддерживается; "Not supported" — не поддерживается;
    - Remote loopback — поддержка технологии Remote loopback: "Supported" — поддерживается; "Not supported" — не поддерживается;
    - MIB retrieval — поддержка технологии MIB retrieval: "Supported" — поддерживается; "Not supported" — не поддерживается;
    - Mtu size — максимальный поддерживаемый размер OAM, байты;
    - Port status — статус порта технологии Link Fault Monitor: "Operational" — в работе; "Not operational" — сбой;
    - Parser action — относится к технологии Loopback, определяет что делать с трафиком, приходящим на порт из внешней среды: "Discard" — отбрасывать; "Loopback" — отправлять обратно во внешнюю среду передачи; "Forward" — обычный режим работы;
    - Multiplexor action — относится к технологии Loopback, определяет может ли трафик с других портов передаваться через данный интерфейс: "Discard" — отбрасывать; "Forward" — обычный режим работы;
    - PDU revision — версия пакета, передающего состояние устройства (увеличивается при изменении состояния);

- Remote client — состояние Ethernet OAM на удаленном устройстве;
    - MAC address — MAC-адрес удаленного устройства, если MAC-адрес 00:00:00:00:00:00, то Discover не завершен;
    - Vendor (oui) — сетевой идентификатор производителя устройства;
    - Mode — режим работы обнаружения: "Active" — активный режим, "Passive" — пассивный режим;
    - Unidirection — поддержка технологии оповещение удаленной стороны об однонаправленном линке: "Supported" — поддерживается; "Not supported" — не поддерживается;
    - Link monitor — поддержка технологии Link Fault Monitor: "Supported" — поддерживается; "Not supported" — не поддерживается;
    - Remote loopback — поддержка технологии Remote loopback: "Supported" — поддерживается; "Not supported" — не поддерживается;
    - MIB retrieval — поддержка технологии MIB retrieval: "Supported" — поддерживается; "Not supported" — не поддерживается;
    - Mtu size — максимальный поддерживаемый размер OAM, байты;
    - Parser action — относится к технологии Loopback, определяет что делать с трафиком, приходящим на порт из внешней среды "Discard" — отбрасывать; "Loopback" — отправлять обратно во внешнюю среду передачи; "Forward" — обычный режим работы;
    - Multiplexor action — относится к технологии Loopback, определяет может ли трафик с других портов передаваться через данный интерфейс: "Discard" — отбрасывать; "Forward" — обычный режим работы;
    - PDU revision — версия пакета, передающего состояние устройства (увеличивается при изменении состояния);
    - Dying gasp received — информация о получении Dying Gasp пакета "Yes" — пакет получен; "No" — пакет не был получен.

Активация OAM Loopback

OAM Loopback служит для диагностики неисправностей, активируется по командам администратора.

Шаг 1. Включение службы Ethernet OAM на устройстве

Глобально включить службу можно командой:

```
(als_sw) #configure
(als_sw) (configure) #ethernet oam
```

Шаг 2. Настройка интерфейсов

Разрешить работу службы на интерфейсах можно командой:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (interface 0/1) #ethernet oam
```

Шаг 3. Разрешение замыкания петли на удаленном устройстве

Для того, чтобы loopback тест был возможен, удаленное устройство должно поддерживать функцию замыкания петли, а также данная функция должна быть разрешена на устройстве.

Включаем данную функцию:

```
(als_sw) #configure
(als_sw) (configure) #interface 0/1
(als_sw) (interface 0/1) #ethernet oam remote-loopback
```

Шаг 4. Проведения тестирования

Обращаем ваше внимание, что устройство в пассивном режиме работы по стандарту IEEE 802.3ah не может активировать петлю на удаленном устройстве.

С помощью команды начинаем тестирование порта:

```
(als_sw) #ethernet oam remote-loopback test 0/1
```

Через некоторое время будет показана статистика по переданным и принятым пакетам:

```
(SW1) #ethernet oam remote-loopback test 0/1

  Packets send without errors.................... 1000
  Packets send with errors....................... 0
  Packets receive without errors................. 1000
  Packets receive with errors.................... 0
  Duplicate packets errors....................... 0
  Unexpected packets errors...................... 0

OAM loopback test successfully complete for interface 0/1
```

Команда выводит следующую информацию:

- Packets send without errors — число тестовых пакетов, успешно вышедших с интерфейса;
- Packets send with errors — число тестовых пакетов, не вышедших с интерфейса в результате ошибки;
- Packets receive without errors — число тестовых пакетов, принятых без ошибок;
- Packets receive with errors — число тестовых пакетов, принятых с ошибками;
- Duplicate packets errors — число тестовых пакетов, принятых более одного раза;
- Unexpected packets errors — число пакетов, не относящихся к тестовым.

Шаг 5. Замыкание петли на удаленном устройстве, без проведения тестирования

В некоторых ситуациях требуется замкнуть петлю на удаленном устройстве без отправки тестовых пакетов.

Активируем замыкание петли на удаленном устройстве:

```
(als_sw) #ethernet oam remote-loopback 0/1
```

Контролируем выполнение команды:

```
(SW1) #show ethernet oam discover 0/1

Local client:
  Mode..................................... Active
  Unidirection............................. Not supported
  Link monitor............................. Supported
  Remote loopback.......................... Not supported
  MIB retrieval............................ Not supported
  Mtu size................................. 1500
  Port status.............................. Not operational
  Parser action............................ Discard
  Multiplexor action....................... Forward
  PDU revision............................. 0

Remote client:
  MAC address.............................. 00:13:aa:00:00:01
  Vendor (oui)............................. 00:13:aa
  Mode..................................... Active
  Unidirection............................. Not supported
  Link monitor............................. Supported
  Remote loopback.......................... Supported
  MIB retrieval............................ Not supported
  Mtu size................................. 1500
  Parser action............................ Loopback
  Multiplexor action....................... Discard
  PDU revision............................. 0
  Dying gasp received...................... No
```

- Local client — состояние Ethernet OAM на локальном устройстве;
    - Parser action — поменяло состояние на "Discard";
    - Multiplexor action — поменяло состояние на "Forward";
- Remote client — состояние Ethernet OAM на удаленном устройстве;
    - Parser action — поменяло состояние на "Loopback";
    - Multiplexor action — поменяло состояние на "Discard".

Шаг 6. Размыкание петли на удаленном устройстве (опционально)

Активируем размыкание петли на удаленном устройстве:

```
(als_sw) #no ethernet oam remote-loopback 0/1
```

Контролируем выполнение команды:

```
(SW1) #show ethernet oam discover 0/1

Local client:
  Mode..................................... Active
  Unidirection............................. Not supported
  Link monitor............................. Supported
  Remote loopback.......................... Not supported
  MIB retrieval............................ Not supported
  Mtu size................................. 1500
  Port status.............................. Not operational
  Parser action............................ Forward
  Multiplexor action....................... Forward
  PDU revision............................. 0

Remote client:
  MAC address.............................. 00:13:aa:00:00:01
  Vendor (oui)............................. 00:13:aa
  Mode..................................... Active
  Unidirection............................. Not supported
  Link monitor............................. Supported
  Remote loopback.......................... Supported
  MIB retrieval............................ Not supported
  Mtu size................................. 1500
  Parser action............................ Forward
  Multiplexor action....................... Forward
  PDU revision............................. 0
  Dying gasp received...................... No
```

- Local client — состояние Ethernet OAM на локальном устройстве;
    - Parser action — поменяло состояние на "Forward";
    - Multiplexor action — поменяло состояние на "Forward";
- Remote client — состояние Ethernet OAM на удаленном устройстве;
    - Parser action — поменяло состояние на "Forward";
    - Multiplexor action — поменяло состояние на "Forward".

Компания АЛСиТЕК — ведущий российский разработчик и производитель оборудования для сетей TDM, NGN и IMS. За 24 года работы компанией АЛСиТЕК установлено более 2,5 миллионов портов коммуникационного оборудования. Научный штат компании состоит из 200 высококвалифицированных инженеров, программистов, схемотехников и конструкторов. АЛСиТЕК выпускает полный спектр как станционного xDSL и Ethernet оборудования, так и абонентских устройств.

ООО «Компания «АЛСиТЕК»
410012 Россия, г.Саратов,
ул. Б.Казачья, 8д
Тел: +7 (8452) 79-94-98
Факс: +7 (8452) 79-94-97
alsitec.ru
office@alsitec.ru

2019