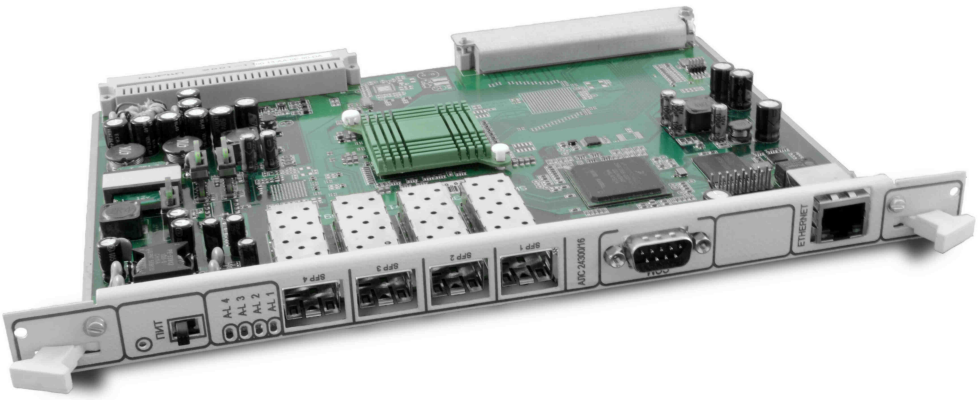
АЛСиТЕК

АЛС-24300

Configuration Manual



ООО «Компания АЛСиТЕК»

ул. Б.Казачья 6

Саратов, 410012

Россия

8-8452-799498

www.alstec.ru

Оглавление

ДАННЫЕ ОБ ОБОРУДОВАНИИ, ПРИВЕДЕННЫЕ В ДАННОМ РУКОВОДСТВЕ, МОГУТ БЫТЬ ИЗМЕНЕНЫ БЕЗ УВЕДОМЛЕНИЯ КЛИЕНТОВ. ВСЯ ИНФОРМАЦИЯ, ФОРМУЛИРОВКИ И РЕКОМЕНДАЦИИ В ЭТОМ РУКОВОДСТВЕ ДОСТОВЕРНЫ, НО ДАЮТСЯ БЕЗ ЯВНЫХ ИЛИ ПОДРАЗУМЕВАЕМЫХ ГАРАНТИЙ. ПОЛЬЗОВАТЕЛИ НЕСУТ ПОЛНУЮ ОТВЕТСТВЕННОСТЬ ЗА ПРИМЕНЕНИЕ ПРОДУКТА.

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ И КОМПЛЕКТУЮЩИЕ ПОСТАВЛЯЮТСЯ ВМЕСТЕ С ПРОДУКТОМ. ЕСЛИ ВЫ НЕ СМОГЛИ НАЙТИ СООТВЕТСТВУЮЩЕЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, ОБРАТИТЕСЬ В НАШУ КОМПАНИЮ ЗА КОПИЕЙ.

ДАННОЕ ОБОРУДОВАНИЕ УСПЕШНО ПРОШЛО ТЕСТИРОВАНИЕ И БЫЛО ПРИЗНАНО РАБОТОСПОСОБНЫМ. КОМПАНИЯ НЕ ГАРАНТИРУЕТ ИСПРАВНУЮ РАБОТУ ОБОРУДОВАНИЯ, В СЛУЧАЕ, ЕСЛИ ОБОРУДОВАНИЕ БЫЛО МОДИФИЦИРОВАНО КЛИЕНТСКОЙ СТОРОНОЙ. ТАКЖЕ, КОРРЕКТНАЯ РАБОТА НЕ ГАРАНТИРУЕТСЯ, ЕСЛИ ОБОРУДОВАНИЕ БЫЛО ПРИОБРЕТЕНО У ТРЕТЬИХ ЛИЦ.

ОРГАНИЗАЦИЯ НЕ НЕСЕТ ОТВЕТСТВЕННОСТИ ЗА ЛЮБОЙ УЩЕРБ, ВКЛЮЧАЯ ПОТЕРЮ ПРИБЫЛИ, ПОВРЕЖДЕНИЕ ИЛИ ПОТЕРЮ ДАННЫХ, КОТОРЫЙ БЫЛ ПОЛУЧЕН ПРИ ИСПОЛЬЗОВАНИИ ПОСТАВЛЕННОГО ОБОРУДОВАНИЯ.

ВСЕ ТОРГОВЫЕ МАРКИ СТОРОННИХ ПРОИЗВОДИТЕЛЕЙ УПОМЯНУТЫЕ В ДАННОМ ДОКУМЕНТЕ ИЛИ НА САЙТЕ КОМПАНИИ ЯВЛЯЮТСЯ СОБСТВЕННОСТЬЮ ИХ ВЛАДЕЛЬЦЕВ.

ЛЮБОЙ IP АДРЕС ИСПОЛЬЗУЕМЫЙ В ДАННОМ ДОКУМЕНТЕ, НЕ ПРЕДНАЗНАЧЕН ДЛЯ ФАКТИЧЕСКОГО ИСПОЛЬЗОВАНИЯ. ЛЮБЫЕ ПРИМЕРЫ, ВЫВОД КОМАНД НА ДИСПЛЕЙ И ЦИФРЫ, УКАЗАННЫЕ В ДОКУМЕНТЕ, ПРИВЕДЕНЫ ТОЛЬКО ДЛЯ НАГЛЯДНОСТИ. ЛЮБОЕ ИСПОЛЬЗОВАНИЕ ФАКТИЧЕСКИХ IP АДРЕСОВ НЕПРЕДНАМЕРЕННО И СЛУЧАЙНО.

ГЛАВА 1

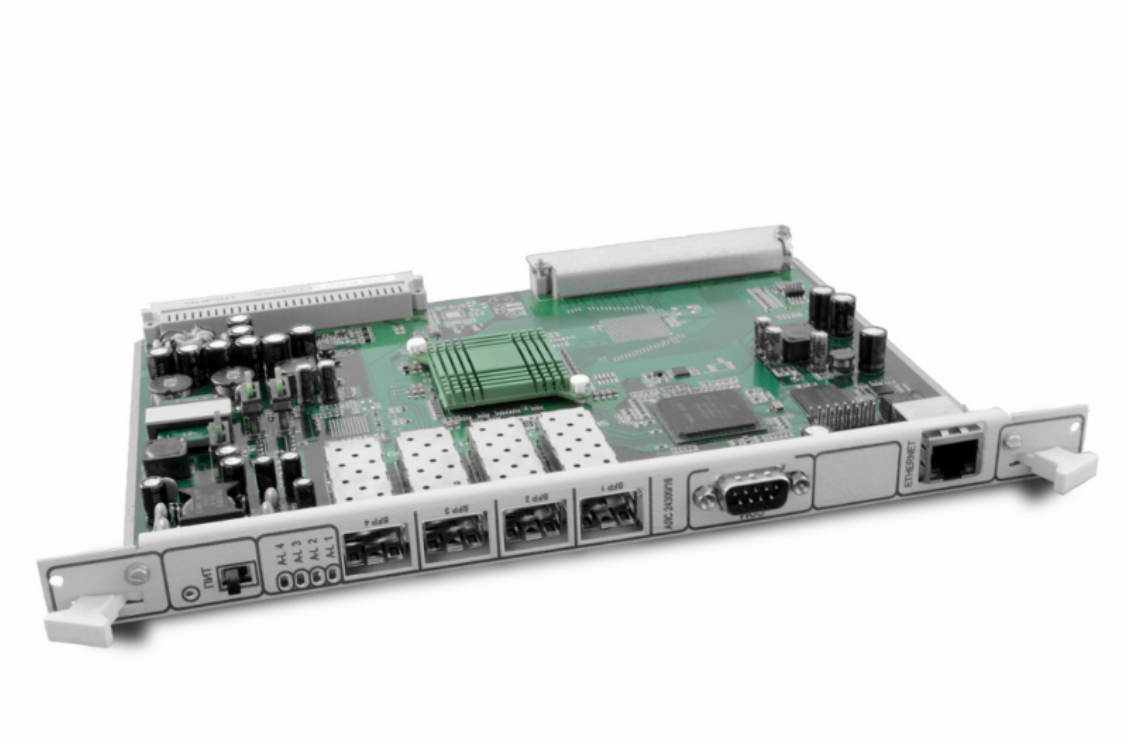
Введение

- Назначение документа
- Область применения

1.1 Назначение документа

Настоящий документ должен позволить сформировать верное и полное представление о возможностях и условиях применения коммутаторов серии АЛС-243000 и АЛС-24300E.

Рисунок 1. АЛС-24300E/16



1.2 Область применения

Коммутаторы АЛС-243000 и АЛС-24300E это программно-аппаратные комплексы, которые применяется на уровне агрегации городских IP-сетей, в университетских сетях, правительственных электронных и крупномасштабных корпоративных сетях и отличаются высокой надежностью, масштабируемостью и широчайшими возможностями предоставления услуг. К данной плате подключаются различные устройства, например, рабочие станции, беспроводные точки доступа, IP - телефоны и другие сетевые устройства, включая серверы, маршрутизаторы, коммутаторы и пр.

Данные коммутаторы могут выступать как в роли отдельных коммутаторов, так и использоваться совместоно с интерфейсными платами SFP-8 и SFP-12 и могут быть установлены в следующие кросс-платы: Cr-BUN4_Eth, Cr-BUN21,Cr-BUN21/V, Cr-BUN21/VA.

Являясь универсальной платформой, коммутаторы АЛС-243000 и АЛС-24300E имеют в своем составе 4 Gigabit Uplink и 16 (24) Gigabit Ethernet SerDes интерфейса, что позволяет им входить в состав шасси в качестве управляющего модуля. Это существенно упрощает схему управления различными линейными картами - коммутаторами (DSL, Ethernet).

ГЛАВА 2

Обзор

2.1 Основные протоколы и поддерживаемые стандарты

Интерфейсы

- АЛС-24300/16 L2
    - 4 «комбо» порта Gigabit Ethernet на лицевой части платы
    - 16 портов Gigabit Ethernet SerDes на тыльной части
    - Консольный порт RS232 на лицевой части платы
- АЛС-24300/24 L2
    - 4 «комбо» порта Gigabit Ethernet на лицевой части платы
    - 24 порта Gigabit Ethernet SerDes на тыльной части
    - Консольный порт RS232 на лицевой части платы
- АЛС-24300E/16 L2, АЛС-24300E/16 L3
    - 4 оптических порта Gigabit Ethernet на лицевой части платы
    - 16 портов Gigabit Ethernet SerDes на тыльной части
    - Консольный порт RS232 на лицевой части платы
    - out-band Ethernet интерфейс
- АЛС-24300E/24 L2, АЛС-24300E/24 L3
    - 4 оптических порта Gigabit Ethernet на лицевой части платы
    - 24 порта Gigabit Ethernet SerDes на тыльной части

- Консольный порт RS232 на лицевой части платы
- out-band Ethernet интерфейс

Поддерживаемые стандарты

- 10Base-T
- 100Base-TX
- 100Base-T4
- 1000Base-TX
- 1000Base-LX
- 1000Base-SX

Опции Layer 2

- таблица MAC-адресов 16K
- Independet VLAN Learning
- Поддержка размера пакетов IEEE 802.3 (64-1522 байта), поддержка jumbo пакетов до 9728 байт
- VLAN 4K VLANs
    - MAC-Based
    - Protocol-Based
    - IP-subnet based VLANs
    - Port-based
- Egress VLAN Trasnlation
- Flexible QinQ (untag->dtag, selective mac, protocol, ip, port, vlan)
- IGMP Snooping 1/2/3
- IGMP Proxy
- Три режима фильтрации Mutlicast пакетов на порту
- Spanning Tree
    - STP
    - RSTP
    - MSTP
- Агрегация портов 802.3ad (до 8 членов в группе)

Опции Layer3 - только для АЛС-24300E/16 L3, АЛС-24300E/24 L3

- Поддержка IPv4 routing
    - Статическая
    - RIP v1/v2
    - OSPF
- IP Multicast
    - IGMP Proxy
    - PIM-SM
    - PIM-DM
    - DVRMP
- Туннелинг IPv4 в IPv6

Управление и контроль

- CLI
- Telnet
- SSH v1/v2
- SNMP v1/v2
- SNMP Trap
- SNTP
- Поддержка sFLOW
- RMON statistics
- WEB интерфейс (опционально)
- Mirroring

Безопасность

- Функция Port Security
- Функция IP-MAC-Port Binding
- Управление широковещательным/многоадресным штормом
- Управление доступом 802.1x на основе портов
- Управление доступом 802.1x на основе MAC-адресов
- RADIUS/TACACS+

Качество обслуживания (QoS)

- Поддержка 802.1p
- Количество очередей на порт: 8
- Методы обработки очередей: Strict Priority, WRR, STWRR
- CoS на основе
    - MAC-адреса
    - Приоритета 802.1p
    - VLAN ID
    - Типа IP-протокола
    - DSCP/IP приоритета
    - IP-адреса
    - Номера порта TCP/UDP
    - Комбинации вышеперечисленного
- Функционал для работы с потоком трафика
    - Перемаркировка 802.1p
    - Перемаркировка приоритетов TOS/ DSCP
    - Ограничение скорости
    - Управление перегрузками трафика
    - Статистика о переданном трафике
    - Управление полосой пропускания для входящего и исходящего трафика: шаг для каждого порта 64 Кб/с
- ACL на основе
    - MAC-адреса
    - Очередей приоритетов 802.1p
    - VLAN ID
    - IP-адреса
    - TOS/DSCP
    - Типа IP-протокола
    - Номера порта TCP/UDP
    - Комбинации вышеперечисленного

ГЛАВА 3

Концепции конфигурирования

- Синтаксис команды
- Командные режимы
- Справочная система
- Доступ к CLI

3.1 Синтаксис команды

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

Пример команды:

```
(als_sw)#network parms <ipaddr> <netmask> [gateway]
```

- work parms — это имя команды
- <ipaaddr> и <netmask> - параметры, котрые должны быть указаны
- [gateway] — это опциональный параметр

Автодополнение делает работу с Command Line Interface (CLI) более удобной и быстрой. Данная функция позваляет дополнить набираемую команду или параметр нажатием на клавишу [Tab], если она может быть однозначно определена, в противном случае ввод команды или параметра можно будет продолжить.

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

Сокращенные формы команд могут применяются во время набора прстой команды. Данная функция позваляет выполнить команду, выполнив ввод нескольких первых символов,

однозначно определяющих данную команду, в противном случае ввод команды или параметра можно будет продолжить. Например

```
(als_sw)#con
(als_sw)(Config)#e
(als_sw)#
```

Некоторые команды могут использоваться совместно со служебным словом no. Такой синтаксис применяется для достижения обратного эффекта по отношению к следующей за ней команде. Например, следующая команда выключает интерфейс

```
(als_sw)(Interface 0/1)#shutdow
```

Отменяя ее действие, команда no shutdown, включает интерфейс

```
(als_sw)(Interface 0/1)#no shutdow
```

3.2 Командные режимы

В интерфейсе командной строки существует несколько различных режимов, группирующих команды по их применению:

- User EXEC
- Privileged EXEC
- Global Config
- VLAN Config
- Interface Config

User EXEC

Включает команды для получения базовой информации о коммутаторе. Приглашение командной строки:

```
(als_sw)>
```

Privileged EXEC

Позволяет выполнять команды по базовой настройке свича и переходить в режим конфигурирования списка обрабатываемых Virtual Local Area Network (VLAN) и в режим глобального конфигурирования. Приглашение командной строки:

```
(als_sw)#
```

Переход в данный режим осуществляется командой

```
(als_sw)>enable
```

Global Config

Группирует основные команды по настройке основных текущих параметров коммутатора. Приглашение командной строки:

```
(als_sw)(Config)#
```

Переход в данный режим осуществляется командой

```
(als_sw)#configure
```

VLAN Config

Группирует команды по настройке обрабатываемых коммутатором VLAN. Приглашение командной строки:

```
(als_sw)(VLAN)#
```

Переход в данный режим осуществляется командой

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

Interface Config

Режим позволяет управлять интерфейсами в режиме коммутации и маршрутизации. Приглашение командной строки:

```
(als_sw)(Interface <u/s/p>)#
```

Переход в данный режим осуществляется командой

```
(als_sw)(Config)#interface <u/s/p>
```

Ряд команд, как например данная, требует указания портов. В таких случаях используется следующий синтаксис <u/s/p>:

- u — указывает идентификатор коммутатора в стеке. Если нет стека устйроств, то unit опускается
- s — слот идентифицирует физическое устройство в наборе или Link aggregation (LAG) или интерфейс для маршрутизации
- p - это либо физический порт либо LAG либо иной логический порт

Из режимов можно выходить с помощью команды exit. Причем из любого режима можно выйти в привилегированный режим (Privileged EXEC) нажав сочетание клавиш <CTRL>-<Z>.

3.3 Справочная система

В интерфейсе командной строки есть возможность дополнения/завершения команды с помощью нажатия клавиш Tab или Пробел.

Справка по тому или иному режиму, доступных конмадах и возможных параметрах вызывается нажатием клаивши <?>.

```
(als_sw)>?
enable Enter into user privilege mode.
help Display help for various special keys.
logout Exit this session.  Any unsaved changes are lost.
password Change an existing user's password.
ping Send ICMP echo packets to a specified IP address.
quit Exit this session.  Any unsaved changes are lost.
show Display Switch Options and Settings.
telnet Telnet to a remote host.

(als_sw)#network parms ?
<ipaddr>
Enter the IP Address.

(als_sw)#show m?
mac mac-addr-table
mldsnooping
mac-address-table
monitor
```

3.4 Доступ к CLI

Telnet

Подключение данным способом является весьма удобным, поскольку нет необходимости присутствовать в непосредственной близости от коммутатора, что необходимо при подключении через последовательный коммуникационный порт. Однако, в ряде случаев доступ к оборудованию при первоначальной настройке может быть затруднен ввиду наличия определенной заводской сетевой конфигурации.

Подключение по протоколу Telnet возможно, если ПК связан с коммутатором непосредственно (при помощи сетевого кабеля UTP категории 5) или или через коммутатор Ethernet, при этом необходимо знать IP-адрес устройства. В противном случае, необходимо воспользоваться последовательным коммуникационным портом.

Подключиться к коммутатору по сети можно с помощью утилиты telnet. Для этого нужно перейти к пункту меню Пуск (Start) - Выполнить (Run). В качестве параметра программе нужно передать IP-адрес устройства, например, для заводской конфигурации:

```
telnet 172.17.1.1
```

После подключения на терминале отобразится диалог входа в систему, где нужно ввести имя пользователя и пароль, для заводской конфигурации: ”admin” и пустой пароль.

После определения IP-адреса устройства необходимо проверить настройки сети на ПК, с которого будет осуществляться конфигурирование. Следует помнить, что связь между рабочей станцией и коммутатором может быть установлена только в том случае, когда они имеют соответствующие IP-адреса из одной подсети. Доступность устройства можно проверить с помощью команды ping. В случае успешного выполнения тестирование настроек IP и доступности коммутатора можно считать успешным.

SSH

Единственным существенным отличаем данного подключения от предыдущего является предоставляемая им безопасность, поскольку SSH шифрует весь поток данных. Спецификация протокола SSH содержится в RFC 4251.

Настройка доступа по SSH осуществляется в 2 этапа

- генерация ключей
- запуск сервера

Генерация ключей

Для генерации ключей необходимо воспользоваться следующими командами.

```
(als_sw)(Config)#crypto key generate dsa
Do you want to overwrite the existing DSA keys? [Y | N] : y

(als_sw)(Config)#crypto key generate rsa
Do you want to overwrite the existing RSA keys? [Y | N] : y
```

Запуск сервера

Для запуска сервера необходимо выполнить следующую команду.

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

Доступ к плате через подключение SSH возможно осуществить при помощи утилиты putty, при этом необходимо ввести значения нескольких параметров.

- адрес хоста: IP-адрес платы
- номер порта: 22
- тип подключения: SSH
- кодировка: UTF-8

После подключения на терминале отобразится диалог входа в систему, где нужно ввести имя пользователя и пароль. После входа в систему отобразится приглашение командной строки.

Serial

Этот способ подключения рекомендуется применять для первичной настройки коммутатора. Для подключения нужно соединить последовательный порт рабочей станции, с которой будет осуществляться конфигурирование, с последовательным портом устройства при помощи консольного кабеля, имеющего соответствующие разъемы на каждом конце.

Начальные установки последовательного порта:

- скорость последовательного порта (Baud Rate): 115200

- биты данных (бит) (Data Bits): 8
- четность (Parity Bits): Нет (None)
- стоповый бит (Stop Bit): 1
- управление потоком (Flow Control): Нет (None)

Далее необходимо сконфигурировать терминал рабочей станции для использования этих установок перед входом в систему коммутатора, при этом можно воспользоваться, например, утилитой putty или HyperTerminal.

Пример настройки утилиты HyperTerminal в Windows:

- запуск программы: [Пуск] - [Программы] - [Стандартные (Accessories)] - [Связь (Communication)] - [Hyper Terminal]
- задание имени соединения: [Имя (Name)] и [Значок (Icon)] в [Описание подключения (Connection Description)]
- выбор порта: в выпадающем списке [Connect To] выбирите порт, через который ПК соединен с коммутатором
- установите указанные выше настройки последовательного порта в диалоге [Свойства COMx (COMx Properties)]
- нажмите кнопку [OK]

Если соединение прошло успешно, на экране терминала отобразится приглашение к вводу имени пользователя и пароля. Имя пользователя по умолчанию - admin, пароля нет. При необходимости пароль можно изменить после входа в систему.

```
User: admin
Password:
(als_sw) #
```

ГЛАВА 4

Базовые операции

- Процесс загрузки
- Образ, загружаемый по умолчанию
- Сетевые параметры
- Управление конфигурациями
- Управление датой и временем

4.1 Процесс загрузки

После включения питания или перезагрузки в коммутаторе происходит загрузка ПО. Во время этого процесса появляется следующее меню:

```
Select startup image
 * 1 - image1
 * 2 - image2
 * 3 - Erase startup-config

default: image1
Select (1, 2, 3):
```

Данное меню позволяет выполнить одно из двух действий:

- удаление сохраненной ранее на коммутаторе конфигурации
- выбор образа ПО для загрузки

На выбор действия дается 2 секунды, при этом, если не был сделан выбор образа, будет загружаться образ, указанный как default.

Удаление сохраненной ранее на коммутаторе конфигурации

Удаление может потребоваться в случае, если коммутатор неправильно сконфигурирован или, например, утерян пароль. Для выполнения данной операции нужно ввести цифру 3 и нажать клавишу ввода.

Выбор образа ПО для загрузки

Выбор может потребоваться в случае, если при обновлении какого-либо образа ПО произошел сбой. Для выбор образа необходимо ввести соответствующую цифру и нажать клавишу ввода.

4.2 Образ, загружаемый по умолчанию

Просмотр информации об доступных образах осуществляется следующей командой.

```
(als_sw)#show bootvar
Image Descriptions

image1 :  default image
image2 :

Images currently available on Flash

----------------------------------------------------------------------
 unit      image1      image2     current-active        next-active
----------------------------------------------------------------------

1    0.3.19.22    0.3.19.30             image2              image2
```

Из вывода команды видно, что и текущим образом, и образом, загружаемым по умолчанию, является image2. Изменение осуществляется командой

```
(als_sw)#boot system image1
Activating image image1 ..

(als_sw)#show bootvar
```

```
Image Descriptions

image1 : default image
image2 :

Images currently available on Flash

--------------------------------------------------------------------
 unit      image1      image2     current-active        next-active
--------------------------------------------------------------------

1    0.3.19.22    0.3.19.30             image2             image1
```

После внесенных изменений образом, загружаемым по умолчанию, стал image1. Смена текущего ядра возможна только посредством перезагрузки, после нее вывод команды show bootvar имеет следующий вид:

```
(als_sw)#show bootvar
Image Descriptions

image1 : default image
image2 :

Images currently available on Flash

--------------------------------------------------------------------
 unit      image1      image2     current-active        next-active
--------------------------------------------------------------------

1    0.3.19.22    0.3.19.30             image1             image1
```

4.3 Сетевые параметры

Конфигурированвие сетевых параметров

Сетевые параметры могут быть сконфигурированиы командой network parms.

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

- <ip address> - IP адрес, обязательный параметр
- <subnet> - маска подсети, обязательный параметр
- [gateway] - шлюз по умолчанию, необязательный параметр

Для данной команды IP-адрес и маска подсети являются обязательными параметрами, в то время как шлюз по умолчанию может не указываться.

4.4 Управление конфигурациями

Текущую конфигурацию (Running Configuration) можно просмотреть выполнив следующую команду.

```
(als_sw)#show running-config
!Current Configuration:
!
!System Description "VDSL2-24 CO - 24 VDSL2, 4 GE/Stack, 0.3.19.30, Linux 2.6.22.1"
!System Software Version "0.3.19.30"
!System Up Time
"0 days 2 hrs 33 mins 0 secs"
!Current SNTP Synchronized Time: Not Synchronized
!
network parms 172.17.1.1 255.255.0.0 0.0.0.0
vlan database
exit
configure
aaa authentication enable "enableList" enable
line console
serial baudrate 115200
```

```
exit
line telnet
exit
line ssh
exit
spanning-tree configuration name ”00-13-AA-06-06-BD”
!
pppoe frmtstr ”$h $v $i”
set igmp mvr 1
set mld mvr 1
exit
```

Вывод команды содержит информацию о системе и все совершенные в ней изменения относительно заводской конфигурации.

Сброс текущей конфигурации

Сброс текущей конфигурации - возврат к заводским настройкам - осуществляется посредством команды clear config.

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

Сохраниение текущей конфигурации

Обычно, изменения, совершенные в конфигурации системы, сохраняют, с тем чтобы при следующем запуске коммутатора его настройки были восстановлены. Подобный функционал предоставляет Startup Configuration. Сохранение текущей конфигурации в качестве Startup Configuration осуществляетсмя командой

```
(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!
```

Резервное копирование

В ряде случаев существует необходимость наличая резервной копии настроек (Backup Configuration). Выполнить такое копирование возможно командой copy.

```
(als_sw)#copy nvram:startup-config nvram:backup-config
```

Резервное восстановление Startup Configuration выполняется следующим образом.

```
(als_sw)#copy nvram:backup-config nvram:startup-config
```

Работа с TFTP сервером

Помимо резервного копирования, существует возможность сохранения и восстановления конфигурация с TFTP сервера. Данные операции выполняются все тойже командой copy.

Копирование Startup Configuration на TFTP сервер осужествимо следующим образом.

```
(als_sw)#copy nvram:startup-config tftp://<ip|host>/<path>/<file>
```

Копирование Startup Configuration с TFTP сервера осуществляется так:

```
(als_sw)#copy tftp://<ip|host>/<path>/<file> nvram:startup-config
```

В обоих командах:

- <ip>|host> - одно из: IP адрес или имя хоста
- <path> - путь к файлу конфигурации на сервере
- <file> - имя файла конфигурации

4.5 Управление датой и временем

Системное время и NTP

Системное время - текущая дата и время. Оно базируется на универсальном глобальном времени (Universal Time Coordinated - UTC).

Network Time Protocol (NTP) — сетевой протокол для синхронизации внутренних часов компьютера с использованием сетей с переменной латентностью. NTP использует для своей работы протокол UDP.

Настройка NTP

Входим в режим конфигурации:

```
(als_sw)#configure
```

Устанавливаем режим односторонней передачи данных:

```
(als_sw)(Config)#sntp client mode unicast
```

Добавляем NTP-сервер, расположенный, например, a.ntp.alphazed.net:

```
(als_sw)(Config)#sntp server a. ntp. alphazed. net
(als_sw)(Config)#sntp unicast client poll- interval 6
```

Настраиваем часовой пояс

```
(als_sw)(Config)#clock timezone +3
```

ГЛАВА 5

Авторизация

5.1 Авторизация по RADIUS

Технология RADIUS

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

RADIUS выполняет три функции:

- Authentication — процесс, позволяющий аутентифицировать (проверить подлинность) субъекта по его идентификационным данным, например, по логину (имя пользователя, номер телефона и т. д.) и паролю
- Authorization — процесс, определяющий полномочия идентифицированного субъекта на доступ к определённым объектам или сервисам
- Accounting — процесс, позволяющий вести сбор сведений (учётных данных) об использованных ресурсах. Первичными данными (то есть, традиционно передаваемых по протоколу RADIUS) являются величины входящего и исходящего трафиков: в байтах/октетах (с недавних пор в гигабайтах). Однако протокол предусматривает передачу данных любого типа, что реализуется посредством VSA (Vendor Specific Attributes)

Возможности RADIUS

Пакеты UDP

Для связи RADIUS между сервером сетевого доступа и сервером защиты используется протокол UDP и порт 1812, официально для этого назначенный. Некоторые реализации RADIUS используют порт 1645. Использование UDP упрощает реализацию как клиента, так и сервера.

Объединение аутентификации и авторизации и выделение аудита

Сервер RADIUS получает запросы пользователя, выполняет аутентификацию и предоставляет клиенту информацию о конфигурации. Аудит выполняется сервером аудита RADIUS.

Шифрование паролей пользователей.

Пароли, содержащиеся в пакетах RADIUS (а это только пользовательские пароли), шифруются посредством хэширования MD5.

Аутентификация PAP и CHAP

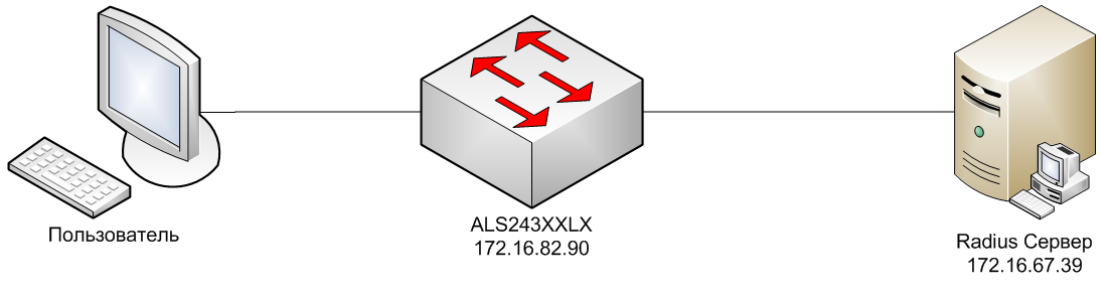
Обеспечивает управление аутентификацией с помощью средств вызова/ответа PAP и CHAP, а также посредством диалога начала сеанса и ввода пароля, аналогично входу в систему UNIX.

Настройка RADIUS

Гостевой режим при доступе по telnet и ssh

Настройка авторизации пользователей коммутатора для гостевого режима при доступе по telnet или SSH на RADIUS сервере. При этом пароль для привилегированного режима (enable) проверяется локально и по умолчанию один для всех пользователей коммутатора. На RADIUS сервере должны быть указаны привилегии (Service-Type) пользователей.

Рисунок 2. Авторизация пользователя на RADIUS сервере




```
(als_sw) #configure
(als_sw) (Config)#radius server ?
attribute                Configure RADIUS Attribute 4 parameters.
host                     Configure RADIUS server host parameters.
key                      Configure the RADIUS key.
msgauth                  Enable/Disable the message authenticator attribute
                         for this server.
primary                  Configure the primary RADIUS server.
retransmit               Configure the retransmit value for the RADIUS server.
timeout                  Configure the RADIUS server timeout value.
```

Указываем сервер RADIUS для данного NAS:

```
(als_sw) (Config)#radius server host auth 172.16.67.39
```

Указываем пароль, заданный для данного NAS в конфигурационном файле clients.conf на сервере RADIUS:

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

Указываем, что пользователь при доступе в гостевой режим должен пройти аутентификацию на сервере RADIUS:

```
(als_sw) (Config)#aaa authentication ?
dot1x                    Create or delete authentication list.
enable                   Create or delete authentication list.
```

```
login                    Create or delete authentication list.
```

```
(als_sw) (Config)#aaa authentication login defaultList ?
enable                   Select enable as the authentication method.
line                     Select line as the authentication method.
local                    Select local as the authentication method.
none                     Select no authentication.
radius                   Select RADIUS as the authentication method.
tacacs                   Select TACACS as the authentication method.
```

```
(als_sw) (Config)#aaa authentication login defaultList radius
```

Указываем, что при доступе по telnet пользователи должны проходить аутентификацию на сервере RADIUS:

```
(als_sw) (Config)#line telnet
(als_sw) (Config-telnet)#login authentication defaultList
(als_sw) (Config-telnet)#exit
```

Указываем, что при доступе по ssh пользователи должны проходить аутентификацию на сервере RADIUS:

```
(als_sw) (Config)#line ssh
(als_sw) (Config-ssh)#login authentication defaultList
(als_sw) (Config-ssh)#exit
(als_sw) (Config)#exit
```

Задаем IP адрес коммутатора

```
(als_sw) #network parms 172.16.82.90 255.255.0.0
```

Проверяем конфигурацию:

```
(als_sw) #show network
Interface Status.............................. Always Up
IP Address.................................... 172.16.82.90
Subnet Mask................................... 255.255.0.0
IP Address.................................... 0.0.0.0
Subnet Mask................................... 0.0.0.0
```

```
Default Gateway.............................. 0.0.0.0
Burned In MAC Address........................ 00:13:AA:0E:60:3D
Locally Administered MAC address............. 00:00:00:00:00:00
MAC Address Type............................. Burned In
Configured IPv4 Protocol..................... None
Management VLAN ID........................... 1
Additional Management VLAN ID................ 4093

(als_sw) #show radius
Number of Configured Authentication Servers.... 1
Number of Configured Accounting Servers........ 0
Number of Named Authentication Server Groups... 1
Number of Named Accounting Server Groups....... 0
Number of Retransmits........................ 4
Timeout Duration............................. 5
RADIUS Accounting Mode....................... Disable
RADIUS Attribute 4 Mode...................... Disable
RADIUS Attribute 4 Value..................... 0.0.0.0

(als_sw) #show radius servers
Cur
rent Host Address             Server Name                      Port  Type
---- ----------------------- ------------------------------- ----- ---------
  *  172.16.67.39            Default-RADIUS-Server            1812  Secondary
```

5.2 Авторизация по TACACS+

Технология TACACS+

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

Улучшена безопасность протокола (шифрование), а также введено разделение функций

аутентификации, авторизации и учёта, которые теперь можно использовать по отдельности.

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

К сожалению, он, несмотря на развитие, перманентно остается в состоянии альфаверсии.

Принцип работы

TACACS+ пользуется транспортным протоколом TCP. Сервер «слушает» порт 49, который является портом протокола IP, выделенным для протокола TACACS. Этот порт зарезервирован для выделенных номеров RFC в протоколах UDP и TCP. Все текущие версии TACACS и расширенные варианты этого протокола используют порт 49.

Протокол TACACS+ работает по технологии клиент/сервер, где клиентом TACACS+, обычно, является NAS. Основополагающим является разделение структурных компонент протокола учета, позволяет обмениваться аутентификационными сообщениями любой длины и содержания и, следовательно, использовать для клиентов TACACS+ любой аутентификационный механизм, в том числе PPP PAP, PPP CHAP, аппаратные карты и Kerberos. Аутентификация не является обязательной, а рассматривается как конфигурируемая опция.

Транзакции между клиентом TACACS+ и сервером TACACS+ идентифицируются с помощью общего «секрета», который никогда не передается по каналам связи. Обычно этот секрет вручную устанавливается на сервере и на клиенте. TACACS+ можно настроить на шифрование всего трафика, который передается между клиентом и сервером TACACS+.

Аутентификация

В ходе аутентификации используются пакеты трех типов: START, CONTINUE и REPLY. START и CONTINUE всегда отправляются клиентом, а REPLY всегда отправляются сервером. Сообщение START описывает тип будущей аутентификации и может содержать имя пользователя и некоторые аутентификационные данные. Пакет START отправляется только в качестве первого сообщения аутентификационной сессии или сразу же после повторного запуска

этой сессии. (Повторный запуск может проводиться по просьбе сервера, которая содержится в пакете REPLY.) Пакет START всегда имеет порядковый номер, равный единице. В ответ на пакет START сервер отправляет пакет REPLY. Сообщение REPLY указывает, завершилась ли аутентификация или ее следует продолжить. Если пакет REPLY требует продолжения аутентификации, он также указывает, какую дополнительную информацию ему нужно предоставить. Клиент собирает эту информацию и отправляет ее серверу в сообщении CONTINUE. По завершении аутентификации клиент может начать процесс авторизации (если она требуется).

Авторизация

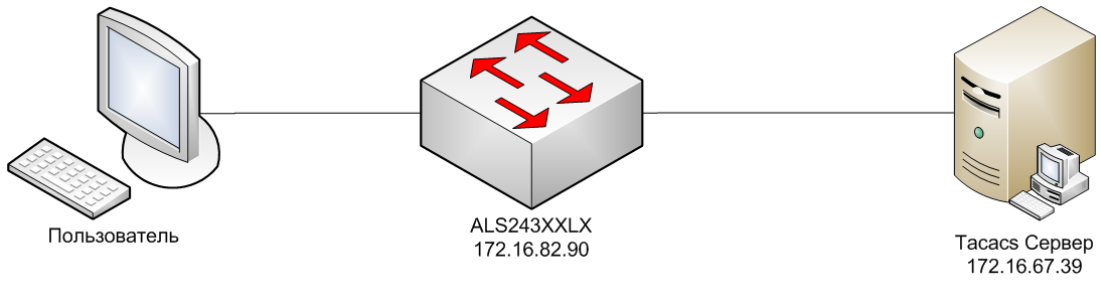
Сессия авторизации состоит из двух сообщений: сообщения REQUEST (запрос) и следующего за ним сообщения RESPONSE (ответ). Сообщение REQUEST содержит фиксированное количество полей, которые описывают пользователя или процесс, и переменный набор аргументов, которые описывают услуги и опции, требующие авторизации.

Аудит

Учет представляет собой запись действий пользователя. В системе TACACS+ учет может выполнять две задачи. Во-первых, он может использоваться для учета использованных услуг (например, для выставления счетов). Во-вторых, его можно использовать в целях безопасности. Для этого TACACS+ поддерживает три типа учетных записей. Записи «старт» указывают, что услуга должна быть запущена. Записи «стоп» говорят о том, что услуга только что окончилась. Записи «обновление» (update) являются промежуточными и указывают на то, что услуга все еще предоставляется. Учетные записи TACACS+ содержат всю информацию, которая используется в ходе авторизации, а также другие данные, такие как время начала и окончания (если это необходимо) и данные об использовании ресурсов.

Настройка TACACS+

Рисунок 3. Авторизация пользователя на Tacacs+ сервере



Гостевой режим при доступе по telnet и ssh

Настройка авторизации пользователей коммутатора для гостевого режима при доступе по telnet или SSH на TACACS+ сервере. При этом пароль для привилегированного режима (enable) проверяется локально и по умолчанию один для всех пользователей коммутатора.

Входим в режим конфигурации:

```
(als_sw) #configure
```

Указываем сервер TACACS+ для данного NAS:

```
(als_sw) (Config)#tacacs-server ?
host                    Configure a TACACS+ server host.
key                     The TACACS+ authentication and encryption key.
timeout                 The connection timeout value in seconds.

(als_sw) (Config)#tacacs-server host ?
<ip-address|hostname>   IP address or Host Name of the TACACS+ server.

(als_sw) (Config)#tacacs-server host 172.16.67.39 ?
<cr>                    Press enter to execute the command.

(als_sw) (Config)#tacacs-server host 172.16.67.39
(als_sw) (Tacacs)#?
exit                    To exit from the mode.
```

```
key                    The TACACS+ authentication and encryption key.
port                   The TACACS+ server port number.
priority               The server priority (0 is the highest priority).
timeout                The connection timeout value in seconds.
```

Указываем пароль, заданный для данного NAS в конфигурационном файле tac_plus.conf на сервере TACACS+:

```
(als_sw) (Tacacs)#key ?
<cr>                   Press enter to execute the command.
<key-string>           Enter a string of length <0 - 128> characters.
encrypted              Enter a pre-encrypted key.

(als_sw) (Tacacs)#key qwerty ?
<cr>                   Press enter to execute the command.

(als_sw) (Tacacs)#key qwerty
(als_sw) (Tacacs)#exit
```

Указываем, что пользователь при доступе в гостевой режим должен пройти аутентификацию на сервере TACACS+:

```
(als_sw) (Config)#aaa authentication ?
dot1x                  Create or delete authentication list.
enable                 Create or delete authentication list.
login                  Create or delete authentication list.

(als_sw) (Config)#aaa authentication login ?
<listname>             Enter a string of not more than 12 characters.
default                The default Authentication List.

(als_sw) (Config)#aaa authentication login defaultList ?
enable                 Select enable as the authentication method.
line                   Select line as the authentication method.
local                  Select local as the authentication method.
```

```
none                    Select no authentication.
radius                  Select RADIUS as the authentication method.
tacacs                  Select TACACS as the authentication method.
```

```
(als_sw) (Config)#aaa authentication login defaultList tacacs
```

Указываем, что при доступе по telnet пользователи должны проходить аутентификацию на сервере TACACS+:

```
(als_sw) (Config)#line telnet
(als_sw) (Config-telnet)#login authentication defaultList
(als_sw) (Config-telnet)#exit
```

Указываем, что при доступе по ssh пользователи должны проходить аутентификацию на сервере TACACS+:

```
(als_sw) (Config)#line ssh
(als_sw) (Config-ssh)#login authentication defaultList
(als_sw) (Config-ssh)#exit
(als_sw) (Config)#exit
```

Задаем IP адрес коммутатора

```
(als_sw) #network parms 172.16.82.90 255.255.0.0
```

Проверяем конфигурацию

```
(als_sw) #show network

Interface Status.............................. Always Up
IP Address.................................... 172.16.82.90
Subnet Mask................................... 255.255.0.0
IP Address.................................... 0.0.0.0
Subnet Mask................................... 0.0.0.0
Default Gateway............................... 0.0.0.0
Burned In MAC Address......................... 00:13:AA:0E:60:3D
Locally Administered MAC address.............. 00:00:00:00:00:00
MAC Address Type.............................. Burned In
Configured IPv4 Protocol...................... None
Management VLAN ID............................ 1
```

```
Additional Management VLAN ID.................. 4093


(als_sw)#show tacacs

Global Timeout: 5

Host address               Port    Timeout   Priority

-------------------------  -----   -------   --------

172.16.67.39               49      Global    0
```

ГЛАВА 6

Управление портами

- Режим конфигурации порта
- Конфигурирование интерфейсов
- Maximum Transmission Unit
- Статус порта

6.1 Режим конфигурации порта

Режим конфигурации порта позволяет управлять интерфейсами в режиме коммутации и маршрутизации. Переход в режим конфигурации порта осуществляется из режима Global Config при помощи следующей команды.

```
(als_sw)(Config)#interface <u/s/p>
```

<u/s/p> - способ указания порта интерфейса, где:

- u – номер устройства в стеке, если коммутатор не в стеке, то поле unit не указывается
- s – слот идентифицирует физическое устройство в наборе или LAG или интерфейс для маршрутизации
- p – номер физического порта

Для некоторых команд возможно задание нескольких интерфейсов в виде диапазона или перечисления, например:

```
(als_sw)(Config)#interface <u/s/p>-<u/s/p>,<u/s/p>
```

Такая возможность доступна для следующих команд:

- auto-negotiate
- set igmp
    - fast-leave

        - mrouter
            - interface
- shutdown
- spanning-tree
    - auto-edge
    - bpdufilter
    - bpduflood
    - edgeport
    - guard
    - mst
    - port
    - tcnguard
- speed
- traffic-shape
- vlan
    - acceptframe
    - ingressfilter
    - participation
    - priority
    - pvid
    - tagging
    - transparent
    - xlate

Команды, не поддерживающие данный режим, будут применены на первом интерфейсе из указанного диапазона/перечисления.

6.2 Конфигурирование интерфейсов

Команды конфигурирования портов позволяют осуществлять:

- просмотр состояния порта
- включение / выключение порта
- управление режимом Auto Negotiation
- управление скоростью

Просмотр состояния порта

По умолчанию все порты включены и настроены в режиме Automatic Negatiation.Состояние одного порта мы можем посмотреть с помощью команды

```
(als_sw)#show port 0/1
              Admin   Physical   Physical   Link   Link    LACP   Actor
Intf   Type    Mode    Mode       Status   Status  Trap    Mode   Timeout
------ ------ ------- ---------- ---------- ------ ------- ------ --------
0/1           Enable  Auto                 Down   Enable  Enable long
```

Состояние всех портов мы можем посмотреть с помощью команды

```
(als_sw)#show port all
              Admin   Physical   Physical   Link   Link    LACP   Actor
Intf   Type    Mode    Mode       Status   Status  Trap    Mode   Timeout
------ ------ ------- ---------- ---------- ------ ------- ------ --------
0/1           Enable  Auto                 Down   Enable  Enable long
0/2           Enable  Auto                 Down   Enable  Enable long
0/3           Enable  Auto                 Down   Enable  Enable long
0/4           Enable  Auto                 Down   Enable  Enable long
0/5           Enable  Auto                 Down   Enable  Enable long
0/6           Enable  Auto                 Down   Enable  Enable long
0/7           Enable  Auto                 Down   Enable  Enable long
0/8           Enable  Auto                 Down   Enable  Enable long
0/9           Enable  Auto                 Down   Enable  Enable long
```

```
0/10          Enable  Auto                 Down   Enable  Enable long
0/11          Enable  Auto                 Down   Enable  Enable long
0/12          Enable  Auto                 Down   Enable  Enable long
0/13          Enable  Auto                 Down   Enable  Enable long
0/14          Enable  Auto                 Down   Enable  Enable long
0/15          Enable  Auto                 Down   Enable  Enable long
0/16          Enable  Auto                 Down   Enable  Enable long
0/17          Enable  Auto                 Down   Enable  Enable long
0/18          Enable  Auto                 Down   Enable  Enable long
0/19          Enable  Auto                 Down   Enable  Enable long
0/20          Enable  Auto                 Down   Enable  Enable long
```

Включение и выключение физического порта

Выключение одного физического порта осуществляется командой

```
(als_sw)(Interface <u/s/p>)#shutdown
```

Выключение всех физических портов осуществляется команды

```
(als_sw)(Config)#shutdown all
```

Включение одного физического порта и всех портов осуществляют теже команды, но с использованием ключевого слова no.

```
(als_sw)(Interface <u/s/p>)#no shutdown
```

и

```
(als_sw)(Config)#no shutdown all
```

Управление параметрами передачи

Auto Negotiation - это процесс автоматического определения параметров передачи, таких как скорость и дуплексность.

Включение режима Auto Negotiation на одном порте осуществляется посредством следующей команды.

```
(als_sw)(Interface <u/s/p>)#auto-negotiate
```

Включение режима Auto Negotiation на всех портах осуществляется посредством следующей команды.

```
(als_sw)(Config)#auto-negotiate all
```

Выключение режима Auto Negotiation на одном и на всех портах выполняют те же команды, но с использованием служебного слова no.

```
(als_sw)(Interface <u/s/p>)#no auto-negotiate
```

и

```
(als_sw)(Config)#no auto-negotiate all
```

Указание скорости и дуплексного режима на одном порте выполняет следующая команда.

```
(als_sw)(Interface <u/s/p>)#speed <10| 100> <half-duplex| full-duplex>
```

Указание скорости и дуплексного режима для всех портов можно осуществить с помощью следующей команды.

```
(als_sw)(Config)#speed all <10| 100> <half-duplex| full-duplex>
```

Данным командам может быть задан только один из двух возможных указанных параметров как для скорости, так и для дуплексности. Например

```
(als_sw)(Interface 0/1)#speed 10 half-duplex
```

6.3 Maximum Transmission Unit

Параметр Maximum Transmission Unit (MTU) определяет максимальный размер блока в байтах, который может быть передан интерфейсом. Задать данный праметр возможно, воспользовавшись следующей командой.

```
(als_sw)(Interface <u/s/p>)#mtu <value>
```

В данной команде <value> - величина параметра, которая может быть задана в пределах от 1518 до 9216. Сброс значения параметра MTU на значение на стандартное значение осуществляется следующей командой.

```
(als_sw)(Interface <u/s/p>)#no mtu
```

6.4 Статус порта

Просмотр информации о принятых / переданных пакетах осуществляется следующей командой.

```
(als_sw)#show interface 0/1
Packets Received Without Error..... 0
Packets Received With Error..... 0
Broadcast Packets Received..... 0
Packets Transmitted Without Errors..... 0
Transmit Packet Errors..... 0
Collision Frames..... 0
Time Since Counters Last Cleared..... 0 day 1 hr 16 min 2 sec
```

Более подробная информация доступна при использовании следующей команды.

```
(als_sw)#show interface ethernet 0/1
Total Packets Received (Octets)..... 0
Packets Received 64 Octets..... 0
Packets Received 65-127 Octets..... 0
Packets Received 128-255 Octets..... 0
Packets Received 256-511 Octets..... 0
Packets Received 512-1023 Octets..... 0
Packets Received 1024-1518 Octets..... 0
Packets Received > 1522 Octets..... 0
Packets RX and TX 64 Octets..... 0
Packets RX and TX 65-127 Octets..... 0
Packets RX and TX 128-255 Octets..... 0
Packets RX and TX 256-511 Octets..... 0
Packets RX and TX 512-1023 Octets..... 0
Packets RX and TX 1024-1518 Octets..... 0
Packets RX and TX 1519-1522 Octets..... 0
Packets RX and TX 1523-2047 Octets..... 0
Packets RX and TX 2048-4095 Octets..... 0
```

```
Packets RX and TX 4096-9216 Octets.............. 0

Total Packets Received Without Errors.......... 0

Unicast Packets Received....................... 0

Multicast Packets Received..................... 0

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 Received Packets Not Forwarded.......... 0

Local Traffic Frames........................... 0

802.3x Pause Frames Received................... 0

Unacceptable Frame Type........................ 0

Multicast Tree Viable Discards................. 0

Reserved Address Discards...................... 0

CFI Discards................................... 0

Upstream Threshold............................. 0

Total Packets Transmitted (Octets)............. 0

Packets Transmitted 64 Octets.................. 0

Packets Transmitted 65-127 Octets.............. 0

Packets Transmitted 128-255 Octets............. 0

Packets Transmitted 256-511 Octets............. 0

Packets Transmitted 512-1023 Octets............ 0

Packets Transmitted 1024-1518 Octets........... 0

Max Frame Size................................. 1518

Total Packets Transmitted Successfully......... 0

Unicast Packets Transmitted.................... 0

Multicast Packets Transmitted.................. 0

Broadcast Packets Transmitted.................. 0

Total Transmit Errors.......................... 0
```

```
FCS Errors..................................... 0
Tx Oversized................................... 0
Underrun Errors................................ 0
Total Transmit Packets Discarded.............. 0
Single Collision Frames........................ 0
Multiple Collision Frames...................... 0
Excessive Collision Frames..................... 0
Port Membership Discards....................... 0
802.3x Pause Frames Transmitted................ 0
GVRP PDUs received............................. 0
GVRP PDUs Transmitted.......................... 0
GVRP Failed Registrations...................... 0
GMRP PDUs Received............................. 0
GMRP PDUs Transmitted.......................... 0
GMRP Failed Registrations...................... 0
STP BPDUs Transmitted.......................... 0
STP BPDUs Received............................. 0
RSTP BPDUs Transmitted......................... 0
RSTP BPDUs Received............................ 0
MSTP BPDUs Transmitted......................... 0
MSTP BPDUs Received............................ 0
EAPOL Frames Transmitted....................... 0
EAPOL Start Frames Received.................... 0
Time Since Counters Last Cleared............... 0 day 1 hr 17 min 28 sec
```

Просмотр статистики по всем портам осуществляется посредством следующей команды.

```
(als_sw)#show interface all
```

Обнулить всю статистику порта возможно следующей командой.

```
(als_sw)#clear counters 0/1
Are you sure you want to clear the port stats? (y/n) *y*

Port Stats Cleared.
```

Обнуление статистики по всем портам осуществляется следующей командой.

```
(als_sw)#clear counters all

Are you sure you want to clear ALL port stats? (y/n) *y*


ALL Port Stats Cleared.
```

ГЛАВА 7

Конфигурирование VLAN

- Что такое VLAN
- Конфигурирование VLAN
- Типовое применение

7.1 Что такое VLAN

VLAN - это технология второго уровня, поддерживаемая большинством современных интеллектуальных коммутаторов, широко используемая для сегментации пользовательского трафика. Процедура тегирования трафика для передачи информации о принадлежности к VLAN описывается открытым стандартом IEEE 802.1Q. Алгоритм работы коммутатора с трафиком VLAN Коммутатору Ethernet, работающему на втором уровне модели OSI и поддерживающему стандарт IEEE 802.1Q постоянно приходится работать с тегированными кадрами. «Внутри» коммутатора не существует кадров, не содержащих тег VLAN. По-умолчанию, т.е. при заводских настройках коммутатора, все порты определены как участники VLAN 1 со снятием на выходе с порта тега VLAN. Ниже представлены этапы обработки кадров при прохождении трафика через коммутатор.

Шаг 1. Filter

При поступлении на порт коммутатора кадра Ethernet, прежде всего проверяется содержит ли кадр тег VLAN. Дальнейшее продвижение кадра зависит от того, включена фильтрация тегированных или не тегированных кадров, или фильтр настроен по умолчанию и пропускает все кадры. Если входящий кадр не соответствует допустимому типу, то он будет отброшен.

Шаг 2. Ingress rule

На этом этапе, устанавливается тег на входящий нетегированный трафик по одному из правил. При этом нужно учитывать, что, если было установлено правило тегирования Port-based, и на

не помеченный кадр вешалась метка 10 VLAN, а потом с помошью соответствующих команд было задано правило MAC based, навешивающее метку 20 VLAN, то в итоге не тегированный кадр получит метку 20 VLAN. Приоритет тегирования идет сверху вниз по списку:

- На основе MAC –адреса (MAC based)
- На основе IP-адреса подсети (IP subnet based)
- На значения поля Ethertype кадра (Protocol based)
- На основе порта (Port-based)

Шаг 3. Ingress filter

Перед тем, как кадр Ethernet будет передан на обработку внутреннему механизму коммутатора, он должен пройти через еще один фильтр. Будет пропущен кадр или нет, зависит от того определено ли на порту, участие в обработке кадров с данным VLAN ID или нет (Про механизм участия портов в обработке тегированных кадров будет сказано в Шаге 4).

Шаг 4. Participation rule

Следующим шагом работы алгоритма является передача кадра Ethernet на обработку внутреннему механизму коммутатора. На данном шаге определяется участие интерфейсов в обработке кадров с определенными значениями VLAN ID. Т.о. устанавливается дальнейший путь кадра внутри коммутатора. Кадр Ethernet будет передан на интерфейсы, которые определены как участники того же VLAN ID, значение которого содержится в поле VLAN ID в кадре. Если ни один из интерфейсов не является участником того же VLAN ID, что определен в поле VLAN ID кадра Ethernet, то кадр будет отброшен.

Шаг 5. Egress rule

Последним шагом алгоритма, является правило для исходящего с интерфейса трафика. Тут определяется для каких значений VLAN ID трафик с интерфейса будет выходить с установленным тегом.

7.2 Конфигурирование VLAN

Настройка VLAN на коммутаторе происходит в несколько этапов:

- VLAN Creation

- Ingress rule
- Participation rule
- Egress rule
- Filter setting

Указание VLAN (VLAN Creation)

Вход в режим VLAN Config

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

Указание списка VLAN которые будут доступны для обработки на всех портах

```
(als_sw)(Vlan)#vlan 10,20,21-25
```

Команда для просмотра созданных VLAN на коммутаторе вводится в режиме Privileged EXEC

```
(als_sw)#show vlan
VLAN ID VLAN Name                         VLAN Type
------- --------------------------------- ---------
1       Default                           Default
10                                        Static
20                                        Static
21                                        Static
22                                        Static
23                                        Static
24                                        Static
25                                        Static
```

Фильтрация (Filter settings)

Фильтрация входящего трафика на одном порте, позваляющая «отбрасывать» все пакеты, относящиеся к VLAN, не включенным в «participation rule» данного порта, настраивается следующим образом

```
(als_sw)#configure
(als_sw)(Config)#interface 0/1
(als_sw)(Interface 0/1)#vlan ingressfilter
```

Фильтрация входящего трафика сразу на всех портах коммутатора задается следующими командами

```
(als_sw)#configure
(als_sw)(Config)#vlan port ingressfilter all
Warning: The use of large numbers of VLANs or interfaces may cause
significant delays in applying the configuration.
```

Фильтрация трафика согласно настройки данного фильтра осуществляется нижеследующим образом

```
(als_sw)#configure
(als_sw)(Config)#interface 0/1
(als_sw)(Interface 0/1)#vlan acceptframe <all/vlanonly/admituntaggedonly>
```

Входящее правило (Ingress rule)

Указывает на интерфейсе правило добавления метки VLAN для входящих untagged пакетов. Существует 4 вида:

MAC based - добавление метки VLAN на основе MAC – адреса

Задаётся глобально на коммутаторе. Метка VLAN навешивается в соответствии с определённым MAC адресом

```
(als_sw)#vlan database
(als_sw)(Vlan)#vlan association mac 00:00:00:01:00:00 10
(als_sw)#show vlan association mac
MAC Address VLAN ID
----------------------- -------
00:0A:00:00:00:01:00:00 10
```

IP based - Добавление метки VLAN на основе IP – адреса

Задаётся глобально на коммутаторе. Метка VLAN навешивается в соответствии с определённым IP адресом

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

```
(als_sw) (Vlan)#vlan association subnet 10.10.101.101 255.0.0.0 10
(als_sw) #show vlan association subnet
IP Subnet          IP Mask            VLAN ID
---------------    ---------------    -------
10.0.0.0           255.0.0.0          10
```

Добавление метки VLAN на основе Ethertype

Добавление метки VLAN на основе Ethertype происходит в несколько этапов.

Сперва, протоколу группы присваивается уникальный адрес, который будет использоваться для определения группы в последующей команде.

```
(als_sw)#configure
(als_sw)(Config)#vlan protocol group test
```

Затем, на основе метки VLAN в соответствующую группу будет добавлен данный протокол.

```
(als_sw)(Config)#vlan protocol group add protocol 1 ?
arp                    Enter a Protocol.
ip                     Enter a Protocol.
ipx                    Enter a Protocol.
pppoe_discovery        Enter a Protocol.
pppoe_session          Enter a Protocol.

(als_sw)(Config)#vlan protocol group add protocol 1 arp
```

Следом, происходит присваение группе нужного VLAN

```
(als_sw) #vlan database
(als_sw) (Vlan)#protocol group 1 20
(als_sw) (Vlan)#exit
```

Наконец, включение участия группы на интерфейсе

```
(als_sw)#configure
(als_sw)(Config)#interface 0/1
(als_sw)(Interface 0/1)#protocol vlan group 1
```

Port Based - Добавление метки VLAN на основе интерфейса

Пример конфигурации

```
(als_sw)#configure
(als_sw)(Config)#interface 0/1
(als_sw)(Interface 0/1)#vlan pvid 20
```

Участие правила (Participation rule)

Указывается участие интерфейса в обработке пакета с определённым VLAN ID. Существует 3 режима участия – include, exclude, auto.

Указание того, что интерфейс будет участвовать в обработке пакетов с данным VLAN ID, задается следующим образом

```
(als_sw)#configure
(als_sw)(Config)#interface 0/1
(als_sw)(Interface 0/1)#vlan participation include 10
```

Указание интерфейу не участвовать в обработке пакетов с данным VLAN ID осуществляется командами

```
(als_sw)#configure
(als_sw)(Config)#interface 0/1
(als_sw)(Interface 0/1)#vlan participation exclude 20
```

Задание участия интерфейса на основе протокола GVRP осуществляется следующими командами, при этом, если данный протокол выключен, то на интерфейсе всегда для пакетов с данной меткой применяется правило exclude

```
(als_sw)#configure
(als_sw)(Config)#interface 0/1
(als_sw)(Interface 0/1)#vlan participation include 10
(als_sw)(Interface 0/1)#vlan participation exclude 20
(als_sw)(Interface 0/1)#vlan participation auto 21
```

Выходящее правило (Egress rule)

Выходящее правило определяет то, как интерфейс будет принимать решение по метке VLAN ID для исходящего пакета - оставлять её или удалять из пакета. Существует 2 типа правил: tagging и untagging. По умолчанию все порты на коммутаторе находятся в режиме untagging.

Указание вставлять метку VLAN ID на всех портах коммутатора осуществляется командами

```
(als_sw)#configure
(als_sw)(Config)#vlan port tagging all 22
Warning: The use of large numbers of VLANs or interfaces may cause
significant delays in applying the configuration.
```

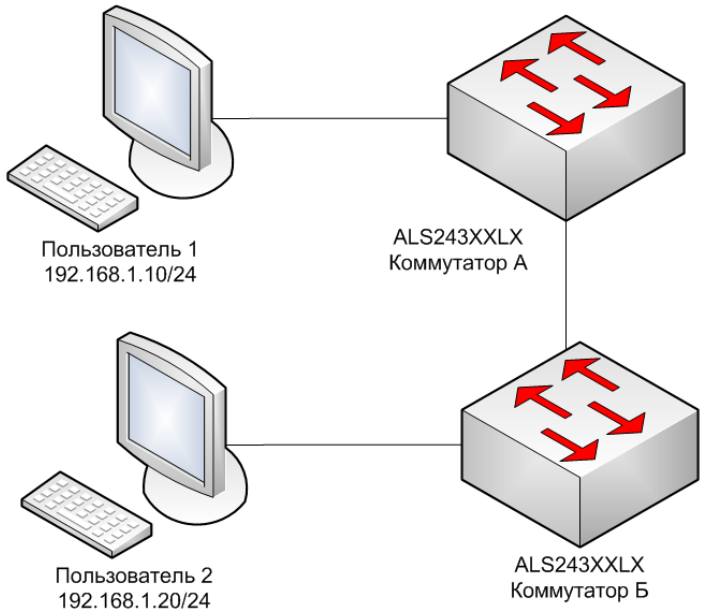
Указание о тегировании пакетов (добавлении метки) для каждого порта отдельно осуществляется следующим образом

```
(als_sw) #configure
(als_sw) (Config)#interface 0/1
(als_sw) (Interface 0/1)#vlan tagging 23
```

7.3 Типовое применение

IEEE 802.1Q trunk

Рисунок 4. Типовая схема сети, в которой настраивается передача тегированного трафика между коммутаторами



Настройка коммуатора А

Создаем VLAN, в котором будут находится хосты.

```
(als_sw)#vlan database
(als_sw)(Vlan)#vlan 100
(als_sw)(Vlan)#exit
(als_sw)#show vlan brief
```

Настраиваем Untagged порты (на схеме только один).

```
(als_sw)#conf
(als_sw)(Config)#interface 0/1
```

Указываем метку VLAN ID, которая будет ставится на входящие от хостов кадры.

```
(als_sw)(Interface 0/1)#vlan pvid 100
```

Даем указание, чтобы интерфейс обрабатывал кадры с заданной меткой. Также, на untagged интерфейсе эта команда устанавливает снятие метки с кадра, при передаче хосту.

```
(als_sw)(Interface 0/1)#vlan participation include 100
```

Настраиваем tagged порты

```
(als_sw)#conf
(als_sw)(Config)#interface 0/2
```

Даем указание интерфейсу пропускать кадры без изменения менки VLAN ID

```
(als_sw)(Interface 0/2)#vlan tagging 100
(als_sw)(Interface 0/2)#vlan participation include 100
```

Настройка коммутатора Б

Создаем VLAN, в котором будут находится хосты.

```
(als_sw)#vlan database
(als_sw)(Vlan)#vlan 100
(als_sw)#show vlan brief
```

Настраиваем Untagged порты

```
(als_sw)#conf
(als_sw)(Config)#interface 0/1
```

Указываем метку VLAN ID, которая будет ставится на входящие от хостов кадры

```
(als_sw)(Interface 0/1)#vlan pvid 100
```

Даем указание, чтобы интерфейс обрабатывал кадры с заданной меткой. Также, на untagged интерфейсе эта команда устанавливает снятие метки с кадра, при передаче хосту.

```
(als_sw) (Interface 0/1)#vlan participation include 100
```

Настраиваем tagged порты

```
(als_sw)#conf
(als_sw)(Config)#interface 0/2
(als_sw)(Interface 0/2)#vlan tagging 100
(als_sw)(Interface 0/2)#vlan participation include 100
```

ГЛАВА 8

Q-in-Q

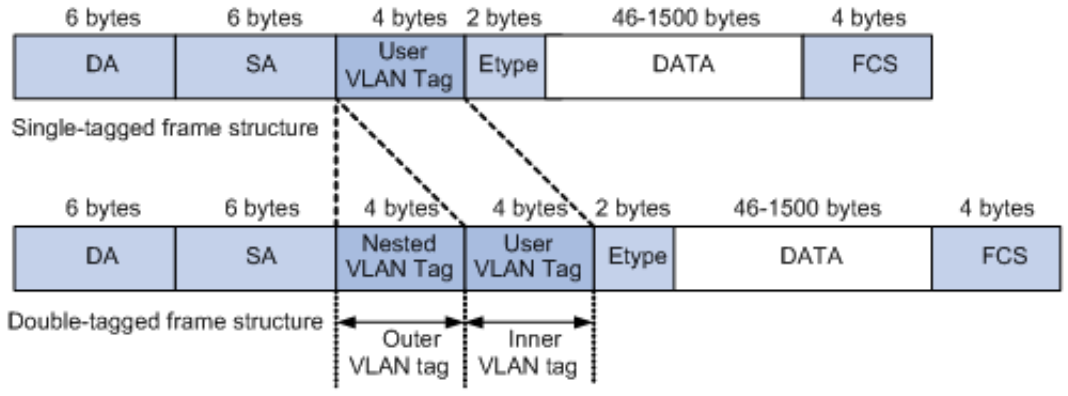
- Что такое Q-in-Q
- Настройка QinQ

8.1 Что такое Q-in-Q

Q-in-Q (Double-Tagging, provider bridging, Stacked VLANs или QinQ) — технология, позволяющая добавлять в Ethernet-фрейм более одного тега VLAN. Описывается стандартом IEEE 802.1ad.

Изначальное назначение техно-логии — проброс трафика из одного подразделения клиента в другое через сеть провайдера с сохранением исход-ных тегов VLAN клиента. Для этого на входящем порту граничного коммутатора про-вайдера ко всему трафику, приходя-щему от клиента, добавляется допол-нительный внешний 802.1Q тег (S-TAG). Данные с внешним тегом проходят через сеть провайдера в другое подразделение клиента, где на выходе внешний тег снимается. При этом разделяется трафик различных клиентов провайдера. Также отсутствует лимит в 4096 VLAN.

Рисунок 5. Структура пакета QinQ



Имеется более сложный вариант (Selective Q-in-Q) предполагающий добавление внешне-го

тега в зависимости от исходного клиентского тега (C-TAG) и использующий механизм VLAN Translation.

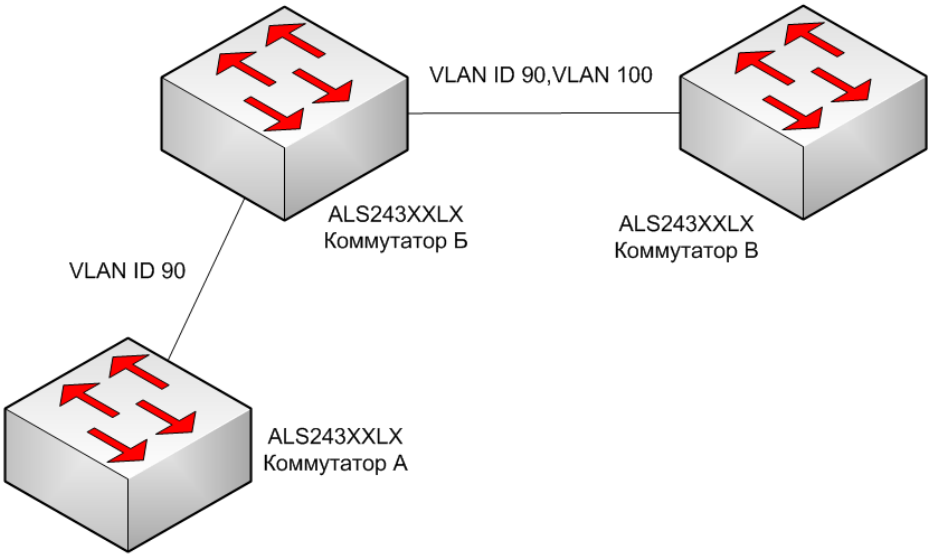
Интерфейсы коммутатора разделены на 2 группы:

- UNI (User-Network Interface) - порт, к которому подключается клиент
- NNI (Network-Network Interface) - порт, смотрящий в сторону ядра сети

Q-in-Q и Selective Q-in-Q настраиваются на UNI.

Базовая архитектура сети при таком использовании Q-in-Q имеет следующий вид:

Рисунок 6. Selective Q-in-Q

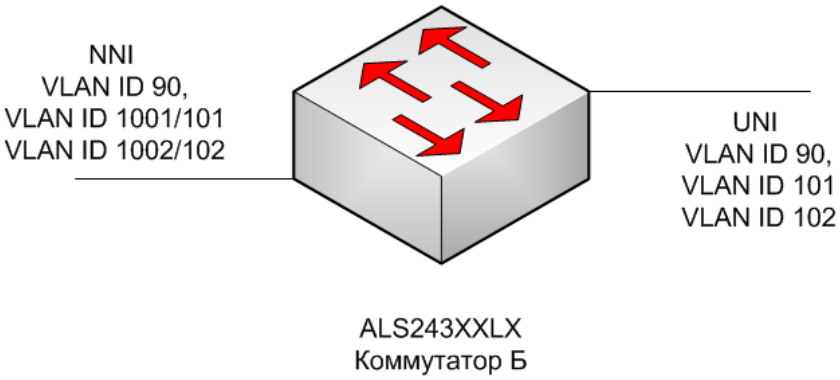


Двойное тегирование может использоваться для реализации идеологий «VLAN на пользователя» и «VLAN на сервис».

8.2 Настройка QinQ

В данном примере необходимо трафик клиента с VLAN ID 101 и 102 пропустить через сеть провайдера с внешними тегами с VLAN ID 1001 и 1002 соответственно, а трафик с VLAN ID 90 пропустить прозрачно, сохранив тег. На обратном пути внешний тег должен сниматься.

Рисунок 7. Схема настройки QinQ



Шаг 1. Создание VLAN на коммутаторе

```
(als_sw)#vlan database
(als_sw)(Vlan)#vlan 90,101,102,1001,1002
(als_sw)(Vlan)#exit
```

Шаг 2. Настройка участия портов в VLAN

В качестве клиентского порта будет использоваться порт 1, провайдерского — 9.

```
(als_sw)#configure
(als_sw)(Config)#interface 0/1
(als_sw)(Interface 0/1)#vlan participation include 90,101,102,1001,1002
(als_sw)(Interface 0/1)#exit
(als_sw)(Config)#interface 0/9
(als_sw)(Interface 0/9)#vlan participation include 90,1001,1002
(als_sw)(Interface 0/9)#exit
```

Шаг 3. Настройка тегирования

На стороне клиента оставляется тег с VLAN ID 90, на стороне провайдера не снимаются теги с VLAN ID 1001 и 1002.

```
(als_sw)#configure
(als_sw)(Config)#interface 0/1
(als_sw)(Interface 0/1)#vlan tagging 90
```

```
(als_sw)(Interface 0/1)#exit
(als_sw)(Config)#interface 0/9
(als_sw)(Interface 0/9)#vlan tagging 1001,1002
(als_sw)(Interface 0/9)#exit
```

Шаг 4. Добавление правил двойного тегирования

К фреймам с VLAN ID 101 и 102 добавляются внешние теги с VLAN ID 1001 и 1002 соответственно.

```
(als_sw)#configure
(als_sw)(Config)#interface 0/1
(als_sw)(Interface 0/1)#vlan xlate add dtag oldvlan 101 newvlan 1001
(als_sw)(Interface 0/1)#vlan xlate add dtag oldvlan 102 newvlan 1002
(als_sw)(Interface 0/1)#exit
```

Шаг 5. Перевод провайдерского порта в режим NNI

```
(als_sw)#configure
(als_sw)(Config)#interface 0/9
(als_sw)(Interface 0/9)#mode dvlan-tunnel
(als_sw)(Interface 0/9)#exit
```

Шаг 6. Включение VLAN Translation

```
(als_sw)#configure
(als_sw)(Config)#vlan xlate
(als_sw)(Config)#exit
```

ГЛАВА 9

Протоколы STP

- STP
- Настройка STP.
- RSTP
- Настройка RSTP
- MSTP
- Настройка MSTP

9.1 STP

Spanning Tree Protocol (STP)— сетевой протокол, работающий на канальном уровне модели OSI. STP предназначен для приведения сети с множественными связями к древовидной топологии, исключающей кольцевые соединения, что приводило бы к широковещательным штормам. 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 — сходимостью. Время сходимости протокола STP — порядка 1 минуты. Протокол STP описан в стандарте IEEE 802.1D.

Ключевые понятия

Bridge Identifier (BID) — идентификатор коммутатора — 8-байтовое число, 6 младших байтов которого составляют его MAC-адрес, а 2 старших байта задают его приоритет, значение по умолчанию которого равно 32768, но может быть изменено, что позволяет влиять на процесс выбора корневого коммутатора.

Port Identifier — идентификатор порта — 2-байтовое число, младший байт которого содержит порядковый номер данного порта в коммутаторе, а значение старшего является приоритетом, его значение по умолчанию равно 128, но может быть изменено.

Bridge Protocol Data Unit (BPDU) — специальные пакеты, которыми обмениваются коммутаторы для автоматического определения конфигурации дерева. BPDU переносят информацию об идентификаторах коммутаторов и портов и о расстоянии до корневого коммутатора. Для пакетов BPDU используется стандартный адрес 01:80:C2:00:00:00. Существует 3 типа сообщений BPDU:

- CBPDU (Configuration BPDU1), используются для построения дерева STP
- TCN (Topology Change Notification) BPDU, используются для оповещения узлов сети об изменении топологии
- TCA (Topology Change Notification Acknowledgment). Коммутатор, который получил TCN BPDU отвечает на него подтверждением. Подтверждение отправляется в следующем CBPDU, которое будет отправлять коммутатор, с помощью вывставления флага TCA

Root Bridge — корневой коммутатор — коммутатор, от которого строится дерево. От корневого коммутатора, до каждого коммутатора в сети выбирается единственный путь.

Root Port —корневой порт — порт некорневого коммутатора, который имеет кратчайшее расстояние до любого из портов корневого коммутатора.

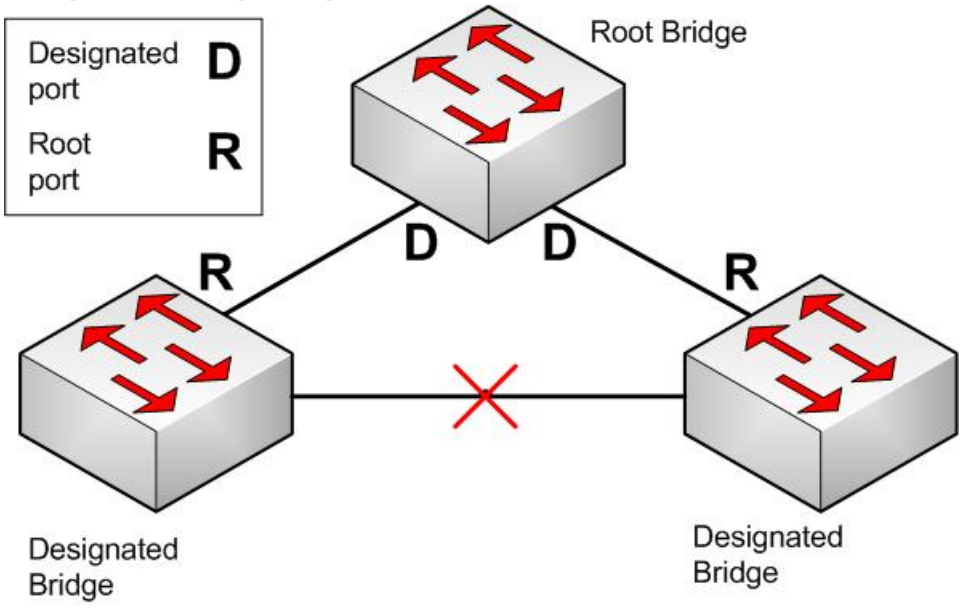
Designated Bridge — назначенный коммутатор для сегмента сети – коммутатор, который принимает пакеты от сегмента и передает их в направлении корневого коммутатора через свой корневой порт, а также принимает пакеты для данного сегмента на корневом порту со стороны корневого коммутатора.

Designated Port — некорневой порт назначенного коммутатора, который связан с сегментом сети. У корневого коммутатора все активные порты являются назначенными.

Root Path Cost — расстояние до корня – суммарное условное время на передачу данных от порта данного коммутатора до порта корневого коммутатора.

Hello time — интервал между генерацией CBPDU, его значение по умолчанию равно 2 секундам, но оно может быть изменено.

Рисунок 8. Пример топологии STP



Состояния портов

Blocking — блокирование — порт только получает и передает BPDU, он не получает и не передает остальные пакеты и не обучается адресам. Следует помнить, что для сохранения блокирования порт должен получать BPDU. Если порт перестанет получать BPDU, то будет инициировано перестроение дерева STP.

Порт в состоянии Blocking выполняет следующие функции:

- отбрасывание кадров, получаемых интерфейсом
- отбрасывание кадров, получаемых от других портов для дальнейшей передачи
- не обучается адресам
- получает BPDU

Listening — прослушивание — порт только получает и передает BPDU, он не получает и не передает остальные пакеты и не обучается адресам. В этом состоянии порты находятся 15 секунд.

Порт в состоянии «listening» выполняет следующие функции:

- отбрасывание кадров, получаемых интерфейсом
- отбрасывание кадров, получаемых от других портов для дальнейшей передачи
- не происходит Learning

- передает и получает BPDU

Learning — обучение — порт только получает и передает BPDU, он не получает и не передает остальные пакеты, но обучается адресам. В этом состоянии порты находятся 15 секунд.

Порт в состоянии «Learning» выполняет следующие функции:

- отбрасывание кадров, получаемых интерфейсом
- отбрасывание кадров, получаемых от других портов для дальнейшей передачи
- происходит Learning
- передает и получает BPDU

Forwarding — передача — порт получает и передает BPDU, и остальные пакеты, обучается адресам. Порт в состоянии «forwarding» выполняет следующие функции:

- передает и продвигает кадры получаемые интерфейсом
- передает кадры коммутируемые от других портов
- происходит Learning
- передает и получает BPDU

Disabled — порт находящийся в выключенном состоянии. Порт в состоянии «disabled» выполняет следующие функции:

- отбрасывание кадров, получаемых интерфейсом
- отбрасывание кадров, получаемых от других портов для дальнейшей передачи
- не происходит Learning
- не передает и не получает BPDU

Алгоритм протокола Spanning Tree

Шаг 1

При включении коммутатора все порты переходят в состояние Listening.

Шаг 2

Выборы корневого коммутатора.

Каждый коммутатор изначально считает себя root bridge. При включении Spanning Tree он формирует и рассылает соседним коммутаторам CBPDU, вставляя в них в качестве

идентификатора root bridge свой BID. CBPDU сравниваются следующим образом: наименьший root ID из CBPDU, имеет наибольший приоритет среди остальных CBPDU. Для CBPDU с тем же root ID, сравнение будет делаться на основании path cost до root bridge. Предположим, что S это сумма root path cost и соответствующего path cost для определенного порта. Меньший размер S, будет означать больший приоритет для CBPDU. Для CBPDU с одинаковыми root ID, и root path cost, сравниваются ID designated port(Bridge ID), designated port ID и ID принимающего порта. Как только коммутатор получает CBPDU, чей BID меньше, чем его собственный, он перестает генерировать свои CBPDU и начинает ретранслировать CBPDU, получаемые от соседей. Этот процесс продолжается до тех пор, пока в сети не останется один коммутатор, генерирующий CBPDU. Он становится корневым.

Шаг 3

Выбор root bridge для каждого коммутатора.

Каждый designated port должен выбрать один порт в качестве корневого. Root port выбирается из портов, на которых принимаются CBPDU root bridge. Для выбора порта коммутатор использует информацию о расстоянии до root bridge из получаемых CBPDU. Каждый коммутатор при ретрансляции CBPDU увеличивает указанное в сообщении расстояние до root bridge. Порт, на котором принимаются CBPDU с наименьшим расстоянием до root bridge, выбирается в качестве root port. Если таких портов оказывается более одного, то из них выбирается порт с наименьшим port ID.

Шаг 4

Выбор designeted bridge и designeted port.

Каждый коммутатор в сети должен выбрать designeted port. Если на порту коммутатора принимаются CBPDU, в которых path cost до root bridge больше, чем в CBPDU, отправляемых с этого порта, то, это означает, что коммутатор является designeted bridge для подсоединенного к конкретному порту сегменту сети, и этот порт становится designeted port. Если path cost до root bridge в принимаемых на порту CBPDU такое же, как и в отправляемых, то порт станет назначенным, если bridge ID, которому принадлежит этот порт, меньше, чем bridge ID, который ретранслирует принимаемые CBPDU. Коммутатор делает все порты, для которых данные условия выполняются, designeted port. Когда имеется несколько портов с одинаковым path cost до root brige, выбирается порт с наименьшим port ID.

Создание конфигурационного BPDU designeted port:

- Формируется path cost. Суммируется path cost содержащийся в CBPDU, который принял root port и его собственный path cost
- В качестве ID designated bridge выступает ID данного коммутатора
- В качестве ID designated port используется ID порта, который будет передавать BPDU пакеты

Шаг 5

Root и designeted порты переходят в состояние learning. При этом они могут изменить свою роль, если будут получены соответствующие BPDU. Все остальные порты, кроме root и designated, переходят в состояние blocking.

Шаг 6

Root и designeted порты переходят в состояние forwarding.

Шаг 7

Если в течении 10 интервалов Hello (по умолчанию; значение может быть изменено) root port коммутатора не получает CBPDU, то он формирует и рассылает TCN BPDU, при получении которых каждый коммутатор должен очистить свою таблицу коммутации, и начать рассылать CBPDU, указывая в качестве root bridge ID, свой BID, тем самым инициализируя новую процедуру выбора root bridge.

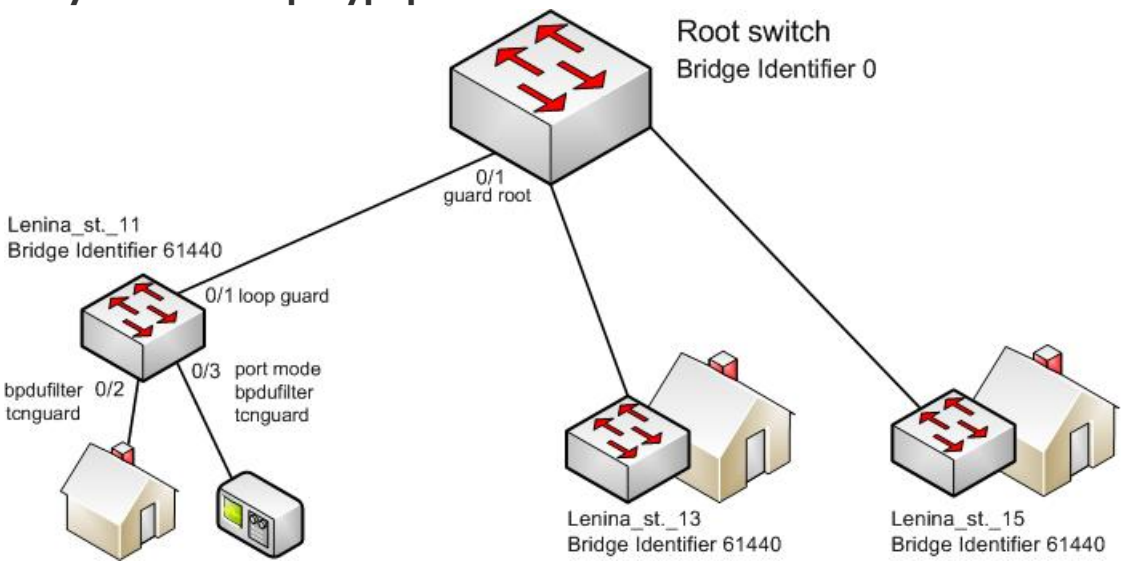
Шаг 8

Алгоритм повторяется с первого шага.

9.2 Настройка STP.

В примере настройки STP приведенном на рисунке ниже, показано применение протокола для распределения трафика от магистрального коммутатора до коммутаторов отделов.

Рисунок 9. Конфигурирование STP



Конфигурирование корневого коммутатора

Назначение коммутатору имени.

```
(als_sw)#hostname Root
```

Переход в режим Global Config

```
(Root) #configure
```

spanning-tree

Применение данной комманды включает механизм STP

```
(Root)(Config)#spanning-tree
```

Назначение коммутатору наиболее низкого приоритета.

```
(Root)(Config)#spanning-tree mst priority 0 0
```

Переходим в режим Interface Config (интерфейс 0/1).

```
(Root)(Config)#interface 0/1
```

Команда позволяет порту принимать и передавать BPDU пакеты, вне зависимости от того включен, или выключен механизм STP на коммутаторе глобально.

```
(Root)(Interface 0/1)#spanning-tree port mode
```

Комманда spanning-tree port mode может быть выполнена из режима Interface Config, для каждого отдельного порта или из режима Global Config для всех портов коммутатора сразу. Данная настройка должна быть применена на всех портах участвующих в STP.

Включение данной функции позволит защитить вышестоящую сеть от атак с применением CBPDU с приоритетом 0.

```
(Root)(Interface 0/1)#spanning-tree guard root
```

Настройки spanning-tree port mode и spanning-tree guard root должны быть применены для всех назначенных портов

Конфигурирование коммутатора уровня доступа

В данном примере настройки коммутаторов «department 1» и «department 2» идентичны друг - другу.

```
(als_sw)#hostname department 1
(department 1)#configure
(department 1)(Config)#spanning-tree
(department 1)(Config)#spanning-tree mst priority 0 61440
```

Порты Designated switch могут быть настроены следюущим образом:

```
(department 1)(Interface 0/1)#interface 0/1
(department 1)(Interface 0/1)#spanning-tree port mode
```

Включение данной функции позволит защитить коммутатор от TCN BPDU.

```
(department 1)(Interface 0/1)#spanning-tree tcnguard
(department 1)(Interface 0/1)#spanning-tree guard root
```

9.3 RSTP

Rapid Spanning Tree Protocol (RSTP; IEEE 802.1w) — протокол быстрого построения древовидной топологии. Построение топлогии STP может занять от 30 до 50 секунд, тогда, как RSTP может изменить топологию в течении трех периодов Hello times (по умолчанию: 6 секунд). RSTP является развитием стандарта 802.1D. RSTP в целях взаимодействия с устаревшим оборудованием, на отдельных портах, может работать как STP; однако, при этом теряются преимущества, которыми обладает новый протокол. В RSTP, для сокращения времени построения активной топологии используется несколько новых механизмов.

Отличия RSTP от STP

Тип соединения

В RSTP учитывается тип сегмента подключенного к порту. Различаются следующие типы сегментов:

- point-to-point (точка — точка): в коммутируемых сетях — единственный вид сегмента, для него у порта существует единственный сосед
- разделяемая среда: стандарт RSTP по-прежнему учитывает существование разделяемой среды, и все стандарты, включая основной стандарт Ethernet IEEE 802.3, описывают работу этого типа, включает устройства физического уровня (повторители/ концетраторы)
- edge port (тупиковая связь): порт коммутатора соединен с конечным узлом сети - по этому сегменту нет смысла ждать прихода BPDU, конфигурируется администратором

Таймеры

Сокращен период «max — edge» — по умолчанию он составляет 3 периода вместо 10, т.е., если в течении 6 секунд (вместо 20) коммутатор не получит CBPDU, то он инициирует перестроение дерева RSTP.

Состояния портов

В протоколе RSTP есть только три состояния порта:

- discarding (отбрасываение) — состояния портов STP - blocking, listening, disabled были объеденены в одно, порт в данном состоянии выполняет следующие функции
    - отбрасывание кадров, получаемых интерфейсом
    - отбрасывание кадров, получаемых от других портов - состояние forwarding не активно
    - нет процесса learning
    - передает и получает BPDU
- learning
- forwarding

Роли портов

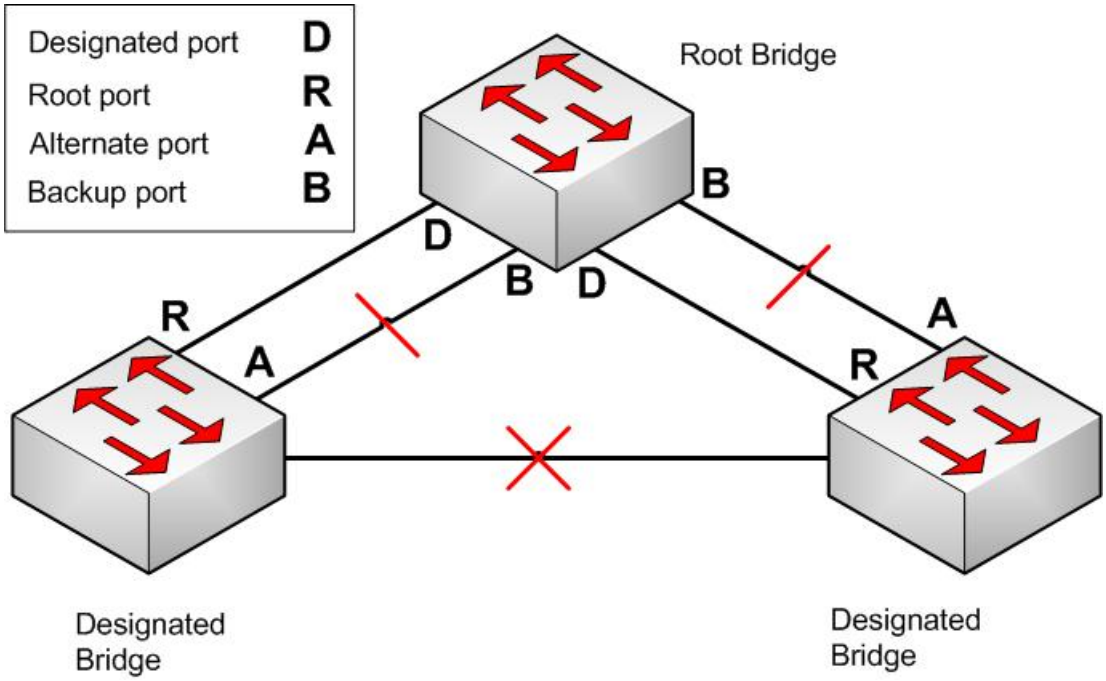
Введены новые роли портов:

- alternate (альтернативный) — дублер root port - переходит в состояние forwarding, когда root port перестает работать, или не получает сообщения BPDU в течении установленного промежутка времени
- backup (резервный) — дублер designeted сегмента

Как alternate, так и backup порты выбираются одновременно с root и designeted портами, что значительно ускоряет реакцию сети на отказы. Например, переход на alternate port происходит сразу же после фиксации отказа и не связан с ожиданием окончания таймеров RSTP.

На рисунке ниже показан пример топологии RSTP; избыточное соединение между Alternate и Backup портами заблокировано.

Рисунок 10. Пример топологии RSTP



Перестроение активной топологии

Введена процедура подтверждения перехода designated порта в состояние forwarding после изменения активной топологии. Безусловный переход для designeted port, который до этого не продвигал кадры из — за того, что порт — сосед в связи «point-to-point» находился в

состоянии discarding, может вызвать образование петель. Для исключения такой ситуации, designated порт запрашивает подтверждение своей роли у соседних коммутаторов. Для этого порт посылает своему соседу по сегменту «point-to-point» BPDU пакет, называемое предложением (proposing). Это сообщение вызывает временный перевод всех назначенных портов соседа в состояние discarding; кроме того, эти порты распространяют собщение с предложеним далее по сети своим нижележащим (в отношении расстояния до root bridge) соседям. Когда это сообщение доходит до коммутатора, у которого все порты либо находятся в стабильном состоянии, либо подключены к конечным узлам сети, то этот коммутатор отвечает на него сообщением согласия. Designeted port, получив в ответ сообщение согласия (оно проходит в обратном направлении, «от листьев к корню»), фиксирует состояние forwarding. Если designated port не получает такого сообщения, то он остается в состоянии discarding. Данная процедура исключает возникновение петель в активной топологии, к тому же она сокращает время работы протокола RSTP, так как именно благодаря ей исключается стадия listening.

Изменения в BPDU

Коммутатор устанавливает флаг proposal (процедура описана выше) в RSTP BPDU для того чтобы предложить себя на роль designeted bridge в сегменте. Роль порта в proposal-сообщении всегда - designated. Коммутатор устанавливает флаг agreement в RSTP BPDU для того чтобы принять предыдущее предложение. Роль порта в agreement-сообщении всегда - root.

В RSTP нет отдельного BPDU для анонсирования изменений в топологии (topology change notification (TCN)). Протокол использует флаг topology change (TC) для того чтобы указать на изменения. Однако, для совместимости с коммутаторами, которые используют 802.1d, коммутаторы использующие RSTP обрабатывают и генерируют TCN BPDU.

Обработка нового BPDU

BPDU отправляются с интервалом, равным hello - timeв, а не просто ретранслируются. Согласно стандарту 802.1d, non root bridge формирует BPDU только в случае их приема через root port. Фактически коммутатор больше ретранслирует BPDU, чем действительно генерирует их. Однако в 802.1w это не всегда так. Теперь коммутатор отправляет BPDU с текущей информацией каждые несколько секунд (hello - time по умолчнанию равно 2 секундам), даже если он не получил их от корневого коммутатора.

Сокращение времени max - age - более быстрое устаревание информации о активной топологии

Если порт не получает CBPDU три раза подряд, информация может немедленно считаться устаревшей (либо по истечению таймера max — age). Из-за ранее упомянутых изменений протокола, BPDU теперь используются как механизм проверки активности соединения между узлами RSTP. При утере трех пакетов CBPDU подряд устройство считает, что связь, со своим прямым ближайшим корневым или назначенным мостом, утрачена. Сокращение времени max - age позволяет быстро обнаруживать ошибки. Если коммутатор не принимает BPDU от соседнего коммутатора, можно с уверенностью говорить, что соединение с этим узлом spanning - tree потеряно. Это одно из отличий от протокола 802.1d, в котором проблема могла крыться в любом месте активной топологии пути spanning - tree. Сбои физического соединения в RSTP обнаруживаются намного быстрее.

Прием CBPDU

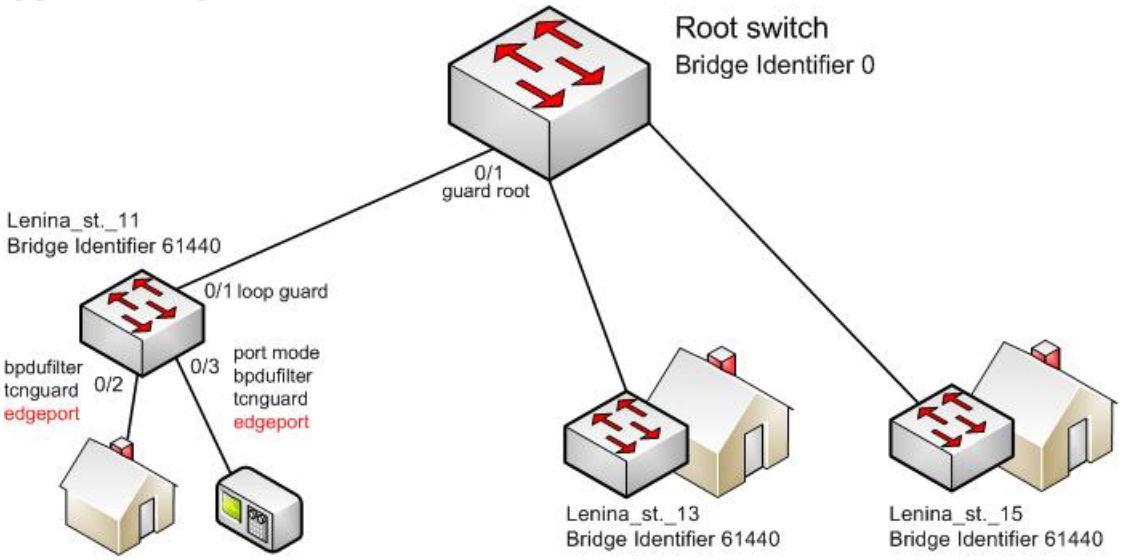
Этот принцип является частью ядра механизма Backbone Fast. Комиссия IEEE 802.1w приняла решение использовать схожий механизм в RSTP. Когда коммутатор получает информацию от designated или root bridge, о добавлении к существующей топологии нового устройства, или сегмента сети, он сразу принимает ее, заменяя информацию, полученную ранее.

9.4 Настройка RSTP

Протокол RSTP имеет обратную сходимость с протоколом STP, поэтому все настройки приведенные для STP справедливы и для RSTP. Отличие заключается только в явном указании режима работы RSTP (настройки приведены ниже) и функции RSTP «port fast».

На рисунке ниже показан пример настройки топологии RSTP с использованием функции «port fast».

Рисунок 11. Пример построения сети с использованием RSTP функции «port fast»



Активация протокола RSTP производится в режиме Global Config, при помощи следующей команды:

```
(als_sw)(Config)#spanning-tree forceversion 802.1w
```

Настройка функции RSTP «port fast», может производиться двумя способами.

Команда переводит порт в режим auto-edge. При этом подразумевается, что порт будет подключен к конечным узлам. Порт должен автоматически переходить в состояние «forwarding» без задержки, как только устройство будет подключено.

```
(als_sw)(Interface 0/2)#spanning-tree auto-edge
```

Команда позволяет явно указать, что порт подключен к конечному устройству. Применение команды позволяет перейти порту в состояние «forwarding» без задержки.

```
(als_sw)(Interface 0/2)#spanning-tree edgeport
```

9.5 MSTP

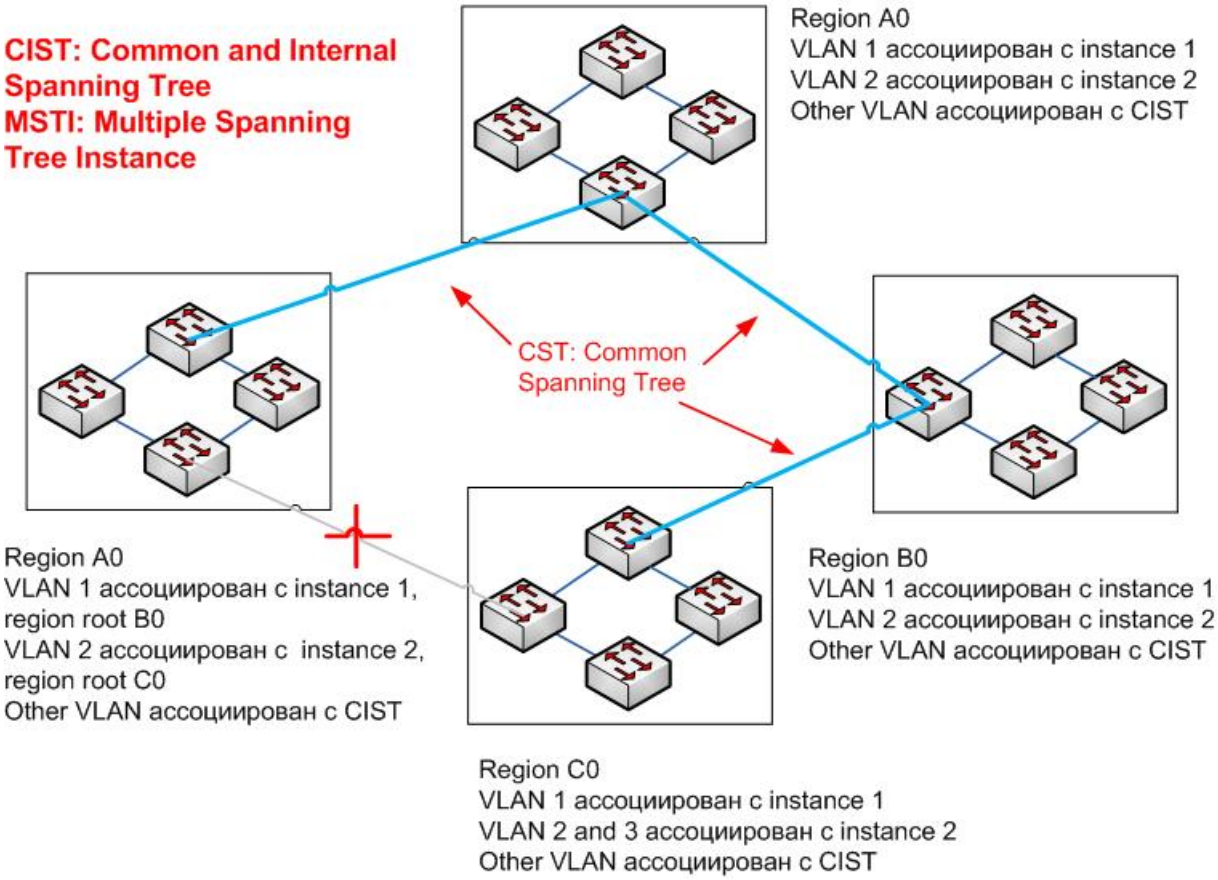
Multiple Spanning Tree Protocol (MSTP; 802.1s) — протокол множественного свзующего дерева. MSTP расширяет стандарт IEEE 802.1w (RSTP) для поддержки нескольких копий древовидной топологии сети, кроме того, протокол MSTP обратно совместим с протоколом IEEE 802.1d STP. Коммутаторы с поддержкой MSTP могут использовать для построения древовидной топологии

BPDU пакеты STP и RSTP, вследствии того, что MSTP обратно совместим с RSTP и MSTP. Протокол MSTP обеспечивает быструю сходимость сети, и вносит дополнения и в стандарт 802.1Q. MSTP позволяет использовать более одной копии spanning tree в сети с 802.1q VLAN. Он позволяет одни VLAN связать с одним instance spanning tree, а другие с другим, обеспечивая несколько связей между коммутаторами. В отличие от RSTP, MSTP строит топологию с учетом настроенных на коммутаторах VLAN.

Архитектура MSTP обеспечивает несколько разных вариантов для передачи данных и позволяет организовать баланс нагрузки. Это свойство позволяет усилить отказоустойчивость сети, т.к. сбой соединений в отдельном «дереве» (маршруте передачи данных) не отразится на других «деревьях» и, соответственно, возможных маршрутах. Также благодаря MSTP облегчается задача администрирования и управления крупными сетями: можно использовать резервные маршруты передачи данных путем настройки нескольких VLAN и настройкой независимых топологий на получившихся сегментах сети. В сетевой топологии построенной с помощью MSTP предупреждение петель происходит не путем отключения порта коммутатора, а с помощью отключения передачи данных для определенной VLAN. MSTP можно настроить таким образом, чтобы для части VLAN было заблокировано одно соединение из дублированных связей, а для других VLAN — другое. Таким образом, все соединения будут передавать данные. В случае повреждения канала связи MSTP обнаружит это и автоматически перестроит структуру.

Главный недостаток решения на протоколе MSTP заключается в сложности настройки такой структуры. Администратор должен четко представлять разбиение системы на VLAN, оценить потоки данных в каждом сегменте сети и путем ручной настройки добиться относительно равномерного использования всех каналов связи. Также следует учитывать, что во многих моделях коммутаторов (особенно бюджетного ряда) поддержка протокола MSTP не реализована.

Рисунок 12. Схематичное отображение механизма MSTP



Ключевые понятия

MST region - включает в себя несколько физически взаимосвязанных поддерживающих MSTP коммутаторов и оответствующих сегментов сети, подключенных к этим коммутаторам. Эти коммутаторы имеют несколько одинаковых параметров:

- конфигурационное имя
- конфигурационный номер ревизии
- VLAN mapping table - таблица, где указано какой VLAN ассоциирован с определенным MSTI

Коммутируемая сеть - может содержать несколько MST region. Можно группировать коммутаторы внутри одного MST region используя соответствующие конфигурационные команды. Например, все коммутаторы региона A0(см. Рис.5.) имеют общую конфигурацию региона:

- конфигурационные имя и номер ревизии (на рисунке не показано)

- VLAN mapping table - таблица ассоциирования VLAN и instance STP
- одинаковые номера ревизии

Multiple spanning tree instance (MSTI). MSTI входт в состав MST region. В MST region может быть создано несколько MSTI. Эти топологии не зависят друг от друга. Например на рисунке, каждый MST region содержит несколько MSTI. Каждая из этих топологий соответствует определенному VLAN.

VLAN mapping table — таблица, где указано какой VLAN ассоциирован с определенным MSTI. Например на рисунке, содержится информация в VLAN mapping table о регионе A0: VLAN 1 ассоциирован с MSTI 1; VLAN 2 ассоциирован с MSTI 2; а остальные VLAN ассоциированы с CIST. В MST балансировка нагрузки достигается с помощью VLAN mapping table.

Internal spanning tree (IST) — внутренняя древовидная топология — древовидная топология содержащаяся внутри MST region. IST вместе с общей древовидной топологией (common spanning tree — CST) формирует внутреннию и общую древовидную топологию (common and internal spanning tree — CIST) всей коммутируемой сети. IST является специальной MSTI; она принадлежит к MST региону и является частью CIST. На рисунке Каждый MST регион имеет IST, который является частью CIST.

Common spanning tree (CST) — общая древовидная топология. CST — это древовидная топология в коммутируемой сети, которая соединяет все MST region в сети. Если рассматривать каждый MST region в сети, как коммутатор, тогда древовидная топология CST образована с помощью STP или RSTP «коммутаторов». На рисунке CST выделена красной линией.

Common and internal spanning tree (CIST) — древовидная топология в коммутируемой сети которая объединяет все коммутаторы. CIST является чем — то средним между IST и CST. На рисунке все IST в MST region и CST соединения формируют CIST.

Region root — это root IST или MSTI в MST регионе. Разные древовидные топологии в MST region могут иметь разные топологии и разные root region. В region D0 на рисунке, корнем region MSTI 1 является коммутатор B, а корнем region MSTI 2 - коммутатор C.

Common root bridge — общий корневой мост - является корнем CIST. Common root bridge показан на рисунке, им является коммутатор region A0.

Region edge port — граничный порт региона, находится на границе MST region и использует

связь между собственным MST region и другим MST region, где топология организована на основе STP или RSTP.

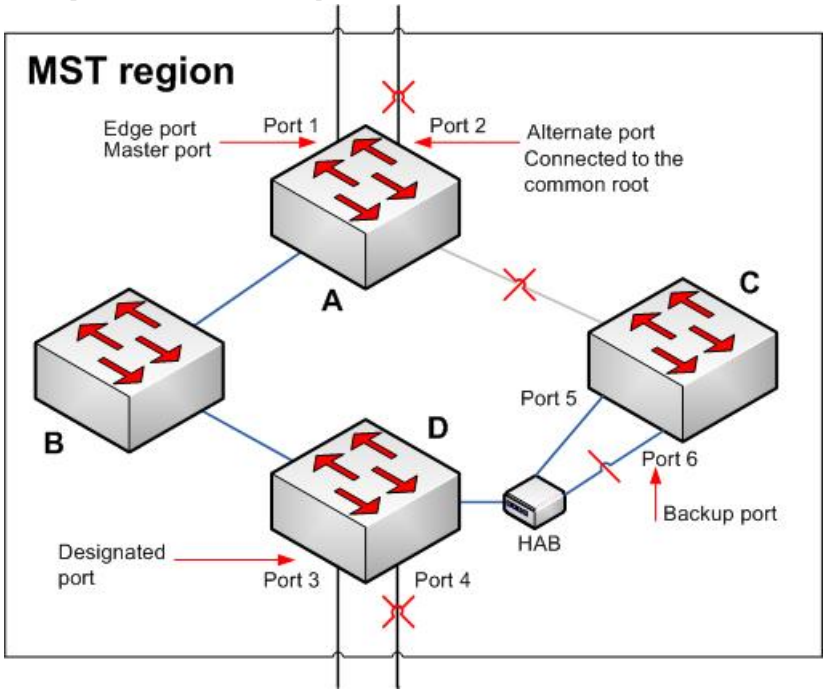
Роли портов

В MSTP определены следующие роли портов:

- root port — корневой порт - используется для продвижения пакетов к корню древовидной топологии
- designated port — назначенный порт - используется для продвижения пакетов в направлении от root bridge
- master port — мастер порт - соединяет MST region и общий порт, путь от master port к общему порту - наиболее короткий путь между MST регионом и общим корнем
- region edge port — граничный порт региона
- alternate port — альтернативный порт - может заменять master port или root port, т.е. работает являясь заменой master port при отключении последнего
- backup port — запасной порт - блокируется коммутатором, как один из двух связанных портов, образовавших петлю

На рисунке ниже коммутаторы A, B, C, и D формируют MST region. Порт 1 и 2 на коммутаторе A подключены общему корню. Порт 5 и порт 6 на коммутаторе C формируют петлю. Порт 3 и порт 4 на коммутаторе D связаны в направлении downstream с остальными MST регионами. Этот рисунок показывает роли портов в действии.

Рисунок 13. MST регион



Порты могут играть различные роли в различных MSTI. Роль region edge port может быть совмещена с ролью которая есть у порта в CIST. Например, порт 1 коммутатора A на рисунке имеет роль region edge port, и он же является master port в CIST. Таким образом это master port во всех MSTI в регионе.

Состояния портов

В протоколе MSTP есть только три состояния порта:

- discarding
- learning
- forwarding

Реализация MSTP

MSTP делит сеть между несколькими MST region на канальном уровне модели OSI. CST созданная между этими MST регионами, и несколькими spanning - tree (или MSTI), может быть создана в каждом MST region. Как и RSTP, MSTP использует CBPDU. Различие только в том, что CBPDU MSTP переносят MSTP конфигурационную информацию между коммутаторами.

Генерация CIST

При сравнении CBPDU, коммутатор обладающий наибольшим приоритетом в сети будет выбран root bridge для всего CIST. В каждом MST region, IST будет сформирован согласно протокола MST. В то же время, в общей сети для MSTP, MST region выглядят, как логические коммутаторы. CST совместно, с IST формирует CIST сети.

Генерация MSTI

В MST region, отдельные MSTI формируют для различных VLAN зависимости для внесения во VLAN mapping table. Каждая топология рассматривается независимо, так же как STP/RSTP.

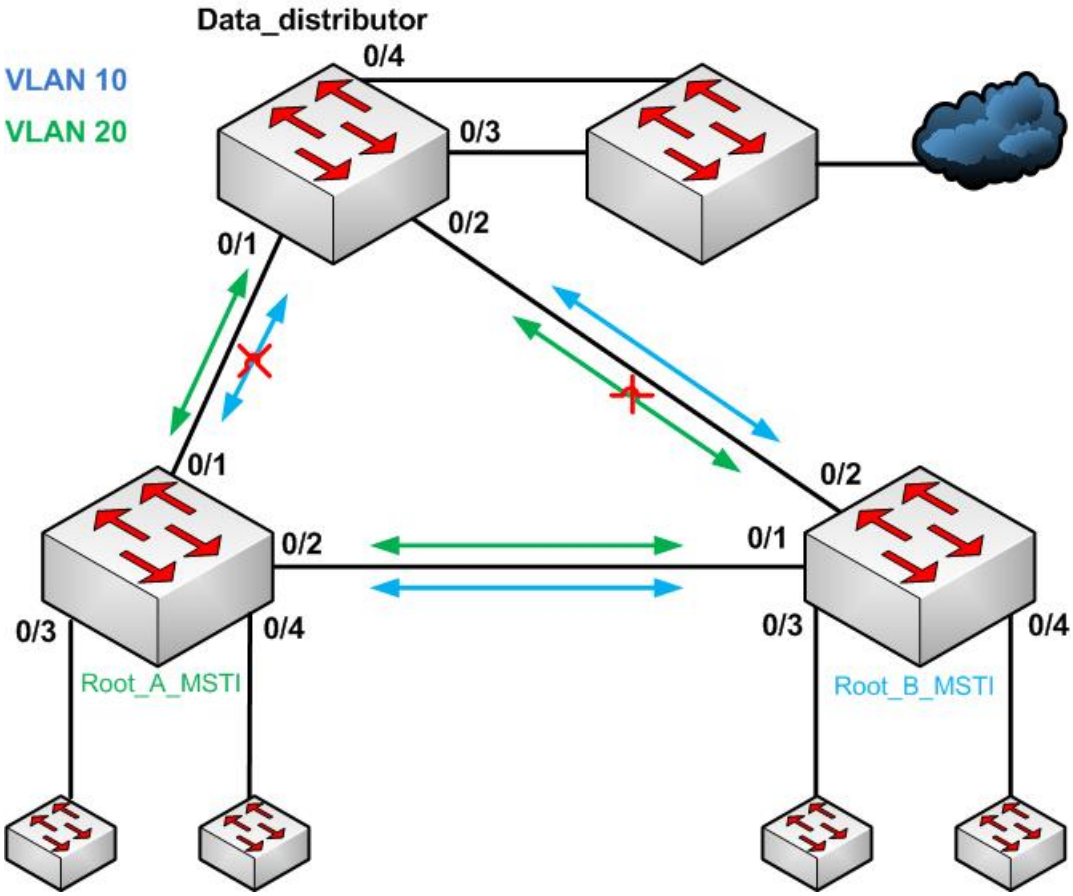
Функционирование MSTP на коммутаторах

MSTP обратно совместим с STP и RSTP. Таким образом коммутаторы с MSTP могут распознать служебные пакеты STP и RSTP и использовать их для построения древовидной топологии.

9.6 Настройка MSTP

На рисунке ниже схематично показано применение MSTP для построения топологии сети. В примере показана отказоустойчивая сеть с распределением нагрузки между магистральными коммутаторами.

Рисунок 14. Применение MSTP для построения топологии сети



Конфигуация коммутатора data_distributor

Создание VLAN и работа с ними подробно описана в соответствующем разделе документации.

```
(als_sw)#vlan database
(als_sw)#vlan 10,20
(als_sw)#exit
```

Назначение коммутатору имени, т.е. вместо имени “als_sw” будет отображаться имя “data_distributor”.

```
(als_sw)#hostname data_distributor
(data_distributor)#configure
(data_distributor)(Config)#no spanning-tree
(data_distributor)(Config)#interface 0/1
(data_distributor)(Interface 0/1)#vlan participation include 10
(data_distributor)(Interface 0/1)#vlan tagging 10
```

```
(data_distributor)(Interface 0/1)#spanning-tree port mode
(data_distributor)(Interface 0/1)#exit
(data_distributor)(Config)#interface 0/2
(data_distributor)(Interface 0/2)#vlan participation include 20
(data_distributor)(Interface 0/2)#vlan tagging 20
(data_distributor)(Interface 0/2)#spanning-tree port mode
(data_distributor)(Interface 0/2)#exit
(data_distributor)(Config)#interface 0/3
(data_distributor)(Interface 0/3)#vlan participation include 10
(data_distributor)(Interface 0/3)#vlan pvid 10
(data_distributor)(Interface 0/3)#exit
(data_distributor)(Config)#interface 0/4
(data_distributor)(Interface 0/4)vlan participation include 20
(data_distributor)(Interface 0/4)vlan pvid 20
(data_distributor)(Interface 0/4)exit
```

Создание instance MST с номером 20

```
(data_distributor)(Config)#spanning-tree mst instance 20
```

Занесение во VLAN mapping table записи о ассоциировании 20 MSTI и 20 VLAN.

```
(data_distributor)(Config)#spanning-tree mst vlan 20 20
```

Команда устанавливает приоритет коммутатора для 20 MSTI.

```
(data_distributor)(Config)#spanning-tree mst priority 20 32768
```

Команда позволяет назначить конфигурации spanning-tree символьного имени.

```
(data_distributor)(Config)#spanning-tree configuration name data_distributor
```

Указание имени ревизии.

```
(data_distributor)(Config)#spanning-tree configuration revision 20
```

Создание instance MST с номером 10

```
(data_distributor)(Config)#spanning-tree mst instance 10
(data_distributor)(Config)#spanning-tree mst vlan 10 10
(data_distributor)(Config)#spanning-tree mst priority 10 32768
```

```
(data_distributor)(Config)#spanning-tree configuration revision 10
(data_distributor)(Config)#spanning-tree forceversion 802.1s
```

Конфигурация магистрального коммутатора A

```
(als_sw)#hostname root_A_MSTI
(root_A_MSTI)#vlan database
(root_A_MSTI)#vlan 10,20
(root_A_MSTI)#exit
(root_A_MSTI)#configure
(root_A_MSTI)(Config)#no spanning-tree
(root_A_MSTI)(Config)#interface 0/1
(root_A_MSTI)(Interface 0/1)#vlan tagging 10,20
(root_A_MSTI)(Interface 0/1)#vlan participation include 10,20
(root_A_MSTI)(Interface 0/1)#spanning-tree guard root
(root_A_MSTI)(Interface 0/1)# spanning-tree port mode
(root_A_MSTI)(Interface 0/1)#exit
(root_A_MSTI)(Config)#interface 0/2
(root_A_MSTI)(Interface 0/2)#vlan tagging 10,20
(root_A_MSTI)(Interface 0/2)#vlan participation include 10,20
(root_A_MSTI)(Interface 0/2)#spanning-tree guard root
(root_A_MSTI)(Interface 0/2)#spanning-tree port mode
(root_A_MSTI)(Interface 0/2)#exit
(root_A_MSTI)(Config)#interface 0/3
(root_A_MSTI)(Interface 0/3)#vlan participation include 10
(root_A_MSTI)(Interface 0/3)#vlan pvid 10
(root_A_MSTI)(Interface 0/3)#exit
(root_A_MSTI)(Config)#interface 0/4
(root_A_MSTI)(Interface 0/4)#vlan participation include 20
(root_A_MSTI)(Interface 0/4)#vlan pvid 20
(root_A_MSTI)(Interface 0/4)#exit
```

Создание на коммутаторе MSTI под номером 10.

```
(root_A_MSTI)(Config)#spanning-tree mst instance 10
(root_A_MSTI)(Config)#spanning-tree mst vlan 10 10
(root_A_MSTI)(Config)#spanning-tree mst priority 10 4096
(root_A_MSTI)(Config)#spanning-tree configuration name right_outhouse
(root_A_MSTI)(Config)#spanning-tree configuration revision 10
```

Создание на коммутаторе MSTI под номером 20.

```
(root_A_MSTI)(Config)#spanning-tree mst instance 20
(root_A_MSTI)(Config)#spanning-tree mst vlan 20 20
(root_A_MSTI)(Config)#spanning-tree mst priority 20 32768
(root_A_MSTI)(Config)#spanning-tree configuration revision 20
(root_A_MSTI)(Config)#spanning-tree forceversion 802.1s
```

Конфигурация магистрального коммутатора B

```
(als_sw)#hostname root_B_MSTI
(root_B_MSTI)#vlan database
(root_B_MSTI)#vlan 10,20
(root_B_MSTI)#exit
(root_B_MSTI)#configure
(root_B_MSTI)(Config)#no spanning-tree
(root_B_MSTI)(Interface 0/1)#interface 0/1
(root_B_MSTI)(Interface 0/1)#vlan tagging 10,20
(root_B_MSTI)(Interface 0/1)#vlan participation include 10,20
(root_B_MSTI)(Interface 0/1)#spanning-tree guard root
(root_A_MSTI)(Interface 0/2)#spanning-tree port mode
(root_B_MSTI)(Interface 0/1)#exit
(root_B_MSTI)(Config)#interface 0/2
(root_B_MSTI)(Interface 0/2)#vlan tagging 10,20
(root_B_MSTI)(Interface 0/2)#vlan participation include 10,20
(root_B_MSTI)(Interface 0/2)#spanning-tree guard root
(root_A_MSTI)(Interface 0/2)#spanning-tree port mode
(root_B_MSTI)(Interface 0/2)#exit
```

```
(root_B_MSTI)(Config)#interface 0/3
(root_B_MSTI)(Interface 0/3)#vlan participation include 10
(root_B_MSTI)(Interface 0/3)#vlan pvid 10
(root_B_MSTI)(Interface 0/3)#exit
(root_B_MSTI)(Config)#interface 0/4
(root_B_MSTI)(Interface 0/4)#vlan participation include 20
(root_B_MSTI)(Interface 0/4)#vlan pvid 20
(root_B_MSTI)(Interface 0/4)#exit
```

Создание MSTI 20.

```
(root_B_MSTI)(Config)#spanning-tree mst instance 20
(root_B_MSTI)(Config)#spanning-tree mst vlan 20 20
(root_B_MSTI)(Config)#spanning-tree mst priority 20 4096
(root_B_MSTI)(Config)#spanning-tree configuration name right_outhouse
(root_B_MSTI)(Config)#spanning-tree configuration revision 20
```

Создание MSTI 10.

```
(root_B_MSTI)(Config)#spanning-tree mst instance 10
(root_B_MSTI)(Config)#spanning-tree mst vlan 10 10
(root_B_MSTI)(Config)#spanning-tree mst priority 10 32768
(root_B_MSTI)(Config)#spanning-tree configuration revision 10
(root_B_MSTI)(Config)#spanning-tree forceversion 802.1s
```

ГЛАВА 10

Конфигурирование IGMP Snooping и MVR

- Что такое IGMP Snooping
- Конфигурирование IGMP Snooping
- IGMP Filtering
- Multicast Vlan Registration
- IGMP Querier

10.1 Что такое IGMP Snooping

IGMP является протоколом управления групповой (multicast) передачей данных. Он основан на протоколе IP. Данный протокол применяется для организации групп сетевых устройств, и определяет следующие типы пакетов:

- Membership Query - используется маршрутизаторами для того чтобы определить наличие хостов, которые хотят получать групповую рассылку
- Join Group - отправляется членом группы для того чтобы сообщить маршрутизатору, что в группе есть как минимум один хост
- Leave Group - отправляется членом группы для того чтобы сообщить маршрутизатору, что хост покидает группу

IP Multicast – это специальная форма широковещания, при которой копии пакетов направляются определенному подмножеству адресатов. Технология групповоий рассылки предоставляет ряд существенных преимуществ по сравнению с традиционным подходом (unicast). Например, добавление новых пользователей не влечет за собой необходимое увеличение пропускной способности сети.

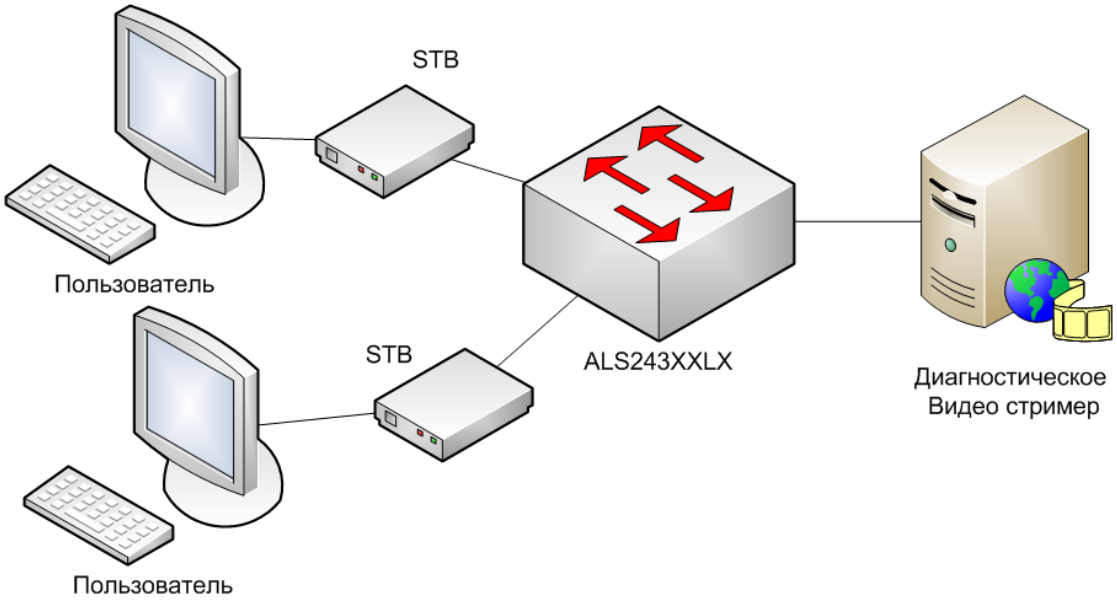
Технология IP Multicast использует адреса с 224.0.0.0 до 239.255.255.255. Поддерживается статическая и динамическая адресация. Примером статических адресов являются:

- 224.0.0.1 - адрес группы, включающей в себя все узлы локальной сети
- 224.0.0.2 - все маршрутизаторы локальной сети
- Диапазон адресов с 224.0.0.0 по 224.0.0.255 зарезервирован для протоколов маршрутизации и других низкоуровневых протоколов поддержки групповой адресации

Остальные адреса динамически используются приложениями.

IGMP Snooping позволяет коммутатору отслеживать IGMP-пакеты потребителей и поставщиков многоадресного (multicast) IP-трафика. Ключевым является тот факт, что коммутатор, работающий на канальном уровне, отслеживает прохождение пакетов сетевоого уровня. Наиболее ярким примером использования IGMP Snooping является его применение в процессе предоставления такой услуги, как IPTV - вещания телевизионных программ абонентам в локальных сетях.

Рисунок 15. Схема предоставления услуги IPTV



10.2 Конфигурирование IGMP Snooping

Конфигурирование IGMP Snooping происходит в несколько этапов.

Остлеживание пакетов, приходящих со стороны клиента

Для того, чтобы начать получение потока данных необходимо присоединиться к соответствующей группе, что осуществляется посредством запроса IGMP Join. Данное сообщение должно обработаться коммутатором, и в таблицу продвижения Multicast пакетов (MFDB) должна попасть запись, содержащая MAC адрес и VLAN, где Mac multicast = 01:00:5e + (23 бита IP Multicast адреса).

01-00-5E-00-00-00 - 01-00-5E-7F-FF-FF 0800 Internet Multicast [RFC1112]

Данная запись позволит предоставлять Multicast поток только тем интерфейсам, с которых пришел запрос Membership Report Join Group.

Пример конфигурации

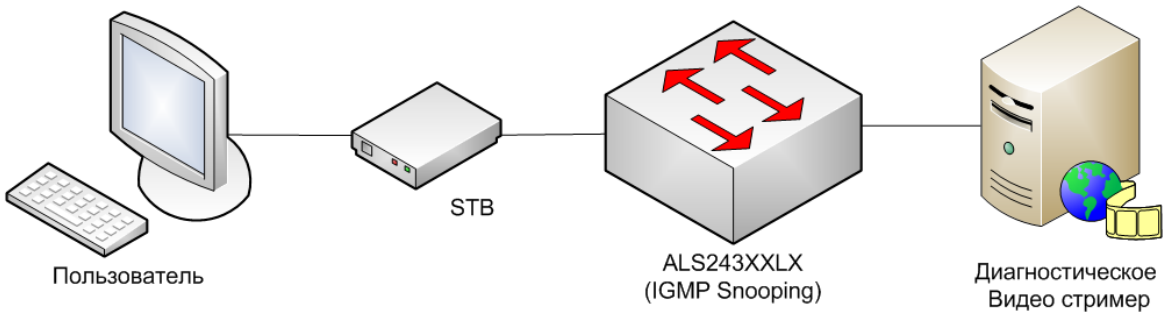
```
(als_sw)#vlan database
(als_sw)(Vlan)#vlan 1000
(als_sw)(Vlan)#set igmp 1000
(als_sw)(Vlan)#exit
(als_sw)#configure
(als_sw)(Config)#interface 0/1
(als_sw)(Interface 0/1)#vlan pvid 1000
(als_sw)(Interface 0/1)#vlan participation include 1000
(als_sw)(Interface 0/1)#set igmp
(als_sw)(Interface 0/1)#exit
(als_sw)(Config)#exit
```

Или

```
(als_sw)#vlan database
(als_sw)(Vlan)#vlan 1000
(als_sw)(Vlan)#exit
(als_sw)#configure
```

```
(als_sw)(Config)#set igmp
(als_sw)(Config)#interface 0/1
(als_sw)(Config)#vlan pvid 1000
(als_sw)(Config)#vlan participation include 1000
(als_sw)(Config)#set igmp
(als_sw)(Interface 0/1)#exit
(als_sw)(Config)#exit
```

Рисунок 16. IGMP Snooping



Пересылка клиентских запросов IGMP Forwarding

Коммутатор должен переслать на маршрутизатор IGMP сообщение о присоединении к группе от клиента, чтобы началось вещание. По умолчанию, коммутатор помещает интерфейс в список интерфейсов за которыми работают маршрутизаторы (для продвижения IGMP со стороны абонента), когда фиксирует на данном порту получение пакета IGMP Query. Параметр Multicast Router Present Expiration Time указывает время ожидания данного пакета на интерфейсе, если пакет IGMP Query в течение этого промежутка времени не фиксируется на интерфейсе, то интерфейс исключается из списка. Можно обойти этот алгоритм и фиксировать данный интерфейс в список, несмотря на то что пакеты IGMP Query могут и не приходить на данный интерфейс. Следующий пример показывает это:

```
(als_sw)#configure
(als_sw)(Config)#interface 0/28
(als_sw)(Config)#vlan tagging 1000
```

```
(als_sw)(Config)#vlan participation include 1000
(als_sw)(Config)#set igmp
(als_sw)(Config)#set igmp mrouter 1000
(als_sw)(Config)#exit
```

Настройка временных параметров IGMP Snooping

Group Membership Interval - указывает время, которое коммутатор будет ждать получение пакета IGMP Report соответствующей группы и с соотвествующего порта перед тем как удалить эту группу из таблицы MFDB.

Пример конфигурации на интерфейсе

```
(als_sw)#configure
(als_sw)(Config)#interface 0/2
(als_sw)(Config)#set igmp groupmembership-interval 300
(als_sw)(Config)#exit
```

Пример конфигурации глобально на свитче

```
(als_sw)#configure
(als_sw)(Config)#set igmp groupmembership-interval 300
(als_sw)(Config)#exit
```

Пример конфигурации глобально для VLAN

```
(als_sw)#vlan database
(als_sw)(Vlan)#set igmp groupmembership-interval 400 300
(als_sw)(Vlan)#exit
```

IGMP Fast-Leave

IGMP Fast-Leave позволяет коммутатору, получив пакет IGMP Leave, незамедлительно удалить соответствующую интрефейс из MFDB для прекращения посылки соответствующей multicast - группы в данный интерфейс.

Данная функция может включаться на интерфейсе или(и) на VLAN.

```
(als_sw)#configure
(als_sw)#interface 0/1
(als_sw)#set igmp fast-leave
```

```
(als_sw)#exit
(als_sw)#exit
(als_sw)#vlan database
(als_sw)#set igmp fast-leave 10
(als_sw)#exit
```

10.3 IGMP Filtering

Коммутатор, получив multicast поток, проверяет его MAC адрес и VLAN в таблице MFDB, обрабатывает его исходя из режима фильтрации multicast потока (mcast_vfm). Пример конфигурации представлен ниже.

Режим продвижения multicast пакетов, когда продвигается группа, только тем интерфейсам, которые подписаны на данную группу и участвуют в обработке данного VLAN

```
(als_sw)#configure
(als_sw)(Config)#mcast_vfm 1000 forward_registered
(als_sw)(Config)#exit
```

Режим продвижения multicast пакетов, когда продвигается группа всем интерфейсам, которые участвуют в обработке данного VLAN, если нет подписанных интерфейсов на эту группу. И только подписчикам в данном VLAN, если таковые есть

```
(als_sw)#configure
(als_sw)(Config)#mcast_vfm 1000 forward_unregistered
(als_sw)(Config)#exit
```

Режим продвижения multicast пакетов, когда продвигается группа всем интерфейсам, которые участвуют в обработке данного VLAN

```
(als_sw)#configure
(als_sw)(Config)#mcast_vfm 1000 forward_all
(als_sw)(Config)#exit
```

10.4 Multicast Vlan Registration

Данная возможность коммутатора необходима для оптимизации выделяемых ресурсов пропускной способности канала. Другими словами, если клиент получает не одну только услугу, предоставляемую посредством IP Multicst, то в большинстве случаев разделение идет с помощью VLAN, когда каждая услуга предоставляется в своем VLAN. Для того чтобы каждому пользователю для multicast вещания не выделялся поток в отдельном VLAN, что значительно бы уменьшало бы пропускную способность канала данных, используется один или несколько VLAN для группы потоков.

Конфигурирование MVR осуществляется следующим образом

```
(als_sw)#configure
(als_sw)(Config)#set igmp mvr 1000
(als_sw)(Config)#set igmp mvr
(als_sw)(Config)#exit
```

10.5 IGMP Querier

Для непрерывного вещания multicast данных конечным пользователям необходимо, чтобы в сети вышестоящий маршрутизатор, находящийся на уровне аггрегации или ядра, запрашивал у всех абонентских устройств информацию об их участии в приеме multicast данных. Коммутатор также может выполнять данную функцию маршрутизатора. Данная функция называется IGMP Querier. IGMP Querier настриавается для конкретного VLAN с опеределенным IP адресом для подстановки в качестве Source IP в пакет IGMP Query. Также можно указать временной интервал между посылками пакета IGMP Query. Если коммутатор получает из вне IGMP Query, то он отключает посылку IGMP Query на определенное время (по умолчанию 60 сек, параметр: IGMP Querier Timer Expiry), после истечении, которого если если не получит снова чужой IGMP Query продолжает сам отправлять IGMP Query.

Конфигурирование IGMP Querier происходит следующим образом

```
(als_sw)#vlan database
(als_sw)(Vlan)# set igmp 90
```

```
(als_sw)(Vlan)# set igmp querier 90 address 192.168.0.1XX
(als_sw)(Vlan)#exit
(als_sw)#configure
(als_sw)(Config)#set igmp querier
(als_sw)(Config)#set igmp querier query-interval 20
(als_sw)(Config)#exit
```

ГЛАВА 11

Контроль трафика на порте

- Storm Control
- Изоляция портов
- Port Security

11.1 Storm Control

Технология шторм контроль

Storm Control предотвращает возникновение ситуаций, когда трафик с одного порта блокирует весь остальной трафик. Он осуществляется следующим образом:

- определение типа пакета (multicast, broadcast или unicast)
- подсчет числа прошедщих через порт пакетов за 1 секунду
- в случае, если количество пакетов достигает заданного уровня, прохождение трафика блокируется

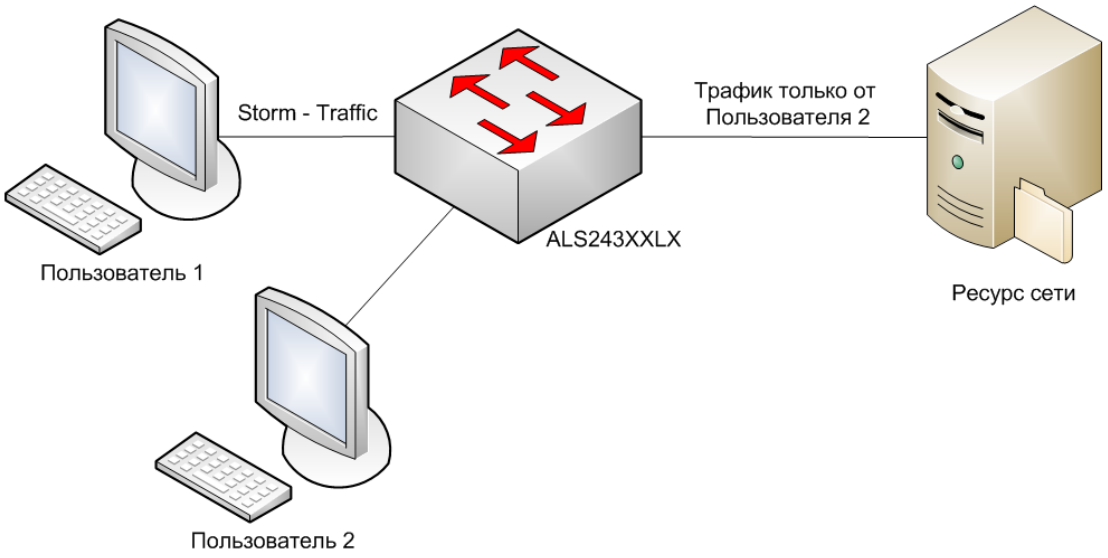
Определение данного уровня осуществляется двумя способами:

- задание максимально допустимого количества пакетов, проходящих через порт за секунду
- задание максимально допустимого уровня загрузки порта в процентах

Трафик блокируется в течение одного временного интервала.

Конфигурирование

Рисунок 17. Конфигурирование порта для ограничения трафика по MAC адресу



Конфигурирование Storm Control осуществляется в режиме настройки интерфейса и выполняется отдельно для каждого типа трафика (multicast, broadcast и unicast).

```
(als_sw)#configure
(als_sw)(Config)#interface 0/1
(als_sw)(Interface 0/1)#storm-control ?
broadcast               Configure broadcast storm-control features.
multicast               Configure multicast storm-control features.
unicast                 Configure destination lookup failure storm-control
                        features.

(als_sw)(Interface 0/1)#storm-control broadcast
<cr>                    Press enter to execute the command.
level                   Configure storm-control thresholds.
rate                    Configure storm-control thresholds.

(als_sw) (Interface 0/1)#storm-control broadcast level ?
```

```
<0-100>                 Enter the storm-control threshold as percent of port
                        speed.
                        Percent of port speed is converted to
                        PacketsPerSecond based on 512 byte
                        average packet size and applied to HW.  Refer to
                        documentation for further
                        details.

(als_sw)(Interface 0/1)#storm-control broadcast rate ?
<0-14880000>            Enter the storm-control threshold in packets per
                        second.
```

Storm Control может быть настроен единовременно для всех портов. Такое конфигурирование осуществляется в режиме Global Configure тойже командой. Например

```
(als_sw)(Config)#storm-control multicast level 50
```

Существует возможнось применения технологии контроля передачи данных (Ethernet Flow Control), которая временно приостанавливает передачу пакетов в случае, когда принемающая сторона не успевает обрабатывать пакеты данных. Данная технология применима только для портов в режиме full-duplex.

```
(als_sw)(Config)#storm-control flowcontrol
```

Отключение контроля, как в режиме конфигурировани интерфейса, так и в режиме Global Configure, осуществляется тойже командой, что и включение, но с исползованием ключевого слова no.

Просмотр конфигурации

Просмотр конфигурации одного порта осуществляется командой show storm-control с указанием интерфейса, например

```
(als_sw)#show storm-control 0/1

        Bcast   Bcast    Mcast   Mcast    Ucast   Ucast
Intf    Mode    Level    Mode    Level    Mode    Level
------ ------- -------- ------- -------- ------- --------
```

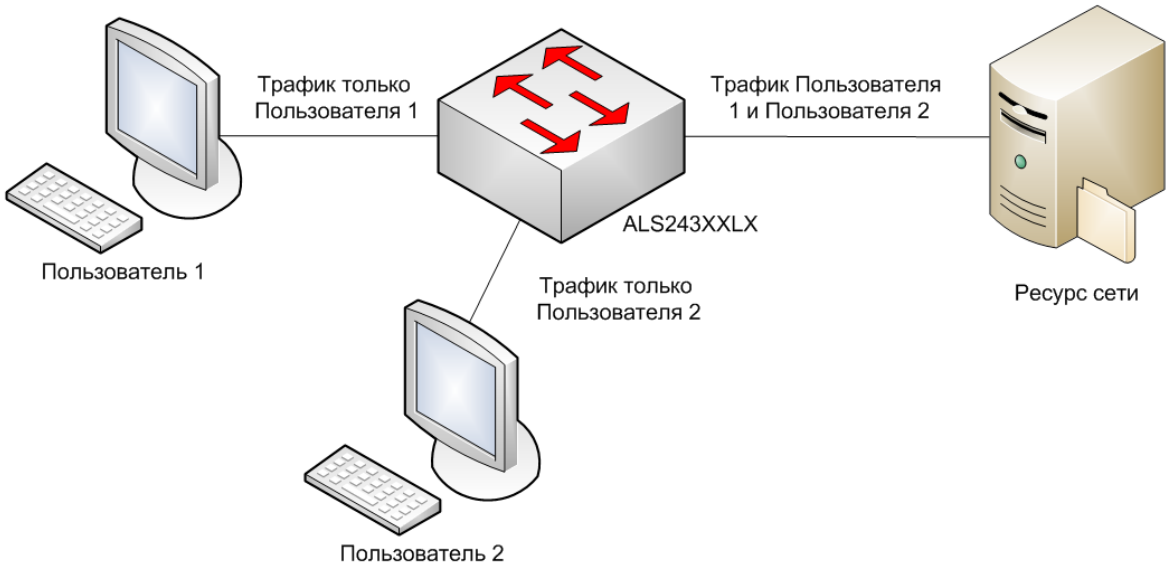
```
0/1    Disable       5% Enable       50% Disable       5%
```

Просмотр конфигурации всех портов выполняется командой

```
(als_sw)#show storm-control all
```

11.2 Изоляция портов

Рисунок 18. Конфигурирование порта для ограничения трафика по MAC адресу



В ряде случаев существует неоходимость в отсутствии возможности прохождения трафика между отдельными портами коммутатора, что достигается т.н. изоляцией портов.

Подобная конфигурация предотвращает передачу пакетов между изолированными портами и не оказывает никакого влияния на обмен пакетами с неизолированными. Изолированные порты объединяются в группы, число которых не превышает трех.

Конфигурирование изоляции портов проходит в два этапа:

- создание группы
- добавление членов группы

Создание группы

Группа создается следующей командой в режиме Global Configure

```
(als_sw)(Config)#switchport protected 0 name groupname
```

Параметрами данной команды являются номер группы (от 0 до 2, в данном случае 0) и ее название (groupname).

Добавление членов группы

Добавление членов группы происходит в режиме конфигурации интерфейса

```
(als_sw)(Interface 0/1)#switchport protected 0
```

Данная команда добавляет конфигурируемый интерфейс в группу с номером 0.

Отмена членства интерфейса в группе, как и удаление самой группы, осуществляется теми же командами и в тех же режимах, но с использованием ключевого слова no.

```
(als_sw)(Interface 0/1)#no switchport protected 0
(als_sw)(Interface 0/1)#exit
(als_sw)(Config)#no switchport protected 0 name
```

Просмотр конфигурации изоляции портов

Просмотреть внесенные в конфигурацию изменения можно следующми командами.

Отображение статуса всех интерфейсов группы

```
(als_sw)#show switchport protected 0
Name..................................... groupname
Member Ports :
0 /1
```

Просмотр участия интерфейса в защищенной группе

```
(als_sw)#show interfaces switchport 0/1 0
Name..................................... groupname
Protected Port ..........................  True in Group   0
```

```
(als_sw)#show interfaces switchport 0/2 1
```

```
Name.....................................

Protected Port ............................. False
```

11.3 Port Security

Технология Port Security

Технология Port Security позваляет ограничить трафик через заданный порт путем сокращения количества и идентификации MAC-адресов узлов сети, имеющих доступ к данному порту.

Трафик от узлов сети, чьи MAC-адреса не заданы в качестве разрешенных, будет блокироваться. Указание единственного разрешенного MAC-адреса в конфигурации гарантирует заданному узлу сети единоличный доступ к порту. В том случае, если достигнуто максимально допустимое количество разрешенных MAC-адресов, и очередной адрес отсутствует в спске идентифицированных, то трафик от данного узла будет блокироваться.

Существует два вида разрешенных адресов:

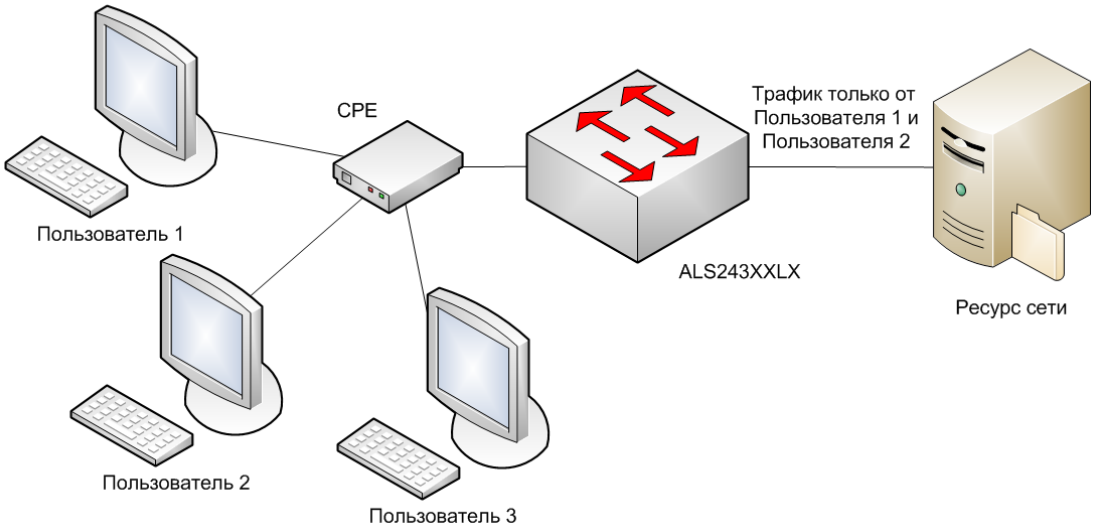
- статические
- динамические

Тип разрешенных адресов зависит от способа их определения. В случае статически разрешенных адресов, определение происходит в Running Configuration - статически. Во втором случае, разрешенные адреса определяются динамически - во время работы коммутатора. Сущетствуют, также, возможность перевода динамически определенных разрешенных адресов в статические.

Существует способ отслеживания ситуаций блокирования пакетов. Он реализуется посредством SNMP. Конфигурирование SNMP см. в соответствующей главе.

Конфигурирование Port Security

Рисунок 19. Конфигурирование порта для ограничения трафика по MAC адресу



Конфигурироание данной технологии осуществляется в режиме Interface Configure, за единственным исключением глобального разрешения в режиме Global Configure командой

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

Включение технологи на конкретном интерфейсе осуществляется следующим образом

```
(als_sw)(Config)#interface 0/1
(als_sw)(Interface 0/1)#port-security
```

Ограничение числа разрешенных MAC-адресов может быть установлено командой

```
(als_sw)(Interface 0/1)#port-security max-dynamic 20
```

и

```
(als_sw)(Interface 0/1)#port-security max-static 30
```

Задание разрешенного MAC-адреса (например, 10:11:12:13:14:15) осуществляется с указанием VLAN ID (например, 16) командой

```
(als_sw)(Interface 0/1)#port-security mac-address 10:11:12:13:14:15 16
```

Просмотр конфигурации осуществляется командой show port-security.

```
(als_sw)#show port-security ?
```

```
<cr>                    Press enter to execute the command.
<slot/port>             Display port security information for a specific
                        interface.
all                    Display port-security information for all interfaces.
dynamic                 Display dynamically learned MAC addresses.
static                  Display statically locked MAC addresses.
violation               Display the source MAC address of the last packet
                        that was discarded on a
                        locked port.
```

Например, просмотр настроек Port Security осуществляется командой

```
(als_sw)#show port-security
Port Security Administration Mode: Enabled
```

Отображение статически разрешенных адресов на конкретном интерфейсе выполняется командой

```
(als_sw)#show port-security static 0/1
Number of static MAC addresses configured: 1
Statically configured MAC Address              VLAN ID
---------------------------------              -------
10:11:12:13:14:15                                   16
```

ГЛАВА 12

Syslog

12.1 Технология Syslog

Syslog — стандарт отправки сообщений о происходящих в системе событиях (логах), использующийся в компьютерных сетях, работающих по протоколу IP. Syslog содержит две части програмного обеспечения:

- клиентская часть генерирует сообщения
- серверная - создает отчеты и анализирует их

Использование syslog особенно важно для устройств, которые не могут каким либо другим образом уведомить о произошедшем сбое. Он реализован для различных платформ и используется во множестве устройств (например принтеры, маршрутизаторы).

Syslog может быть использован для управления и аудита безопасности, а также для обобщения информационных, аналитических и отладочных сообщениях. Журналируемые события имеют уровни от 0 до 7:

- 0 - mergency
- 1 - alert
- 2 - critical
- 3 - error
- 4 - warning
- 5 - notice
- 6 - info
- 7 - debug

Syslog сервер записывает все сообщения в файл, который может быть впоследствии просмотрен. Syslog сервер можно сконфигурировать 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.

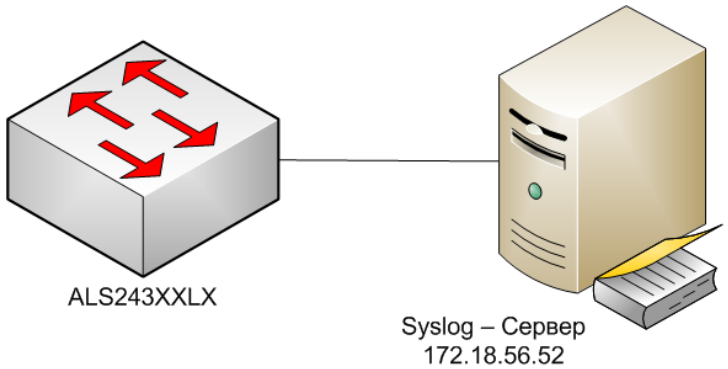
12.2 Механизм протокола syslog

Отправитель посылает получателю короткое текстовое сообщение размером менее 1024 байт. Получателем является syslog сервер. Для транспортировки сообщений syslog используется протокол UDP.

Командный интерфейс к syslog — logging. Некоторые реализации позволяют фильтровать и отображать сообщения системного журнала. Syslog сейчас стандартизирован в пределах Syslog working group of the IETF.

12.3 Настройка syslog

Рисунок 20. Пример использования протокола syslog



Коммутатору присваивается имя ”switch”

```
(als_sw)#hostname switch
```

Переход в режим Global Config

```
(switch)#configure
```

Указание адреса сервера, порта и типа сообщений.

```
(switch)(Config)#logging host 172.18.56.52 ipv4 514 5
```

Включение отправки логов событий на Syslog сервер.

```
(switch)(Config)#logging syslog
```

Переход из режима конфигурирования Global Config в режим Privileged EXEC.

```
(switch)(Config)#exit
```

Команда позволяет посмотреть журнал событий для указанных выше устройств.

```
(switch)#show logging hosts
```

```
Index  IP Address/Hostname       Severity    Port   Status
----- ----------------------  ----------  ------ -------------
1      172.18.56.52             notice      514    Active
```

ГЛАВА 13

Настройка SNMP

- Описание SNMP
- Настройки SNMP
- Отображение настроек SNMP

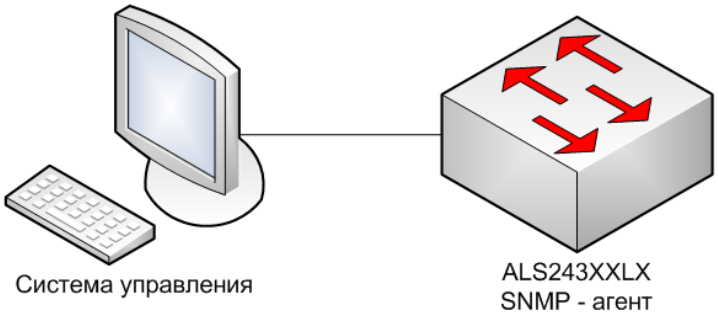
13.1 Описание SNMP

SNMP является протоколом прикладного уровня, предназначенным для облегчения обмена информацией управления между сетевыми устройствами. Основными взаимодействующими лицами протокола являются агенты и системы управления. Если рассматривать эти два понятия на языке ”клиент-сервер”, то роль сервера выполняют агенты, то есть те самые устройства, для опроса состояния которых и был разработан протокол. Соответственно, роль клиентов отводится системам управления - сетевым приложениям, необходимым для сбора информации о функционировании агентов.

Вся информация об объектах системы-агента содержится в так называемой MIB (management information base) - базе управляющей информации, другими словами MIB представляет собой совокупность объектов, доступных для операций записи-чтения для каждого конкретного клиента, в зависимости от структуры и предназначения самого клиента.

SNMP агент может отправлять так называемые трапы(trap) которые сообщают об изменении состояния агента. Например, аутентификация пользователя, изменение статуса порта и т.д. Эти сообщения могут отправляться агентом как на одну систему управления, так и на несколько систем, установленных на разных компьютерах.

Рисунок 21. SNMP



Версии SNMP

Программное обеспечение поддерживает следующие версии SNMP:

- SNMP v.1
- SNMP v.2 / SNMP v.2c

В SNMP v.1 и SNMP v.2 (SNMP v.2c) авторизация пользователя выполняется посредством «строки сообщества» - Community String(community), которая действует как пароль. Удаленная пользовательская программа SNMP и агент SNMP должны использовать одну и ту же community. Пакеты SNMP от любой станции, которая не была авторизована, игнорируются (отбрасываются).

В SNMP v.2 (SNMP v.2c) реализованы более детальные отчеты об ошибках, а так же увеличена скорость получения данных, путем увеличения максимального размера запроса.

Функции SNMP агента

SNMP агент реагирует на запросы со стороны системы управления. Он может считывать данные и отправлять их запрашивающей системе , а также записывать данные и отправлять отчет о записи. В протоколе SNMP существуют следующие команды:

- GET-request - получение значения указанной переменной или информацию о состоянии сетевого элемента
- GET_next_request - получение значения переменной, не зная точного ее имени (следующий логический идентификатор на дереве MIB)

- SET-request - присвоение переменной соответствующего значения. Используется для описания действия, которое должно быть выполнено
- GET response - отклик на GET-request, GET_next_request и SET-request. Содержит также информацию о состоянии (коды ошибок и другие данные)
- GetBulkRequest (только SNMP v.2/v.2c) - запрос пересылки больших объемов данных, например, таблиц

SNMP community

Community выступают в роли своеобразных паролей для доступа к агенту. Для получения доступа к агенту «строка сообщества» системы управления должна совпадать с одной из строк записанных в настройках агента.

Для community в настройках агента существуют 2 уровня разрешений доступа:

- read-only (RO) — дает пользователю право считывать данные из агента, но не дает права на запись
- read-write (RW) — дает пользователю полные права для работы с агентом. Включая как чтение так и запись данных

Обычно используются строки public с разрешением read-only и private с разрешением read-write.

13.2 Настройки SNMP

При загрузке коммутатора snmp агент загружается автоматически. В настройках есть две таблицы: таблица community и таблица трапов. В каждой строке таблицы хранится конфигурация для одного community или для одного трапа.

Для community существуют следующие параметры:

- название (SNMP Community Name)
- ip адрес с которого разрешен запрос системы управления - должен быть одинаков в обоих параметрах (Client IP Address, Client IP Mask), при эжтом по умолчанию в обоих полях ставятся нули, что означает разрешение на доступ с любого адреса
- уровень разрешений (Access Mode) - по умолчанию ставится read-only

- статус - активно/не активно (Status)

Для трапов:

- название (SNMP Trap Name)
- ip адрес на который будет происходить пересылка трапов (IP Address)
- версия протокола, на которой будет происходить передача (SNMP Version)
- статус (Status)

Настройка SNMP community

Community можно использовать для установки доступа систем упарвления к коммутатору. Может быть задать список определенных адресов с которых разрешен доступ.

- snmp-server community <name> - создание нового community с именем <name>
- snmp-server community ipaddr <ipaddr> <name>, snmp-server community ipmask <ipaddr> <name> - разрешение на доступ только с ip адреса <ipaddr> с именем строки <name>
- snmp-server community ro <name> - установка разрешенний в режим read-only
- snmp-server community rw <name> - установка разрешенний в режим read-write

Настройка трапов

Настройки трапов осуществляются подобно настройкам community.

- snmptrap <name> ipaddr <ipaddr> - создание нового трапа с именем <name> и адресом для отправки сообщений <ipaddr>
- snmptrap ipaddr <name> <old_ipaddr> <new_ipaddr> - изменение ip адреса на который будут отсылаться трапы
- snmptrap snmpversion<name> <ipaddr> snmpv1/snmpv2 - установка версии snmp которой будут отправлять трапы

13.3 Отображение настроек SNMP

Таблицы со списком всех «строк сообществ» можно посмотреть с помощью команды:

```
(als_sw)#show snmpcommunity
```

Результатом команды будет описанная в пункте Настройки SNMP таблица, где каждая строка это community, а столбцы — параметры настроек. По умолчанию данная таблица имеет вид:

```
SNMP Community Name  Client IP Address  Client IP Mask  Access Mode  Status
      public              0.0.0.0           0.0.0.0      Read Only   Enable
      private             0.0.0.0           0.0.0.0      Read/Write  Enable
```

Таблицу со списком всех настроенных трапов можно посмотреть с помощью команды:

```
(als_sw)#show snmptrap
```

Результатом будет описанная в пнкте Настройки SNMP таблица, где каждая строка это трап, а столбец – параметры настроек. По умолчанию эта таблица пуста.

ГЛАВА 14

Защита с помощью списков контроля доступа ACL

- Технлогия ACL
- Конфигурирование ACL
- Просмотр совершенных изменений

14.1 Технлогия ACL

ACL позволяет фильтровать сетевой трафик от конкретных узлов сети.

Данная технология, основываясь на полях пакета, отбрасывает их в соответствии со списком контроля доступа. Элемент списка - ACE (access control entries) - является правилом, в соответствии с которым пакет может быть отброшен или пропущен. ACE представляет из себя запись, содержащую указание действия с пакетом (permit или deny) и набор условий, на основе которого будет принематься решение о блокировании пакета. Сравнение прекращается после первого совпадения, вследствие чего существенным является порядок следования элементов списка. Более того, редактирование самого списка ограничено лишь добавлением нового правила,- удаление или редактирование призвольного правила невозможно. Также, список содержит неявное правило (автоматически добавляется в конец списка), которе запрещает прохождение любого пакета. Таким образом, в случае отсутствия совпадений с элементами списка контроля доступа пакет будет отброшен. При этом, порт может иметь только один список контроля доступа. Замена происходит на основе приоритетов (sequence number): старый спиок будет заменен, только если его sequence number больше нового.

Технлогия ACL поддерживает два типа фильтрации:

- на основе IP - адреса

- на основе MAC - адреса

При этом, существует три типа списков контроля доступа:

- расширеный, на основе MAC - адресов отправителя и получателя и, опционально, типе протокола
- расширеный, на основе IP - адресов отправителя и получателя и, опционально, типе протокола
- стандартный, на основе IP - адреса отправителя

14.2 Конфигурирование ACL

Конфигурирование ACL проходит в три этапа:

- создание списка контроля доступа
- добавление элементов списка
- ассоциирование интерфейса с созданным списком

ACL MAC

Блокирование трафика от пользователя 1 к пользователю 2 на основе MAC ACL происходит следующим образом.

Создание MAC - списка доступа

```
(als_sw)(Config)#mac access-list extended test
```

Задание правила

```
(als_sw)(Config-mac-access-list)#deny any 00:90:f5:52:22:2f
00:00:00:00:00:00
```

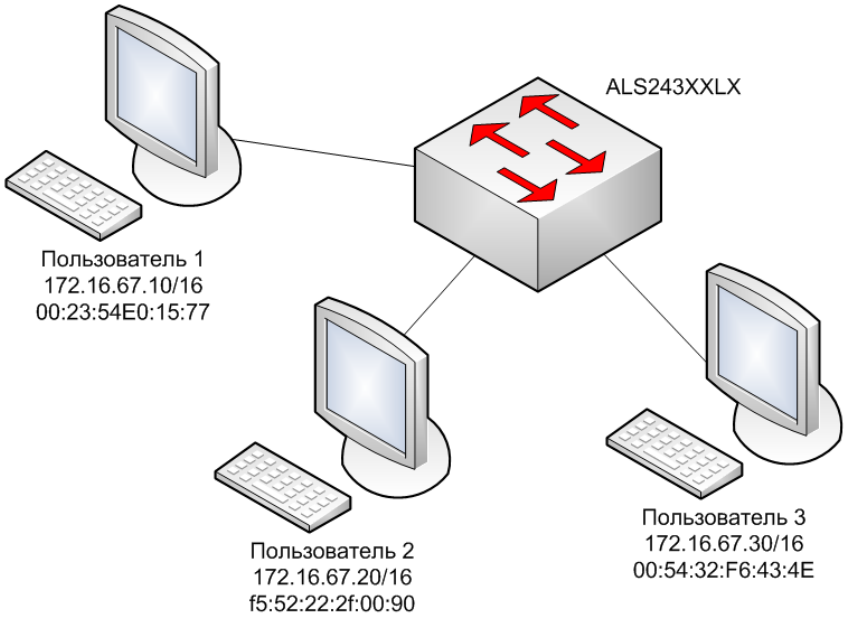
Разрешение всего остального трафика

```
(als_sw)(Config-mac-access-list)#permit any any
(als_sw)(Config-mac-access-list)#exit
```

Применение созданного списка доступа к интерфейсу

```
(als_sw)(Config)#interface 0/19
(als_sw)(Interface 0/19)#mac access-group test in
```

Рисунок 22. MAC ACL



IP ACL

Блокирование трафика от пользователя 1 к пользователю 3 на основе IP ACL.

Создание IP - списка доступа

```
(als_sw)(Config)#ip access-list test
```

Задание правила

```
(als_sw)(Config-ipv4-acl)#deny ip any 172.16.0.30 255.255.0.0
```

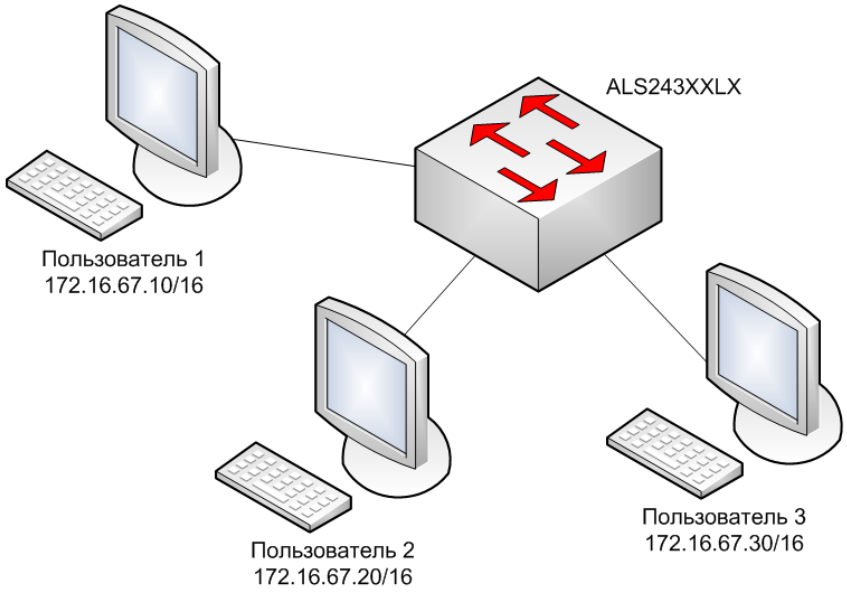
Разрешение всего остального трафика

```
(als_sw)(Config-ipv4-acl)#permit ip any any
(als_sw)(Config-ipv4-acl)#exit
```

Применение созданного списока доступа к интерфейсу с указанием группы, направления и последовательности

```
(als_sw)(Config)#interface 0/1
(als_sw)(Interface 0/1)#ip access-group test in 1
```

Рисунок 23. IP ACL



14.3 Просмотр совершенных изменений

Просмотр ограничения на основе mac-адресов можно осуществить с помощью команды

```
(als_sw)#show mac access-lists [name]
```

В данная команда имеет необязаттельный парметр - имя списка.

Просмотр ограничения на основе ip-адресов можно осуществить с помощью команды

```
(als_sw)#show ip access-lists <number>
```

Парметром данной команды является номер списка для просмотра.

Существует также команда для просмотра ACL на оределенном интерфейсе

```
(als_sw)#show access-lists interface <u/s/p> in
```

ГЛАВА 15

Конфигурирование QoS

- Понятие QoS
- Сервисная модель Differentiated Service (DiffServ)
- Базовые функции QoS
- Настройка QoS

15.1 Понятие QoS

QoS определяет вероятность обеспения сетью соответствия трафика указанного сервиса заданным критериям. К числу критериев можно отнести:

- bandwidth - полоса пропускания, описывает номинальную пропускную способность среды передачи информации, определяет ширину канала
- delay - задержка при передаче пакета
- jitter - изменение величины задержки при передаче пакетов
- packet loss - потери пакетов, определяет количество пакетов, недостигших узла назначения

15.2 Сервисная модель Differentiated Service (DiffServ)

Differentiated Service (DiffServ, RFC 2474/2475) - модель дифференцированного обслуживания. Определяет обеспечение QoS на основе четко определенных компонентов, комбинируемых с целью предоставления требуемых услуг. Архитектура DiffServ предполагает наличие классификаторов и формирователей трафика на границе сети, а также поддержку функции распределения ресурсов в ядре сети в целях обеспечения требуемой политики пошагового обслуживания (Per-Hop Behavior - PHB). Данная модель разделяет трафик на классы, вводя несколько уровней QoS. DiffServ состоит из следующих функциональных блоков:

- граничные формирователи трафика (классификация пакетов, маркировка, управление интенсивностью)
- реализаторы PHB политики (распределение ресурсов, политика отбрасывания пакетов)

Differentiated Services Code Point (DSCP) используется для обозначения специального байта данных стандартного заголовка IP — пакета. Этот байт несет информацию о приоритете трафика. Пакеты, помеченные более высоким приоритетом, передаются быстрее (менее приоритетные становятся в очередь).

Использование терминологии DSCP подразумевает 6 старших бит, которые кодируют уровень класса обслуживания от 0 (минимальный приоритет) до 7 (максимальный приоритет) и приоритет удаления (от 0, когда приоритет удаления максимальный, до 7, когда приоритет удаления минимальный — кодирование приоритета удаления «обратное»). Таким образом получается число от 0, до 64, кодирующее приоритет (чем больше число, тем трафик важнее).

DiffServ характеризуется приоритезацией трафика (Prioritization).

15.3 Базовые функции QoS

Базовые функции QoS заключаются в обеспечении необходимых параметров сервиса и определяются по отношению к трафику через классификацию, разметку, управление перегрузками, предотвращение перегрузок и регулирование. Функционально классификация и разметка чаще всего обеспечиваются на входных портах оборудования, а управление и предотвращение перегрузок – на выходных.

15.4 Настройка QoS

Далее рассмотрен пример настройки QoS в соответствии с моделью DiffServ, где разделение на классы (mgmt и iptv) будет происходить в соответствии с VID (118 и 90).

Шаг 1. Определение классов трафика

```
configure
class-map match-all mgmt ipv4
```

```
match vlan 118
exit
class-map match-all iptv ipv4
match vlan 90
exit
```

Шаг 2. Определение политик для классов трафика

```
configure
policy-map qos_set in
class mgmt
assign-queue 7
exit
class iptvm
assign-queue 5
exit
exit
```

Шаг 3. Определение дисциплины обработки очередей

На основе созданных политик пакеты помещаются в отдельные очереди, обработка которых может происходить в соответствии с одним из двух алгоритмов: Weighted Round Robin (работает по-умолчанию) или Strict.

Strict

Данный алгоритм на каждом шаге выбирает пакет из наиболее приоритетной среди непустых очереди.

Включение алгоритма Strict

```
cos-queue strict 0 1 2 3 4 5 6 7
```

Задание приоритетов очередей происходит в соответствии с порядком следования.

Weighted Round Robin

Данный алгоритм выбирает пакеты из очередей циклически, при этом количество пакетов, выбранных из одной очереди, соответствует заданному весу в процентном соотношении.

Включеие алгоритма Weighted Round Robin

```
cos-queue min-bandwidth 1 3 6 10 14 16 20 30
```

Задание весов очередей происходит в соответствии с порядком следования, при этом сумма всех весов не должна превышать 100.

Применение созданной ранее политики на интерфейсе

```
interface 0/1
service-policy in qos_set
exit
interface 0/2
service-policy in qos_set
exit
```

ГЛАВА 16

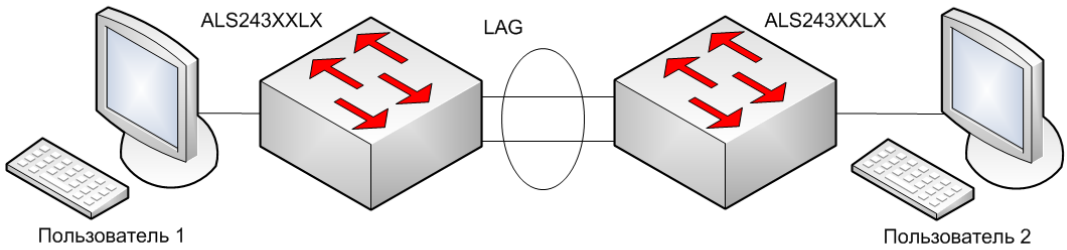
Конфигурирование Link Aggregation

- Что такое Link Aggregation
- Конфигурирование Link Aggregation
- Просмотр статуса Link Aggregation

16.1 Что такое Link Aggregation

Агрегация каналов (Link aggregation, IEEE 802.3ad) — это технология объединения нескольких физических каналов в один логический, что увеличивает пропускную способность магистральных каналов и повышает их надежность. Главное преимущество агрегации каналов заключается в повышении скорости, что достигается посредством наращивания числа входящих в него физических каналов. Это также существенно повышет и надежность канала, так как при невозможности передачи данных по одному из физических каналов, это осуществимо по другому.

Рисунок 24. Типовая схема сети с агрегацией каналов



16.2 Конфигурирование Link Aggregation

Конфигурирование LAG проходит в два этапа:

- создание LAG
- добавление членов LAG

Создание LAG

Для создания LAG с именем, например test, в который будут включаться физические интерфейсы (по умолчанию создается динамический LAG), необходимо выполнить следующие команды.

```
(als_sw)#configure
(als_sw)(Config)#port-channel test
```

Просмотреть статус созданного логического интерфейса можно слудующим образом

```
(als_sw)#show port-channel brief
Logical   Port-Channel Name Link State Trap     Type    Mbr Ports Active Ports
 Interface                              Flag
--------- ---------------- ---------- -------- ------- --------- ------------
 1/1       test             Down       Enabled  Dynamic
```

Данный логический интерфейс имеет имя test (Port-Channel Name) и номер 1/1 (Logical Interface) и не содержит физичечких интерфейсов (Mbr Ports).

Добавление членов LAG

Добавление интерфейсов к членам созданного LAG осуществляется с помощью следующей команды (выполняемой в режиме конфигурирования интерфейса).

```
(als_sw)(Interface unit/slot/port)#addport unit/slot/port
```

В данной команде unit/slot/port, указываемый в аттрибутах, есть номер созданного логического интерфейса.

Добавление в члены LAG с номером 1/1 двух физических интерфейсов осуществляется с помощью следующей команды.

```
(als_sw)(Interface 0/19)#addport 1/1
```

```
(als_sw)(Interface 0/20)#addport 1/1
```

16.3 Просмотр статуса Link Aggregation

Просмотр статуса Link Aggregation можно осуществиь при помощи следующей команды.

```
(als_sw)#show port-channel brief
Logical   Port-Channel Name Link State Trap     Type    Mbr Ports Active Ports
 Interface                              Flag
--------- ---------------- ---------- -------- ------- --------- ------------
 1/1       test             Down       Enabled  Dynamic 0/19,0/20
```

Изменения в столбце Mbr Ports показывают, что созданный нами LAG теперь содержит два интерфейса. Просмотр статуса Link Aggregation можно осуществить также при помощи следующей команды.

```
(als_sw)#show port-channel all
 Log.       Channel            Adm.      Mbr    Device/       Port      Port
 Intf        Name        Link  Mode Type Ports  Timeout       Speed     Active
------ -------------- ------ ---- ---- ------ ------------ --------- -------
 1/1    test           Down   En.  Dyn. 0/19   actor/long    Auto      False
                                                partner/long
                                      0/20   actor/long    Auto      False
                                             partner/long
```

Столбец Port Active показывает активность порта в LAG: True — порт включен, False — выключен.

ООО «Компания АЛСиТЕК»,
410012 Россия, г.Саратов,
ул. Б.Казачья, 6
Тел: +7-8452-79-94-98
Факс: +7-8452-79-94-97
www.alstec.ru
office@alstec.ru