ЛАБОРАТОРНАЯ РАБОТА № 1

Общие настройки интерфейса switch и router на платформе CISCO Packet Tracer

1.1. Цель работы

Целью работы являются ознакомление с сетевым оборудованием фирмы Cisco, изучение особенностей операционной среды CISCO IOS, получение навыков по базовой настройке маршрутизатора и коммутатора CISCO.

1.2. Теоретическая часть

На сегодняшний день на рынке IT существуют не так уж и много сетевых симуляторов.

Широко известны такие симуляторы, как:

- BOSON NET SIM;
- CISCO Router eSim;
- Cisco Packet Tracker;
- Network Emulator;
- Dynamips;
- Cisco 7200 Simulator.

Из них наиболее распространенными в плане использования для обучения являются Boson Net Sim, Cisco Packet Tracer и Network Emulator. Данный цикл лабораторных работ посвящён оборудованию Cisco. Рассмотрим возможности Cisco Packet Tracer более подробно.

1.2.1. Cisco Packet Tracer

Данный программный продукт разработан компанией Cisco и рекомендован для использования при изучении телекоммуникационных сетей и сетевого оборудования.

Packet Tracer 6.0 включает следующие особенности:

- моделирование логической топологии: рабочее пространство для того, чтобы создать сети любого размера на CCNA-уровне сложности;
- моделирование в режиме реального времени;
- режим симуляции;
- моделирование физической топологии: более понятное взаимодействие с физическими устройствами с использованием таких понятий, как город, здание, стойка и т.д.;
- улучшенный GUI, необходимый для более качественного понимания организации сети, принципов работы устройства;

- многоязыковая поддержка: возможность перевода данного программного продукта практически на любой язык, необходимый пользователю;
- усовершенствованное изображение сетевого оборудования со способностью добавлять / удалять различные компоненты;
- наличие Activity Wizard позволяет студентам и преподавателям создавать шаблоны сетей и использовать их в дальнейшем.

С помощью данного программного продукта возможно конструировать и конфигурировать сети и производить в них поиск неисправностей. Packet Tracer дает возможность более подробно представлять новейшие технологии, тем самым делая учебный процесс чрезвычайно полезным с точки зрения усвоения полученного материала.

Данный симулятор позволяет проектировать свои собственные сети, создавая и отправляя различные пакеты данных, сохранять и комментировать свою работу. Возможно использовать такие сетевые устройства, как коммутаторы второго и третьего уровней, рабочие станции, определять типы связей между ними и соединять их. После того, как сеть спроектирована, можно приступать к конфигурированию выбранных устройств по средству терминального доступа или командной строки (см. рис.1.1).

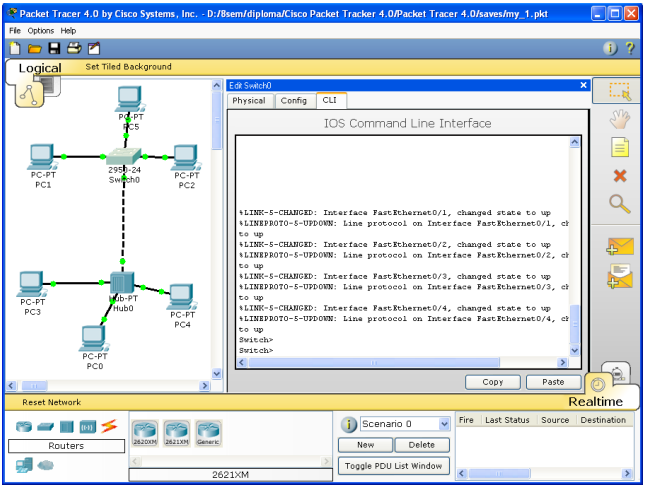


Рис. 1.1. Cisco Packet Tracer 4.0

Отличительной особенностью данного симулятора является наличие в нем «Режима симуляции» (рис.1.2).

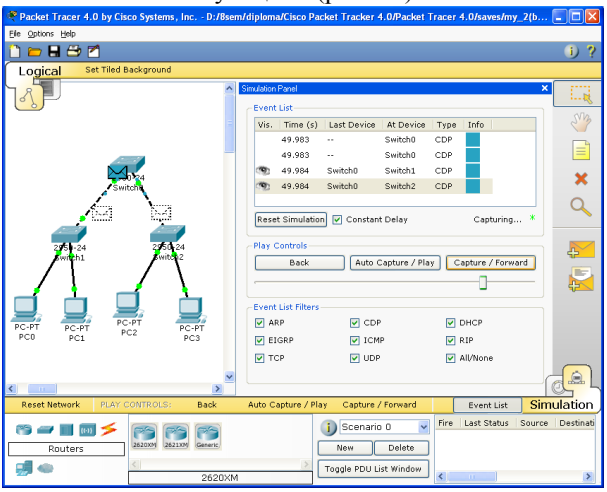


Рис. 1.2. Режим «Симуляции» в Cisco Packet Tracer 6.0

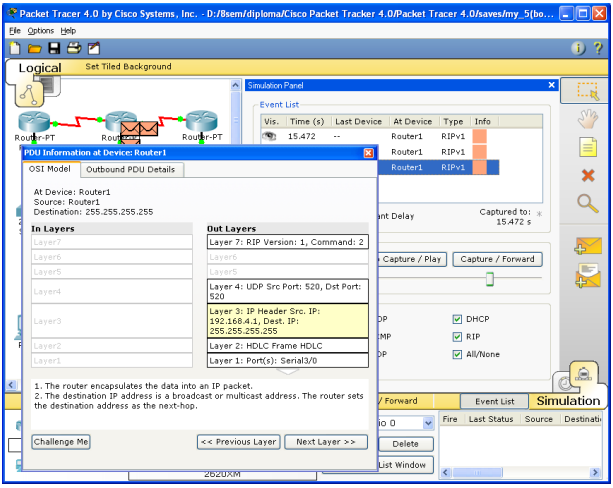


Рис. 1.3. Анализ семиуровневой модели OSI в Cisco Packet Tracer 6.0

В данном режиме все пакеты, пересылаемые внутри сети, отображаются графически. Эта возможность позволяет наглядно продемонстрировать, по какому интерфейсу в данный момент перемещается пакет, какой протокол используется и т.д.

В «Режиме симуляции» возможно не только отслеживать используемые протоколы, но и видеть, на каком из семи уровней модели OSI данный протокол задействован (см.рис.1.3).

Такая кажущаяся на первый взгляд простота и наглядность делает практические занятия чрезвычайно полезными, совмещая в них как получение, так и закрепление полученного материала.

Packet Tracer способен моделировать большое количество устройств различного назначения, а также немало различных типов связей, что позволяет проектировать сети любого размера на высоком уровне сложности:

моделируемые устройства:

- коммутаторы третьего уровня:
  - Router 2620 XM;
  - Router 2621 XM;
  - Router-PT;
- коммутаторы второго уровня:
  - Switch 2950-24;
  - Switch 2950T;
  - Switch-PT;
  - соединение типа «мост» Bridge-PT;
- сетевые концентраторы:
  - Hub-PT;
  - повторитель Repeater-PT;
- оконечные устройства:
  - рабочая станция PC-PT;
  - сервер Server-PT;
  - принтер Printer-PT;
- беспроводные устройства:
  - точка доступа AccessPoint-PT;
- глобальная сеть WAN.

типы связей:

- консоль;
- медный кабель без перекрещивания (прямой кабель);
- медный кабель с перекрещиванием (кросс-кабель);
- волоконно-оптический кабель;
- телефонная линия;
- Serial DCE;
- Serial DTE.

Также целесообразно привести те протоколы, которые можено отслеживать:

- ARP (Address Resolution Protocol);

- CDP (Cisco Discovery Protocol);
- DHCP (Dynamic Host Configuration Protocol);
- EIDRP (Enhanced Interior Gateway Protocol);
- ICMP (Internet Control Message Protocol);
- RIP (Routing Information Protocol);
- TCP (Transmission Control Protocol);
- UDP (User Datagram Protocol).

1.2.2. Cisco IOS

Режимы ввода команд

Cisco IOS (от англ. Internetwork Operating System — межсетевая операционная система) – это многозадачная операционная система, выполняющая функции маршрутизации, коммутации, балансировки нагрузки на сеть ЭВМ и передачи данных по сети.

Cisco IOS имеет специфичный интерфейс командной строки (command line interface, CLI). CLI включает широкий набор команд. Доступность команды определяется «режимом» и уровнем привилегий данного пользователя. При работе в командной строке Cisco IOS существует несколько режимов ввода команд («контекстов»).

Контекст пользователя активизируется при подсоединении к устройству (коммутатору, маршрутизатору); обычно при подключении через сеть требуется пароль, а при подключении через консольный порт пароль не нужен. В этот же контекст командная строка автоматически переходит при продолжительном отсутствии ввода в контексте администратора. В контексте пользователя доступны только простые команды, не влияющие на конфигурацию оборудования. Вид приглашения командной строки данного контекста:

name[1]*>*

Контекст администратора активизируется командой *enable*, выполненной в контексте пользователя; при этом требуется пароль администратора. В контексте администратора доступны команды, позволяющие получить полную информацию о конфигурации марш-оборудования и его состоянии, команды перехода в режим конфигурирования, команды сохранения и загрузки конфигурации. Вид приглашения командной строки:

name#

Обратный переход в контекст пользователя производится по команде *disable* или по истечении установленного времени отсутствия активности. Завершение сеанса работы - команда *exit*.

---

[1] Вместо *name* на начальном этапе обычно указывается Switch или Router в зависимости от типа используемого оборудования, а при наличии имени устройства выводится оно.

Глобальный контекст конфигурирования активизируется командой config terminal ("конфигурировать через терминал"), выполненной в контексте администратора. Вид приглашения командной строки следующий:

name(config)#

Глобальный контекст конфигурирования содержит как непосредственно команды конфигурирования оборудования, так и команды перехода в контексты конфигурирования подсистем.

Контекст конфигурирования интерфейса активизируется командой *interface имя_интерфейса* (например, *interface serial0 или fastethernet0*), выполненной в глобальном контексте конфигурирования. Приглашение контекста:

name(config-if)#

Типы интерфейсов, используемые в распространенных сетевых технологиях, хорошо известны - это интерфейсы ethernet (fastethernet, gigabitethernet), применяемые в сетях **LAN**, интерфейсы vlan виртуальных локальных сетей **VLAN**, интерфейсы serial последовательного подключения Serial, применяемые, например, для соединения с устройствами CSU/DSU (Channel Service Unit/Digital Service Unit), асинхронные интерфейсы async для связи с модемами, интерфейсы atm, используемые для соединения с ATM-коммутаторами, например: компании поставщика услуг или локальные коммутирующие звенья сети, локальные интерфейсы, петли обратной связи (программные loopback-интерфейсы) и т.д.

Конфигурирование интерфейса выполняется с помощью команды *interface*, которой передаются в качестве аргументов тип интерфейса и номер физического порта.

Например, чтобы обратиться к 0-му порту FastEthernet, следует ввести:

name > enable
name # conf t
name (config)# interface fastethernet 0
name (config - if)# ip address ...
name (config - if)# encapsulation ...

Или, например, можно обратиться к последовательному интерфейсу:

name (config)# interface serial 0

Контекст конфигурирования процесса динамической маршрутизации[2] активизируется командой *router протокол*

[2] Контекст применим для маршрутизаторов

номер_процесса (например, *router ospf 1*), выполненной в глобальном контексте конфигурирования. Вид приглашения командной строки:

router(config-router)#

Контекст конфигурирования параметров терминальной линии активизируется командой *line номер_линии* (*line con 0* – для настройки консоли, *line vty 0÷4* – для настройки сеансов Telnet), выполненной в глобальном контексте конфигурирования. Приглашение данного режима:

name(config-line)#

Контекст ROM Monitor. Если при загрузке произошли ошибки и устройство не смогло загрузить IOS, оно перейдет в так называемый ROM Monitor-режим с приглашением *rommon#>* или >. Из него можно вручную с помощью команды *boot* пытаться запустить другой файл из памяти.

Существует множество других режимов, некоторые из них находятся внутри других контекстов конфигурирования.

Выход из глобального контекста конфигурирования в контекст администратора, а также выход из любого подконтекста конфигурирования в контекст верхнего уровня производится командой *exit* или нажатием сочетания клавиш <*Ctrl-Z*>. Кроме того, команда *end*, выполненная в любом из контекстов конфигурирования, немедленно завершает процесс конфигурирования и возвращает оператора в контекст администратора.

Любая команда конфигурации вступает в действие немедленно после ввода (а не после возврата в контекст администратора).

Отмена любой команды (отключение опции или режима, включаемых командой, снятие или удаление параметров, назначаемых командой) производится подачей этой же команды с префиксом "*no*", например:

router(config-if)#shutdown !отключить интерфейс

router(config-if)#no shutdown !включить интерфейс

Команда *show* с многочисленными параметрами позволяет просматривать всевозможную информацию о состоянии памяти, операционной системы, линий, протоколов и многое другое.

Одна из особенностей Cisco IOS - особая система контекстно-чувствительной справки. Для работы с контекстными справками пользуются короткой командой «*?*». Если ввести только «*?*», то выведутся все возможные команды, чтобы узнать о нужной команде конкретнее, нужно передать команду как параметр команде «*?*». Например:

show ?

Если в конце выведенного списка команд стоит слово <*--more--*>, это означает, что список содержит более одной страницы. Для

просмотра следующей страницы вы можете нажать пробел, одной следующей строки – <*Enter*>.

Cisco IOS поддерживает автозавершение вводимой команды при нажатии клавиши <*Tab*>, если команда распознана.

Перезагрузка, если таковая необходима, выполняется по команде *reload*, выполненной в контексте администратора.

Схема взаимосвязи контекстов Cisco IOS показана на рис. 1.4.

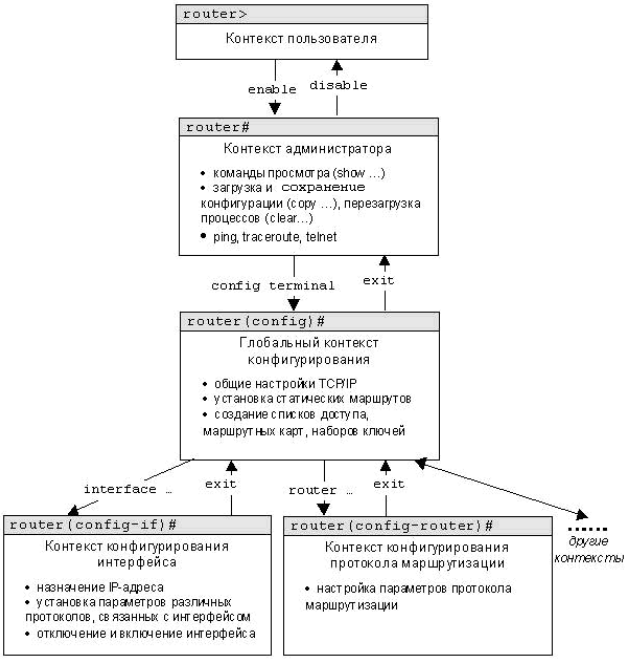


Рис. 1.4. Контексты (режимы ввода команд) командной строки Cisco IOS

Задание имени устройства

Также обычным для начальной настройки маршрутизатора является задание для него более понятного и удобного в данной сети имени, для этого служит команда *hostname новое_имя*. Рекомендовано применять единые правила именования всех устройств.

Имея логическую единообразную схему именования, вам будет легко запомнить имя устройства или подобрать его, если вы вдруг забудете точное название.

Операции с конфигурационными файлами

Cisco IOS хранит конфигурационные файлы, в которых хранится информация о всех настройках маршрутизатора, паролях и логинах, используемых схемах и протоколах. Таких файлов два:

– начальная конфигурация, которую Cisco IOS использует при загрузке (*startup-config*);

– текущая конфигурация, в которую записываются все изменения, внесенные администратором в текущем сеансе работы (running-config).

Текущая отредактированная конфигурация устройства может быть переписана в качестве начальной конфигурации загрузки с помощью команды:

name# copy running-config startup-config

Если этого не делать, изменения после перезагрузки устройства применены не будут.

Чтобы сбросить все только что внесенные изменения, нужно выполнить команду:

name# copy startup-config running-config

Текущую конфигурацию всегда можно получить командой:

name# show run

Одной из проблем открытого хранения и пересылки конфигурационных файлов всегда была опасность разглашения паролей. В случае задания пароля в системе с помощью обычной команды

enable password <пароль>

пароль хранится в открытом виде, и любой перехвативший конфигурационный файл злоумышленник становится обладателем данного пароля. Для защиты от этого используется модифицированная команда

enable secret <пароль>.

Просмотрев конфигурационный файл, можно убедиться, что в открытом виде пароль администратора не отображается. Но в этом случае, если пароль забыт, восстановить его по зашифрованной записи невозможно.

Задание системного приглашения

Часто бывает необходимо добавлять в системное приглашение другую информацию — для этого предназначена команда *prompt*.

name(config)#prompt %h:%n%p

name(config)#exit

name:5#

Здесь – три управляющие последовательности, чтобы составить приглашение из имени узла *(%h),* номера команды (%n) и соответствующего символа приглашения для текущего командного режима (%p).

Для восстановления приглашения, применяемого по умолчанию, предназначена команда *no prompt*.

Установка внутренних часов

Внутренние часы устанавливаются с помощью команды *clock set*. Это не конфигурационная команда (чтобы выполнить ее, необходимо находиться в режиме настройки, но предварительно исполнять команду *configure terminal* не нужно), поэтому она не хранится в конфигурации. Для представления времени используется 24-часовая система.

name#clock set 13:00:00 20 mar 2014

Системные баннеры

Предусмотрено несколько стандартных сообщений для пользователей. Обычно такие сообщения ассоциируются с процессом входа в систему. Например, при входе пользователь может видеть баннер сообщения дня (*motd banner*), за которым идет баннер входа в систему (*login banner*), после чего уже следует само приглашение. После успешной регистрации на экране пользователя может выводиться баннер исполнения (*exec banner*).

Все команды настройки баннеров имеют один и тот же формат:

тип_баннера # сообщение #

Для создания баннеров любого типа используйте команду *banner*, указав тип баннера и сообщение:

name(config)#banner motd # сообщение #

name(config)#banner login # сообщение #

name(config)#banner exec # сообщение #

Обычно если баннеры определены, то они выводятся на экране. Отключить баннер невозможно, его можно удалить с помощью ключевого слова *no* в команде *banner*:

Router(config)#no banner incoming

Включение системы DNS

Система доменных имен (DNS) позволяет, в принципе, обойтись без ведения таблицы узлов, однако, как уже отмечалось, необходимо явно указать узлы для любых IP-адресов, присутствующих в конфигурации. По умолчанию система DNS включена; чтобы специально включить ее, используется команда

ip domain-lookup.

Чтобы отключить поиск в системе DNS, используйте команду с ключевым словом *no*:

no ip domain-lookup

Установка паролей

Для защиты маршрутизатора Cisco используются пять паролей. Первые два пароля служат для установки разрешенного пароля, который защищает привилегированный режим. Пароль запрашивается у пользователя после ввода команды *enable*. Остальные три пароля служат для настройки паролей для доступа пользователя через

консольный порт, вспомогательный порт и по протоколу Telnet.

Разрешенные пароли

Для установки разрешенного пароля необходимо находиться в режиме глобального конфигурирования.

Router(config)#enable ?

last-resort	Define enable action if no TACACS servers respond
password	Assign the privileged level password
secre	Assign the privileged level secret
use-tacacs	Use TACACS to check enable passwords

Last-resort (крайний случай) используется после установки аутентификации через сервер *tacacs*, который недоступен. Такой режим позволяет администратору даже в этом случае войти в систему маршрутизатора. Однако подобный режим недоступен при работающем сервере *tacacs.*

Password (пароль) служит для установки разрешенного пароля (enable password) в устаревших системах с версиями до 10.3. Не используется, если установлен разрешенный секрет (enable secret).

Secret (секрет) – новый шифрованный пароль. После установки перекрывает действие разрешенного пароля.

Use-tacacs (использовать tacacs) указывает маршрутизатору на аутентификацию через сервер tacacs. Это удобно, когда приходится обслуживать десятки и сотни маршрутизаторов. Как иначе изменить пароль на 200 маршрутизаторах? Сервер tacacs позволяет однократно изменить пароль, который будет действовать на все устройства.

Router(config)#enable secret todd

Router(config)#enable password todd

При попытке ввода одинакового разрешенного пароля и разрешенного секрета выводится вежливое предупреждение о недопустимости такого выбора. Однако при повторном вводе того же самого пароля он будет установлен в маршрутизаторе даже при совпадении с разрешенным секретом. Между тем пароли не работают одновременно. В новых маршрутизаторах (а не в старых унаследованных) можно не беспокоиться об использовании разрешенного пароля.

Пароли пользовательского режима присваиваются командой *line.*

Aux (вспомогательный) служит для установки пароля пользовательского режима для вспомогательного порта. Обычно применяется для настройки в маршрутизаторе параметров модема, но может служить и для доступа к консоли.

Console (консоль) служит для установки пароля консоли пользовательского режима.

Vty (виртуальный терминал) служит для установки в маршрутизаторе пароля Telnet. Если такой пароль не установлен, то по

умолчанию использование Telnet запрещено.

Для настройки паролей пользовательского режима сначала конфигурируется нужная линия (line), а затем вводится команда *login* или *no login* для вывода из маршрутизатора приглашения аутентификации.

Вспомогательный пароль

Для настройки вспомогательного пароля следует перейти в режим глобального конфигурирования и ввести *line aux ?*. Список выбора содержит только строку 0-0, поскольку существует один вспомогательный порт.

Router#config t

Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#line aux ?

<0-0> First Line number

Router(config)#line aux 0

Router(config-line)#login

Router(config-line)#password todd

Важно помнить о команде login, иначе вспомогательный порт не выведет приглашения для аутентификации.

Пароль консоли

Для установки пароля консоли служит команда line console 0. Однако если подобно настройке вспомогательного порта ввести line console 0 ?, то вы получите ошибку. Необходимо ввести команду line console 0, причем после этого недоступен экран справки. Для возвращения на один уровень вверх следует задать " exit ".

Router(config-line)#line console ?

% Unrecognized command, (нераспознанная команда)

Router(config-line)#exit

Router(config)#line console ?

<0-0> First Line number

Router(config)#line console 0

Router(config-line)#login

Router(config-line)#password todd1

Поскольку существует только один консольный порт, доступен вариант *line console 0*.

Пароль Telnet

Для установки пароля пользовательского режима при доступе по Telnet к маршрутизатору служит команда line vty. Маршрутизаторы, которые не исполняют версию Enterprise операционной системы Cisco IOS, по умолчанию имеют пять линий VTY (от 0 до 4). Однако в версии Enterprise таких линий намного больше — 198 (0 — 197). Проще всего узнать количество линий с помощью вопросительного знака.

Router(config-line)#line vty 0 ?
<1-197> Last Line Number *
<cr>
Router(config-line)#line vty 0 197
Router(config-line)#login
Router(config-line)# password todd 2

Если попытаться установить сеанс Telnet с маршрутизатором, не имеющим заданных паролей для линий VTY, то будет получено сообщение об ошибке "подключение прервано, поскольку не установлен пароль". Можно указать маршрутизатору на разрешение подключений по Telnet без пароля, использовав команду no login.

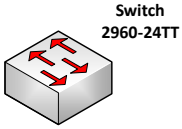
Router(config-line)#line vty 0 197
Router(config-line)#no login

1.3. Практическая часть

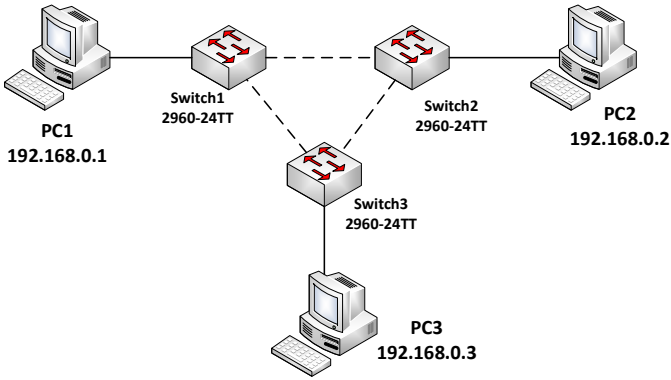
Данная лабораторная работа может быть выполнена на реальном оборудовании или в Cisco Packet Tracer. Все необходимые действия указаны по порядку их выполнения. Для начала выполнения лабораторной работы необходимо соединить физическую сеть в соответствии со схемой сети или построить соответствующий проект в Cisco Packet Tracer.

Используемая топология

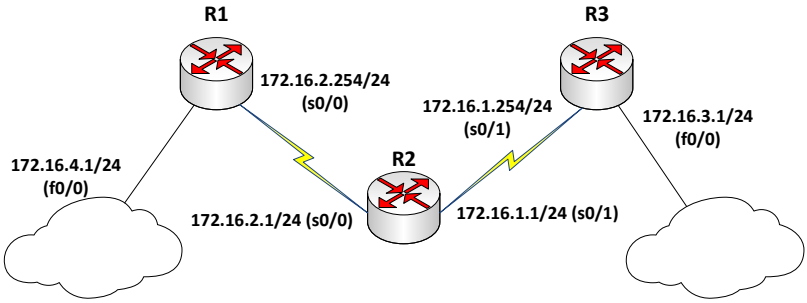
Для пункта 1



Для пункта 2



Для пункта 3



Порядок выполнения работы

1. Выполнить базовую конфигурацию отдельного коммутатора.

1.1. Задать hostname.
1.2. Отключить DNS lookup.
1.3. Установить пароль для EXEC mode.
1.4. Настроить message-of-the-day banner.
1.5. Установить пароль для console и vty.

Рассмотрим выполнение базовой конфигурации на одном из коммутаторов:

Switch>enable //входим в привилегированный режим.
Switch#configure terminal //входим в режим глобальной конфигурации.
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#hostname S1 //даём имя маршрутизатору, в данном случае S1.
S1(config)#enable secret class //включаем пароль на вход привилегированного режима.
S1(config)#no ip domain-lookup //выключаем поиск DNS
S1(config)#line console 0 //входим в режим настройки консоли.
S1(config-line)#password cisco //назначаем пароль на вход.
S1(config-line)#login //включаем запрос пароля перед входом в консоль.
S1(config-line)#line vty 0 15 //входим в режим настройки телнета.
S1(config-line)#password cisco //назначаем пароль на вход.
S1(config-line)#login //включаем запрос пароля перед входом в консоль.
S1(config-line)#end //выходим в привилегированный режим EXEC Mode.
%SYS-5-CONFIG_I: Configured from console by console
S1#copy running-config startup-config //сохраняем произведенную настройку в энергонезависимую память.
Destination filename [startup-config]?
Building configuration...
[OK]

2. **Выполнить конфигурирование сети простой топологии.**
   2.1. Выполнить пп. 1.1 – 1.5 для каждого устройства.
   2.2. Отключить неиспользуемые порты .
   2.3. Настроить активные порты в требуемый режим (access или trunk).
   2.4. Настроить сетевые интерфейсы PC1, PC2 и PC3 в соответствии с топологией.
3. **Выполнить базовую конфигурацию маршрутизаторов (см.п.2).**
   3.1. Провести начальную конфигурацию маршрутизаторов R1, R2, R3.
      3.1.1. Задать hostname.
      3.1.2. Отключить DNS lookup.
      3.1.3. Запретить вывод каких-либо консольных сообщений с помощью команды *logging synchronous* в режиме настройки *console 0*.
   3.2. Настроить интерфейсы маршрутизаторов R1, R2, R3.
      Рассмотреть настройку интерфейса на одном из коммутаторов (например, R1):
      R1
      R1(config)#interface FastEthernet0/0
      R1(config-if)#ip address 172.16.4.1 255.255.255.0
      R1(config-if)#no shutdown
      R1(config-if)#exit
      R1(config)#interface Serial 0/0
      R1(config-if)#clock rate 64000
      R1(config-if)#encapsulation ppp
      R1(config-if)#ip address 172.16.2.254 255.255.255.0
      R1(config-if)#no shutdown
      R1(config-if)#
   3.3. Добавить статические маршруты в таблицы маршрутизаторов R1, R2, R3.
      R1
      R1(config)#ip route 0.0.0.0 0.0.0.0 Serial 0/0
      R2
      R2(config)#ip route 172.16.4.0 255.255.255.0 Serial 0/0
      R2(config)#ip route 172.16.3.0 255.255.255.0 Serial 0/1
      R3
      R3(config)#ip route 0.0.0.0 0.0.0.0 Serial 0/0
   3.4. Проверить сделанные настройки **show ip route** и утилитой ping.
      R1>ping 172.16.3.1
      R3>ping 172.16.4.1

Содержание отчета

Отчет о лабораторной работе должен содержать перечень команд, а также результатов их выполнения для каждого из пунктов порядка выполнения.

ЛАБОРАТОРНАЯ РАБОТА № 2

Базовая настройка виртуальных локальных сетей VLAN

2.1. Цель работы

Целью работы является ознакомление с виртуальными локальными сетями (VLAN) и настройка VLAN на коммутаторе фирмы Cisco, а также закрепление полученных навыков по базовой настройке маршрутизатора и коммутатора.

2.2. Теоретическая часть

Виртуальная локальная сеть (**Virtual Local Area Netwrok, VLAN**) представляет собой коммутируемый сегмент сети, который логически выделен по выполняемым функциям, рабочим группам или приложениям, вне зависимости от физического расположения пользователей.

Виртуальные локальные сети обладают всеми свойствами физических локальных сетей, но рабочие станции можно группировать, даже если они физически расположены не в одном сегменте, так как любой порт коммутатора можно настроить на принадлежность определенной VLAN. При этом одноадресный, многоадресный и широковещательный трафики будут передаваться только между рабочими станциями, принадлежащими одной VLAN.

Каждая VLAN рассматривается как логическая сеть, т.е. пакеты, предназначенные станциям, которые не принадлежат данной VLAN, должны передаваться через маршрутизирующее устройство (маршрутизатор или коммутатор 3-го уровня). Таким образом, с помощью виртуальных сетей решается проблема ограничений при передаче широковещательных пакетов и вызываемых ими последствий, которые существенно снижают производительность сети, вызывают широковещательные штормы.

2.2.1. Типы VLAN

В коммутаторах могут быть реализованы следующие типы VLAN:

- на основе портов;
- на основе стандарта IEEE 802.1Q;
- на основе стандарта IEEE 802.1ad (Q-in-Q VLAN);
- на основе портов и протоколов IEEE 802.1v;
- на основе MAC-адресов;

- асимметричные.

2.2.2. Преимущества

Гибкое разделение устройств на группы

Как правило, одному VLAN соответствует одна подсеть. Устройства, находящиеся в разных VLAN, будут находиться в разных подсетях. Но в то же время VLAN не привязан к местоположению устройств и поэтому устройства, находящиеся на расстоянии друг от друга, все равно могут быть в одном VLAN независимо от местоположения.

Уменьшение количества широковещательного трафика в сети

Каждый VLAN — это отдельный широковещательный домен. Например, коммутатор — это устройство 2 уровня модели OSI. Все порты на коммутаторе с лишь одним VLAN находятся в одном широковещательном домене. Создание дополнительных VLAN на коммутаторе означает разбиение коммутатора на несколько широковещательных доменов. Если один и тот же VLAN настроен на разных коммутаторах, то порты разных коммутаторов будут образовывать один широковещательный домен.

Увеличение безопасности и управляемости сети

Когда сеть разбита на VLAN, упрощается задача применения политик и правил безопасности. С VLAN политики можно применять к целым подсетям, а не к отдельному устройству. Кроме того, переход из одного VLAN в другой предполагает прохождение через устройство 3 уровня, на котором, как правило, применяются политики, разрешающие или запрещающие доступ из VLAN в VLAN.

2.2.3. Технология виртуальных сетей (VLAN)

В технологии VLAN применяются различные способы настройки коммутаторов в зависимости от структуры исходной сети.

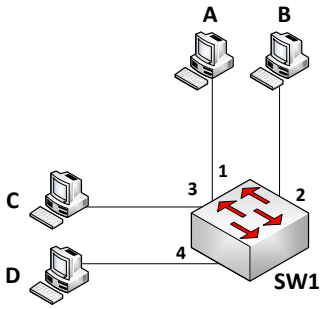
Организация VLAN в компьютерных сетях с одним-двумя коммутаторами

При организации VLAN на одном коммутаторе порты этого коммутатора распределяются между создаваемыми виртуальными сетями. В таблицу коммутации вводится дополнительный столбец для индексов VLAN_ID, определяющих принадлежность порта к определенной VLAN. Коммутируемые кадры могут передаваться только между портами, относящимися к одной виртуальной сети.

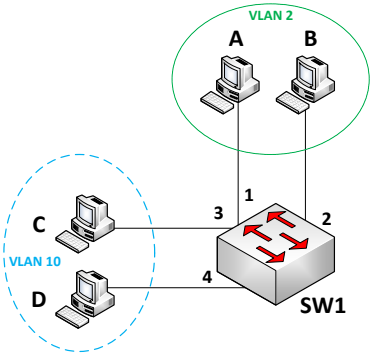
По существу, порты и подключенные к ним сетевые узлы, входящие в одну виртуальную локальную сеть, используют собственную таблицу коммутации. Такой метод организации виртуальных сетей называется методом группирования портов.

Группирование портов, как правило, осуществляется вручную сетевым администратором.

Организация VLAN методом группирования портов на одном коммутаторе показана на рис. 2.1.



а



б

Порт коммутатор	VLAN ID	MAC-адрес компьютера
1	1	A
2	1	B
3	1	C
4	1	D

в

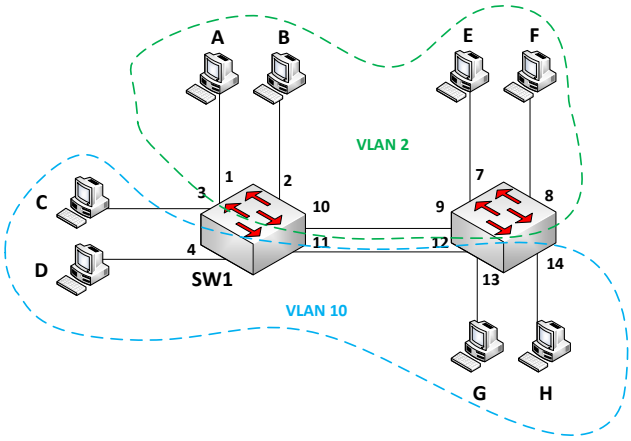
Порт коммутатор	VLAN ID	MAC-адрес компьютера
1	2	A
2	2	B
3	10	C
4	10	D

г

Рис. 2.1. Организация VLAN на одном коммутаторе:
а – схема исходной сети;
б – сеть, разделенная на VLAN;
в – таблица коммутации до формирования VLAN;
г – таблица коммутации с VLAN.

Для организации VLAN методом группирования портов в компьютерных сетях с несколькими коммутаторами необходимо связать коммутаторы, содержащие порты одной VLAN, соединительными линиями. При этом, чтобы устранить переходы трафика между виртуальными сетями, каждая виртуальная сеть должна иметь собственные соединительные линии.

Организация VLAN методом группирования портов на двух коммутаторах показана на рис. 2.2.



а

Таблица коммутации SW1

Порт коммутатор 1	VLAN ID	MAC-адрес компьютера
1	2	A
2	2	B
10	2	E
10	2	F
3	10	C
4	10	D
11	10	G
11	10	H

Таблица коммутации SW2

Порт коммутатор 2	VLAN ID	MAC-адрес компьютера
7	2	E
8	2	F
9	2	A
9	2	B
13	10	G
14	10	H
12	10	C
12	10	D

б

Рис. 2.2 Организация VLAN на двух коммутаторах методом группирования портов:
а – схема сети с двумя VLAN на двух коммутаторах;
б – таблицы коммутации SW1 и SW2.

Применение данного метода требует избыточных связей между коммутаторами. При увеличении количества коммутаторов в исходной сети быстро возрастает число соединительных линий и существенно увеличивается объем работ сетевого администратора по организации и изменении состава виртуальных сетей.

Поэтому группирование портов применяется для организации VLAN лишь в простых компьютерных сетях, использующих один-два коммутатора.

Организация VLAN в сложных компьютерных сетях

Для объединения в виртуальные сети узлов, подключенных к различным коммутаторам, существует более экономичный способ, основанный на стандарте IEEEE 802.1Q.

Все порты коммутаторов разделяются на две группы:

- **порты доступа** (англ. access), подключающие оконечные устройства (компьютеры, серверы, принтеры и пр.) к коммутатору;
- **порты линий связи** (англ. trunk), соединяющие коммутаторы между собой.

Порты доступа коммутаторов, как и ранее, распределяются по создаваемым виртуальным сетям. Получив от оконечного устройства кадр для передачи по сети Ethernet, оборудование портов доступа вводит в кадр специальные метки, свидетельствующие о принадлежности данного кадра к определенной виртуальной сети. Кадр с такой меткой называется «тегированным» - помеченным (англ. Tag – ярлык, метка). Внутри коммутатора передаются только тегированные кадры. Коммутатор продвигает кадр только между портами, имеющими общий тег, т.е. входящими в одну виртуальную сеть.

При передаче коммутированного кадра получателю информации на конечный сетевой узел порт доступа изымает из кадра ранее введенный тег, и сетевые пользователи получают исходные информационные кадры без каких-либо следов тегирования.

Порты доступа работают в режиме **access.**

В отличие от портов доступа порты, подключенные к линиям связи между коммутаторами, могут принимать и передавать кадры различных виртуальных сетей.

Наличие меток в передаваемых кадрах позволяет использовать общие соединительные линии между коммутаторами для передачи кадров нескольких виртуальных сетей при обеспечении изоляции трафика каждой сети.

Порты линий связи между коммутаторами работают в режиме **trunk.**

По умолчанию все конечные пользователи и порты коммутаторов относятся к исходной сети **VLAN1.**

При формировании виртуальных сетей на коммутаторах порты доступа вводятся в режим работы **access** и распределяются между различными виртуальными сетями, получая соответствующее значение идентификатора для каждого порта.

Порты линий связи между коммутаторами вводятся в режим работы **trunk**. Такие порты получают несколько значений идентификаторов, соответствующих виртуальным сетям, трафик которых должен передаваться по данной линии связи.

Организация VLAN на основе тегирования портов и использования общей линии связи между коммутаторами показана на рис. 2.3.

Порты 21 и 22 – транкинговые порты, имеющие идентификаторы VLAN_2 и VLAN_10.

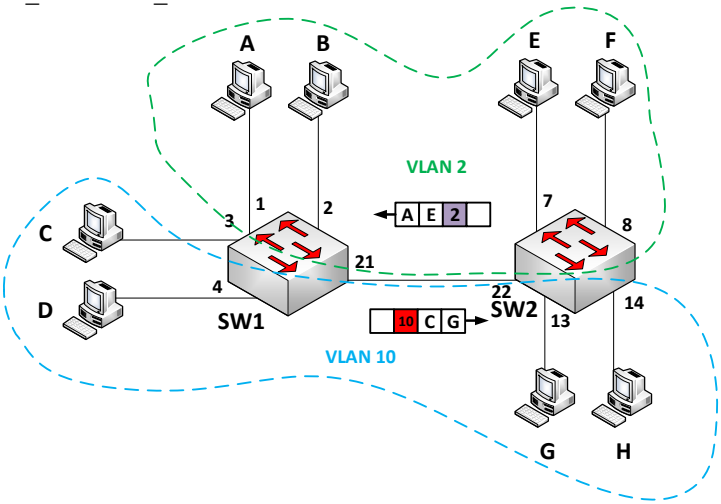


Рис. 2.3. Организация VLAN путем тегирования передаваемых кадров по стандарту IEEEE 802.1Q

В лабораторной работе предлагается создать VLAN на сети, состоящей из трех коммутаторов, порты которых работают в режимах *access* и *trunk*.

2.2.4. Настройка VLAN на коммутаторах Cisco под управлением IOS

Терминология Cisco:

- **access port** — порт, принадлежащий одному VLAN и передающий нетегированный трафик;
- **trunk port** — порт, передающий тегированный трафик одного или нескольких VLAN.

Коммутаторы Cisco ранее поддерживали два протокола **802.1Q** и **ISL**. **ISL** — проприетарный протокол, использующийся в оборудовании Cisco. ISL, полностью инкапсулирует фрейм для передачи информации о принадлежности к VLAN.

В современных моделях коммутаторов Cisco ISL не поддерживается.

Рассмотрим частичную настройку VLAN на примере простой топологии (см. рис. 2.4).

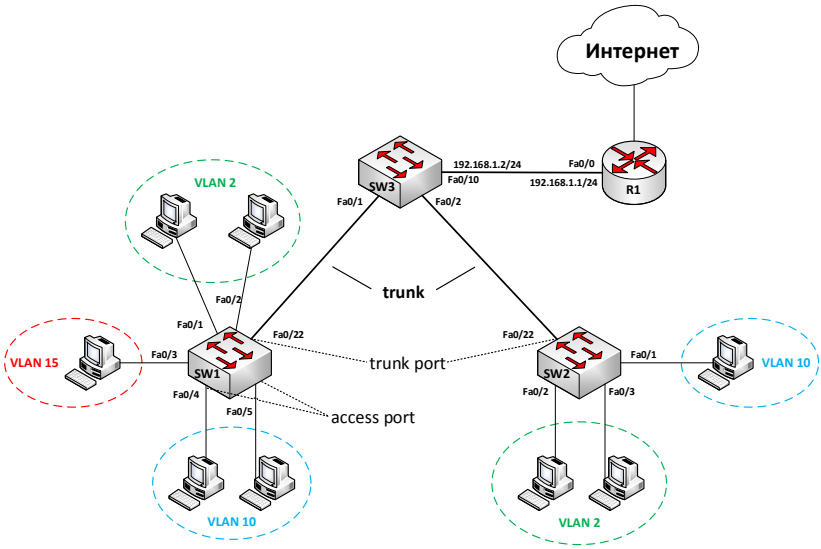


Рис. 2.4. Пример простой топологии

Создание VLAN с идентификатором 2 и задание имени для него:

sw1(config)# vlan 2

sw1(config-vlan)# name test

Удаление VLAN с идентификатором 2:

sw1(config)# no vlan 2

Настройка access портов

Назначение порта коммутатора в VLAN:

sw1(config)# interface fa0/1

sw1(config-if)# switchport mode access // включение режима access

sw1(config-if)# switchport access vlan 2

Назначение диапазона портов с fa0/4 до fa0/5 в vlan 10:

sw1(config)# interface range fa0/4 - 5

sw1(config-if-range)# switchport mode access

sw1(config-if-range)# switchport access vlan 10

Просмотр информации о VLAN:

sw1# show vlan brief

```
VLAN Name                             Status    Ports
---------------------------------- --------- -------------------------------
1    default                          active    Fa0/6, Fa0/7, Fa0/8, Fa0/9,
                                                Fa0/10, Fa0/11, Fa0/12, Fa0/13,
                                                Fa0/14, Fa0/15, Fa0/16, Fa0/17,
```

```
                                    Fa0/18, Fa0/19, Fa0/20, Fa0/21,
                                    Fa0/22, Fa0/23, Fa0/24
2    test                 active    Fa0/1, Fa0/2
10   VLAN0010             active    Fa0/4, Fa0/5
15   VLAN0015             active    Fa0/3
```

Настройка транка (trunk)

Для того чтобы передать через порт трафик нескольких VLAN, порт переводится в режим транка.

Режимы интерфейса (режим по умолчанию зависит от модели коммутатора):

- ***auto*** — порт находится в автоматическом режиме и будет переведён в состояние trunk, только если порт на другом конце находится в режиме on или desirable, т.е. если порты на обоих концах находятся в режиме "auto", то trunk применяться не будет;
- ***desirable*** — порт находится в режиме "готов перейти в состояние trunk"; периодически передает DTP-кадры порту на другом конце, запрашивая удаленный порт перейти в состояние trunk (состояние trunk будет установлено, если порт на другом конце находится в режиме on, desirable или auto);
- ***trunk*** — порт постоянно находится в состоянии trunk, даже если порт на другом конце не поддерживает этот режим.
- ***nonegotiate*** — порт готов перейти в режим trunk, но при этом не передает DTP-кадры порту на другом конце. Этот режим используется для предотвращения конфликтов с другим "не-cisco" оборудованием. В этом случае коммутатор на другом конце должен быть вручную настроен на использование trunk'а.

По умолчанию в транке разрешены все VLAN. Для того чтобы через соответствующий VLAN в транке передавались данные как минимум необходимо, чтобы VLAN был активным. Активным VLAN становится тогда, когда он создан на коммутаторе и в нём есть хотя бы один порт в состоянии up/up.

VLAN можно создать на коммутаторе с помощью команды vlan. Кроме того, VLAN автоматически создается на коммутаторе в момент добавления в него интерфейсов в режиме access.

В схеме, которая используется для демонстрации настроек, на коммутаторах sw1 и sw2, нужные VLAN будут созданы в момент добавления access-портов в соответствующие VLAN:

```
sw1(config)# interface fa0/3
sw1(config-if)# switchport mode access
sw1(config-if)# switchport access vlan 15
% Access VLAN does not exist. Creating vlan 15.
```

На коммутаторе sw3 access-портов нет. Поэтому необходимо явно создать все необходимые VLAN:

sw3(config)# vlan 2,10,15

Для автоматического создания VLAN на коммутаторах может использоваться протокол VTP.

Настройка статического транка

Создание статического транка:

sw1(config)# interface fa0/22

sw1(config-if)# switchport mode trunk

На некоторых моделях коммутаторов (на которых поддерживается ISL) после попытки перевести интерфейс в режим статического транка может появиться такая ошибка:

sw1(config-if)# switchport mode trunk

Command rejected: An interface whose trunk encapsulation is "Auto" can not be configured to "trunk" mode.

Это происходит из-за того, что динамическое определение инкапсуляции (ISL или 802.1Q) работает только с динамическими режимами транка. И для того, чтобы настроить статический транк, необходимо инкапсуляцию также настроить статически.

Для таких коммутаторов необходимо явно указать тип инкапсуляции для интерфейса:

sw1(config-if)# switchport trunk encapsulation dot1q

И после этого снова повторить команду настройки статического транка (switchport mode trunk).

Разрешённые VLAN

По умолчанию в транке разрешены все VLAN. Можно ограничить перечень VLAN, которые могут передаваться через конкретный транк.

Указать перечень разрешенных VLAN для транкового порта fa0/22:

sw1(config)# interface fa0/22

sw1(config-if)# switchport trunk allowed vlan 1-2,10,15

Добавление ещё одного разрешенного VLAN:

sw1(config)# interface fa0/22

sw1(config-if)# switchport trunk allowed vlan add 160

Удаление VLAN из списка разрешенных:

sw1(config)# interface fa0/22

sw1(config-if)# switchport trunk allowed vlan remove 160

Native VLAN

В стандарте 802.1Q существует понятие native VLAN. Трафик этого VLAN передается нетегированным. По умолчанию это VLAN 1. Однако можно изменить это и указать другой VLAN как native.

Настройка VLAN 5 как native:

sw1(config-if)# switchport trunk native vlan 5

Теперь весь трафик, принадлежащий VLAN 5, будет передаваться через транковый интерфейс нетегированным, а весь пришедший на транковый интерфейс нетегированный трафик будет промаркирован как принадлежащий VLAN 5 (по умолчанию VLAN 1).

Настройка маршрутизации между VLAN

Все настройки по назначению портов в VLAN, сделанные ранее для sw1, sw2 и sw3, сохраняются. Дальнейшие настройки подразумевают использование sw3 как коммутатора 3-го уровня.

При такой схеме работы никаких дополнительных настроек на маршрутизаторе не требуется. Коммутатор осуществляет маршрутизацию между сетями разных VLAN, а на маршрутизатор отправляет трафик, предназначенный в другие сети.

Настройки на коммутаторе sw3:

VLAN / интерфейс 3-го уровня	IP-адрес
VLAN 2	10.0.2.1 /24
VLAN 10	10.0.10.1 /24
VLAN 15	10.0.15.1 /24
Fa 0/10	192.168.1.2 /24

Включение маршрутизации на коммутаторе:

sw3(config)# ip routing

Задание адреса в VLAN. Этот адрес будет маршрутом по умолчанию для компьютеров в VLAN 2:

sw3(config)# interface Vlan2
sw3(config-if)# ip address 10.0.2.1 255.255.255.0
sw3(config-if)# no shutdown

Задание адреса в VLAN 10:

sw3(config)# interface Vlan10
sw3(config-if)# ip address 10.0.10.1 255.255.255.0
sw3(config-if)# no shutdown

Перевод интерфейса в режим 3-го уровня

Интерфейс fa0/10 соединен с маршрутизатором. Этот интерфейс можно перевести в режим 3-го уровня.

Перевод fa0/10 в режим интерфейса 3-го уровня и задание IP-адреса:

sw3(config)#interface FastEthernet 0/10
sw3(config-if)# no switchport
sw3(config-if)# ip address 192.168.1.2 255.255.255.0
sw3(config-if)# no shutdown

Коммутатор sw3 использует R1 как шлюз по умолчанию для рассматриваемой сети. Трафик, не предназначенный сетям VLAN будет передаваться на R1.

Настройка маршрута по умолчанию:

sw3(config) ip route 0.0.0.0 0.0.0.0 192.168.1.1

Просмотр информации

Просмотр информации о VLAN:

sw1# show vlan brief

```
VLAN Name        Status          Ports
---- ----------  ------  -------------------------
1    default     active  Fa0/6, Fa0/7, Fa0/8, Fa0/9,
                         Fa0/10, Fa0/11, Fa0/12, Fa0/13,
                         Fa0/14, Fa0/15, Fa0/16, Fa0/17,
                         Fa0/18, Fa0/19, Fa0/20, Fa0/21,
                         Fa0/22, Fa0/23, Fa0/24
2    test        active  Fa0/1, Fa0/2
10   VLAN0010    active  Fa0/4, Fa0/5
15   VLAN0015    active  Fa0/3
```

Настройка port security

Настройка port security:

switch(config)# port-security <port-list>
[learn-mode < continuous | static | port-access | configured | limited-continuous>]
[action < none | send-alarm | send-disable >]
[address-limit <1-8 | 1-32>
[mac-address <mac-addr1 [mac-addr2]...>]
[clear-intrusion-flag]

Параметры команды port-security:

- learn-mode — режим запоминания MAC-адресов. По умолчанию все порты в режиме continuous;
- action — действие, которое будет выполняться при нарушении:
- none — не выполнять никаких действий;
- send-alarm — отправить сообщение о нарушении (SNMP, log);
- send-disable — отправить сообщение о нарушении и выключить порт;
- address-limit — максимальное количество MAC-адресов, которое будет разрешено на порту:
  - Применяется к режимам static, configured и limited-continuous;
  - Для static и configured значения от 1 до 8;
  - Для limited-continuous — от 1 до 32;
  - По умолчанию для всех режимов разрешен 1 MAC-адрес;
- mac-address — статическое задание разрешенных MAC-адресов для режимов static и configured;

- clear-intrusion-flag — очистить intrusion flag для указанных портов. После этого можно включать порт, который выключился из-за нарушения port security.

Отмена настройки port security

Отмена настройки port security:

switch(config)# no port-security <port>

Безопасные MAC-адреса

Коммутатор поддерживает такие типы безопасных MAC-адресов:

- **статические MAC-адреса:**
  - задаются статически командой *switchport port-security mac-address mac-address* в режиме настройки интерфейса;
  - хранятся в таблице адресов;
  - добавляются в текущую конфигурацию коммутатора;
- **динамические MAC-адреса:**
  - динамически выучиваются;
  - хранятся только в таблице адресов;
  - удаляются при перезагрузке коммутатора;
- **sticky MAC-адреса:**
  - могут быть статически настроены или динамически выучены;
  - хранятся в таблице адресов;
  - добавляются в текущую конфигурацию коммутатора.

Если эти адреса сохранены в конфигурационном файле, после перезагрузки коммутатора их не надо заново перенастраивать.

На коммутаторах Cisco такие настройки по умолчанию для функции **port security**:

- port security — выключен;
- запоминание sticky-адресов — выключено;
- максимальное количество безопасных MAC-адресов на порту — 1;
- режим реагирования на нарушения — shutdown;
- время хранения адресов:
  - отключено. Значение aging time — 0;
  - для статических адресов — отключено;
  - тип времени — абсолютное.

Port security настраивается в режиме настройки интерфейса. На многих коммутаторах Cisco по умолчанию порт находится в режиме **dynamic auto**, однако этот режим не совместим с функцией **port security**. Поэтому интерфейс надо перевести в режим **trunk** или **access**:

switch(config-if)# switchport mode <access | trunk>

Включение port security на интерфейсе (после этого включены настройки по умолчанию):

switch(config-if)# switchport port-security

Максимальное количество безопасных MAC-адресов

Максимальное количество безопасных MAC-адресов на интерфейсе или в VLAN:

switch(config-if)# switchport port-security maximum <value> [vlan <vlan-list>].

Например, на интерфейсе разрешить 2 MAC-адреса, а остальные настройки по умолчанию:

switch(config)# interface Fastethernet0/3

switch(config-if)# switchport mode access

switch(config-if)# switchport port-security

switch(config-if)# switchport port-security maximum 2.

Заданием опционального параметра *vlan*, при указании максимального количества безопасных MAC-адресов, можно ограничить количество MAC-адресов для VLAN или перечня VLAN (добавлено с версии IOS 12.2.37SE).

Например, настроить транк и разрешить 20 MAC-адресов в VLAN7:

switch(config)# interface Fastethernet0/3

switch(config-if)# switchport mode trunk

switch(config-if)# switchport port-security

switch(config-if)# switchport port-security maximum 20 vlan 7.

Если на интерфейсе fa0/3 возникнет нарушение безопасности в VLAN 7, например появится 21 адрес, то заблокирован будет только трафик этого VLAN.

Просмотр информации о настройках port-security для VLAN 7:

switch# show port-security vlan 7

Настройка безопасных MAC-адресов

Включение sticky запоминания адресов:

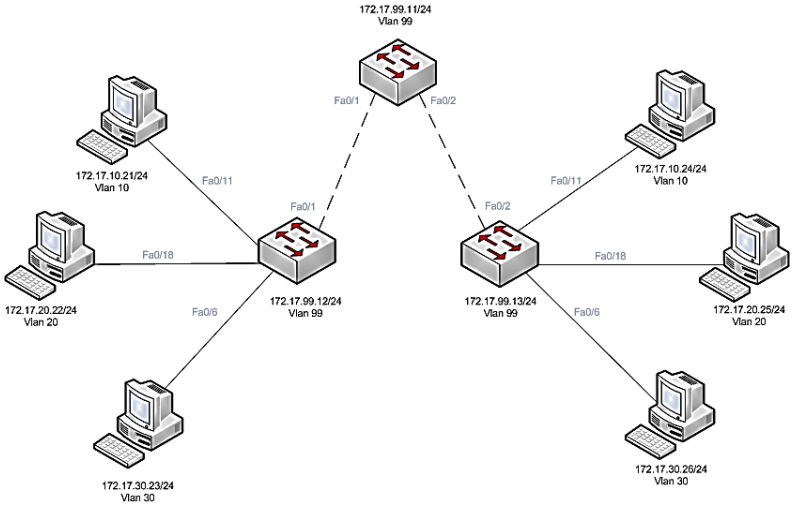
switch(config-if)# switchport port-security mac-address sticky

switch(config-if)# switchport port-security mac-address sticky [mac-address | vlan <vlan-id | <access | voice>>]

2.3. Практическая часть

Для начала выполнения лабораторной работы необходимо соединить физическую сеть в соответствии со схемой сети или построить соответствующий проект в Cisco Packet Tracer. Сразу после схемы сети в таблице указан план адресации, который нужно применять только тогда, когда это будет явно указано в тексте лабораторной работы.

Используемая топология



План адресации

Device	Interface	IP Address	Subnet Mask	Default Gateway
S1	VLAN 99	172.17.99.11	255.255.255.0	N/A
S2	VLAN 99	172.17.99.12	255.255.255.0	N/A
S3	VLAN 99	172.17.99.13	255.255.255.0	N/A
PC1	NIC	172.17.10.21	255.255.255.0	172.17.10.1
PC2	NIC	172.17.20.22	255.255.255.0	172.17.20.1
PC3	NIC	172.17.30.23	255.255.255.0	172.17.30.1
PC4	NIC	172.17.10.24	255.255.255.0	172.17.10.1
PC5	NIC	172.17.20.25	255.255.255.0	172.17.20.1
PC6	NIC	172.17.30.26	255.255.255.0	172.17.30.1

Назначения портов (S1, S2 и S3)

Port	Assignment	Network
Fa0/1 – 0/5	802.1q Trunks (Native VLAN 99) management	172.17.99.0 /24
Fa0/6 – 0/10	VLAN 30 – Red	172.17.30.0 /24
Fa0/11 – 0/17	VLAN 10 – Green	172.17.10.0 /24
Fa0/18 – 0/24	VLAN 20 – Amber	172.17.20.0 /24

Порядок выполнения работы

1. **Базовая конфигурация оборудования.**
   - Настроить hostname.
   - Отключить DNS lookup.
   - Установить пароль для EXEC mode
   - Настроить message-of-the-day banner.
   - Установить пароль для console
2. **Активировать пользовательские порты на S2 и S3.**

Настроить пользовательские порты в режиме access в соответствии со схемой.

3. **Настроить Ethernet интерфейсы персональных компьютеров.**

Настроить сетевые интерфейсы PC1, PC2, PC3, PC4, PC5 и PC6 в соответствии с планом адресации.

4. **Настроить VLAN на коммутаторах.**
   - На каждом из коммутаторов (**S1**, **S2** и **S3**) настроить VLAN.
   - Проверить выполненные действия на каждом из коммутаторов командой **show vlan brief.**
   - На коммутаторах S1, S2 и S3 назначить порты соответствующим VLAN согласно таблице назначения портов.
   - Проверить выполненные настройки командой **show vlan id *vlan-number***.
   - Настроить IP-адреса для интерфейсов управления на **S1**, **S2** и **S3**.
   - На каждом из коммутаторов настроить **trunk**-порты и **native VLAN.**
   - Проверить выполненные действия командой **show interface trunk.**
5. **Проверить наличие связи в сети.**
   - Из консоли коммутатора **S1** выполнить команду **ping** для интерфейсов управления **S2** и **S3**.

Содержание отчета

Отчет о лабораторной работе должен содержать перечень команд, а также результатов их выполнения для каждого из пунктов порядка выполнения.

ЛАБОРАТОРНАЯ РАБОТА № 3

Базовая настройка VTP

3.1. Цель работы

Цель данной работы - ознакомить с протоколом VTP и закрепить полученные навыки в предыдущих работах.

3.2. Теоретическая часть

VLAN Trunking Protocol (VTP) — проприетарный протокол компании Cisco Systems, предназначенный для создания, удаления и переименования VLAN на сетевых устройствах. Передавать информацию о том, какой порт находится в каком VLAN, он не может.

Протокол VTP был создан для решения возможных проблем в среде коммутации виртуальных локальных сетей VLAN. Например, рассмотрим домен, в котором имеются несколько связанных друг с другом коммутаторов, которые поддерживают несколько VLAN-сетей. Для создания и поддержки соединений внутри VLAN-сетей каждая из них должна быть сконфигурирована вручную на каждом коммутаторе. По мере роста организации и увеличения количества коммутаторов в сети каждый новый коммутатор должен быть сконфигурирован вручную с вводом информации о VLAN-сетях. Всего лишь одно неправильное назначение в сети VLAN может вызвать две потенциальных проблемы.

- Перекрестное соединение VLAN-сетей вследствие несогласованности в конфигурации VLAN-сетей.
- Согласование и ликвидация противоречивости конфигураций в смешанной среде передачи, например в среде, включающей в себя сегменты Ethernet и Fiber Distributed Data Interface (FDDT).

В протоколе VTP согласованность конфигураций VLAN-сетей поддерживается в общем административном домене. Кроме того, протокол VTP уменьшает сложность управления и мониторинга VLAN-сетей.

3.2.1. Общие положения протокола VTP

Назначение протокола VTP состоит в поддержке согласованности конфигураций в общем административном сетевом домене. Протокол VTP является протоколом обмена сообщениями, использующим магистральные фреймы 2-го уровня для управления добавлением, удалением и переименованием VLAN-сетей в одном домене.

Кроме того, протокол VTP позволяет осуществлять централизованные изменения в сети, о которых сообщается всем другим коммутаторам сети.

Сообщения протокола VTP инкапсулируются в фирменные фреймы протоколов ISL или IEEE 802.lQ и передаются далее по магистральным каналам другим устройствам. К фреймам IEEE 802.1 Q

в качестве тега добавляется 4-байтовое поле. В обоих форматах передается идентификатор ID VLAN-сети. В то время как порты коммутаторов обычно назначаются только одной VLAN сети, магистральные порты, по умолчанию, передают фреймы всем VLAN-сетям.

3.2.2. Преимущества использования протокола VTP

Протокол VTP позволяет свести к минимуму возможную рассогласованность конфигураций, которая возникает при изменении топологии сети. Эта несогласованность может привести к снижению защищённости сети, поскольку перекрестное соединение VLAN-сетей при использовании дублирующих имен может привести к внутреннему разъединению при преобразовании от одного типа VLAN к другому (например, от Ethernet к ATM или FDDI).

Использование протокола VTP предоставляет следующие преимущества.

• Поддержка согласованности конфигураций VLAN-сетей во всей объединенной сети.

• Поддержка схемы преобразования, которая позволяет VLAN-сети осуществлять магистральное соединение по смешанной среде, например преобразование VLAN-сети в высокоскоростную магистральную VLAN-сеть LANE ATM или FDDI.

• Точное отслеживание и мониторинг VLAN-сетей.

• Динамическое оповещение всех устройств сети о добавлении новой VLAN-сети.

• Конфигурирование режима "Plug-and-play" при добавлении новой VLAN-сети.

Перед созданием на коммутаторе VLAN-сети необходимо сначала создать домен управления протокола VTP, в котором можно протестировать созданную VLAN-сеть.

Все коммутаторы в одном и том же домене управления, которые совместно используют информацию о VLAN-сетях. Коммутатор может присутствовать только в одном домене управления протокола VTP. Находящиеся в разных доменах коммутаторы не могут совместно использовать информацию протокола VTP.

В протоколе VTP каждый коммутатор семейства Catalyst передает со своих магистральных портов следующую информацию:

- домен управления;
- номер версии конфигурации;
- известные VLAN-сети и их конкретные параметры.

3.2.3. Режимы протокола VTP

Протокол VTP может работать в одном из трех режимов:

1) **VTP Server** — серверный режим. В этом режиме можно создавать, удалять и изменять VLAN, а также задавать различные

параметры, такие как версию протокола (vtp version), vtp фильтрацию (vtp pruning) для всего VTP домена. VTP сервер извещает о своей конфигурации VLAN другие коммутаторы, находящиеся в том же VTP домене, и синхронизирует их конфигурацию VLAN. Так же VTP сервер может синхронизировать свою конфигурацию с конфигурацией VTP клиента, если у клиента выше номер ревизии конфигурации. Обмен VTP информацией происходит через транковые порты;

2) **VTP Client** — режим клиента. На коммутаторе с режимом VTP Client нельзя создавать, удалять или изменять VLAN. Всю настройку VLAN коммутатор берёт от VTP сервера;

3) **VTP Transparent** — прозрачный режим. В этом режиме коммутатор не применяет к себе конфигурацию VLAN от VTP сервера и не извещает о своей конфигурации другие коммутаторы, но позволяет пропускать через свои транковые порты VTP извещения от других коммутаторов.

Возможности

Server (режим по умолчанию):

- можно создавать, изменять и удалять VLAN из командной строки коммутатора;
- генерирует объявления VTP и передает объявления от других коммутаторов;
- может обновлять свою базу данных VLAN при получении информации не только от других VTP серверов, но и от других VTP клиентов в одном домене, с более высоким номером ревизии;
- сохраняет информацию о настройках VLAN в файле vlan.dat во flash.

Client:

- нельзя создавать, изменять и удалять VLAN из командной строки коммутатора;
- передает объявления от других коммутаторов;
- синхронизирует свою базу данных VLAN при получении информации VTP;
- сохраняет информацию о настройках VLAN в файле vlan.dat во flash.

Transparent:

- можно создавать, изменять и удалять VLAN из командной строки коммутатора, но только для локального коммутатора;
- не генерирует объявления VTP;
- передает объявления от других коммутаторов;

- не обновляет свою базу данных VLAN при получении информации по VTP;
- сохраняет информацию о настройках VLAN в NVRAM;
- всегда использует configuration revision number 0.

В версии 3 VTP добавился новый режим работы и изменились некоторые режимы работы по сравнению с предыдущими версиями:

Server:

- добавилась поддержка Private VLAN;
- могут анонсироваться VLAN из расширенного диапазона.

Client:

- настройки VLAN сохраняются в NVRAM и в режиме клиента;
- добавилась поддержка Private VLAN;
- могут анонсироваться VLAN из расширенного диапазона.

Transparent:

- без изменений.

Off:

- новый режим работы VTP, который добавился в 3 версии;
- не передает объявления VTP;
- в остальном аналогичен режиму Transparent.

3.2.4. Диапазоны VLAN

Диапазоны VLAN

VTP версии 1 и 2 не анонсирует эти VLAN, они не могут быть исключены (pruned):

1 — нормальный (normal) диапазон;

1002-1005 — нормальный (normal) диапазон;

1006-4094 — расширенный (extended) диапазон;

VTP версии 1 и 2 анонсирует эти VLAN, они могут быть исключены (pruned);

2-1001 — нормальный (normal) диапазон.

3.2.5. Перед настройкой VTP нужно знать следующее.

Все коммутаторы в сети должны иметь одно и то же имя домена VTP (VTP domain).

Все коммутаторы в одном VTP домене должны использовать одну и ту же версию протокола VTP (VTP version).

Все коммутаторы в одном VTP домене должны использовать один и тот же VTP пароль, если он установлен (VTP password).

Все VTP сервера должны иметь одинаковый номер ревизии конфигурации, и этот номер должн быть наибольшим в VTP домене (VTP revision number).

При смене режима Transparent на Server VLAN, которые были на коммутаторе в прозрачном режиме, должны существовать на VTP сервере.

В версии протокола 1 и 2 анонсируются VLAN только из основного (VLAN 1-1005) диапазона, если нужен расширенный диапазон (VLAN 1006-4094), то следует использовать версию VTP 3.

3.2.6. Базовая настройка протокола VTP

Первым делом – нарисуйте топологию сети, в которой собираетесь применять протокол VTP. Посмотрите, какие версии протокола поддерживаются устройствами (обычно везде есть VTPv2). Выберите устройство, которое будет сервером (ему не надо быть каким-то особо быстрым, это не STP, специфической нагрузки на VTP Server нет, ему лишь желательно обладать максимальным uptime). Если не хотите использовать VTP (например, из соображений безопасности), тогда просто переведите все устройства в режим VTP Transparent (либо off, если поддерживается оборудованием и ОС).

Настройка имени домена VTP

host(config)#vtp domain имя_домена

Стереть имя домена штатно нельзя, только сменить, т.е. если стартово заменили дефолтное значение, которое NULL, на своё, то всё.

Примечание: ну, конечно, не “всё” совсем. Перезагрузите устройство, зайдите в rommon, сотрите файл vlan.dat, и всё ОК – настройки VTP у держателя роли VTP Server хранятся там, поэтому он всё сразу “забудет”.

Настройка пароля VTP

host(config)#vtp password пароль

Пароль можно сбросить на пустой, если ввести команду *no vtp password*. Запомните, пароль VTP хранится небезопасно (у VTP Server– в файле vlan.dat, у VTP Transparent – в NVRAM), поэтому если пользуетесь VTP, делайте такой пароль, который более нигде не дублируется, так как получить пароль VTP – относительно несложно. Всё, от чего защищает этот пароль, – это, например, случайное добавление в сеть неправильно настроенного коммутатора и последующие проблемы. Пароль VTP не защищает передаваемую между коммутаторами информацию.

Настройка версии VTP

host(config)#vtp version версия

На момент написания версии были от 1-й до 3-й.

Настройка режима VTP

host(config)#vtp mode режим

Где режим – это ***server***, ***client***, ***transparent*** или ***off***. Режим ***off*** получится поставить только на устройствах,

поддерживающих VTPv3; на коммутаторах, которые поддерживают только VTPv1 и VTPv2, отключить протокол нельзя.

Устранение неисправностей (troubleshooting) протокола VTP

Неисправностей в VTP может быть очень много. Давайте перечислим основные из них. Запомните эти “лица”, может пригодиться в дальнейшем.

Проверяем каналы между коммутаторами

- Проверьте физическую доступность интерфейсов.
- Проверьте корректность режима дуплекса и скорости.
- Проверьте, что корректно согласовался транк.
- Проверьте, совпадают ли native vlan.

Проверяем настройки VTP

- Участвующие коммутаторы должны быть непосредственно подключены друг к другу.
- Должен быть хотя бы один VTP Server.
- Версии VTP, а также имя домена и пароль должны быть идентичны у всех устройств.

VTP Pruning

VTP Pruning – это технология, придуманная для снижения нагрузки на магистральные каналы (транки). Главное в том, что нет необходимости отправлять коммутатору информацию о VLAN, если у него нет активных портов в этом VLAN. Так же блокирует броткаст – не отправляет широковещательные пакеты на коммутатор если у него нет активных портов в том VLAN, что и отправитель бродкаста.

Включается командой:

Switch(config)#set vtp pruning enable.

VTP pruning по умолчанию отключен.

Без включенного VTP pruning коммутатор рассылает широковещательный трафик по всему VTP домену.

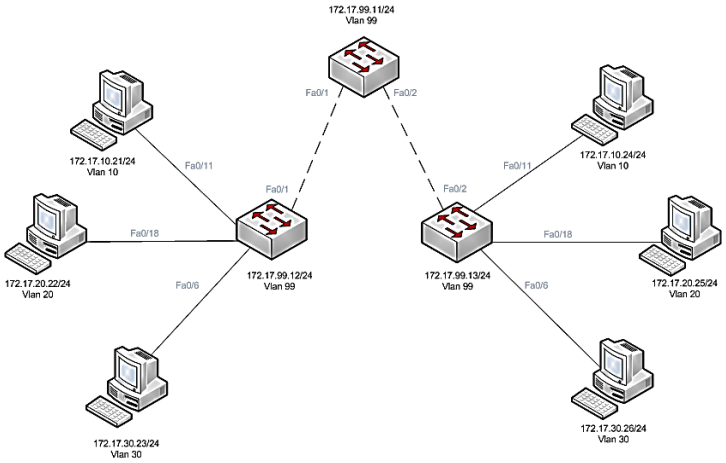
Основной задачей же VTP pruning является обрезание broadcast, multicast и неопознанного unicast трафика в рамках одного VLAN на магистральных интерфейсах (trunk links). Это сделано для того, чтобы пресечь распространение широковещательных передач на те коммутаторы, где они не нужны. Или, попросту говоря, не стоит передавать лишний трафик в те места, где его никто не услышит, и, как следствие, высвобождаем занятую этим трафиком пропускную полосу для более полезных задач.

3.3. Практическая часть

Для начала выполнения лабораторной работы необходимо соединить физическую сеть в соответствии со схемой сети или построить соответствующий проект в Cisco Packet Tracer. Сразу после схемы сети в таблице указан план адресации, который нужно

применять только тогда, когда это будет явно указано в тексте лабораторной работы.

Используемая топология



План адресации

Device	Interface	IP Address	Subnet Mask	Default Gateway
S1	VLAN 99	172.17.99.11	255.255.255.0	N/A
S2	VLAN 99	172.17.99.12	255.255.255.0	N/A
S3	VLAN 99	172.17.99.13	255.255.255.0	N/A
PC1	NIC	172.17.10.21	255.255.255.0	172.17.10.1
PC2	NIC	172.17.20.22	255.255.255.0	172.17.20.1
PC3	NIC	172.17.30.23	255.255.255.0	172.17.30.1
PC4	NIC	172.17.10.24	255.255.255.0	172.17.10.1
PC5	NIC	172.17.20.25	255.255.255.0	172.17.20.1
PC6	NIC	172.17.30.26	255.255.255.0	172.17.30.1

Назначения портов (S1, S2 и S3)

Port	Assignment	Network
Fa0/1 – 0/5	802.1q Trunks (Native VLAN 99) management	172.17.99.0 /24
Fa0/6 – 0/10	VLAN 30 – Red	172.17.30.0 /24
Fa0/11 – 0/17	VLAN 10 – Green	172.17.10.0 /24
Fa0/18 – 0/24	VLAN 20 – Amber	172.17.20.0 /24

Порядок выполнения работы

1. **Базовая конфигурация оборудования**
   1.1. Настроить hostname.
   1.2. Отключить DNS lookup.
   1.3. Установить пароль для EXEC mode.
   1.4. Настроить message-of-the-day banner.
   1.5. Установить пароль для console.
   1.6. Активировать пользовательские порты на S2 и S3.
   1.7. Настроить пользовательские порты в режиме **access** в соответствии со схемой.
   1.8. Активировать trunk -порты на S1, S2 и S3.
2. **Настроить Ethernet интерфейсы персональных компьютеров.**
   2.1. Настроить сетевые интерфейсы PC1, PC2, PC3, PC4, PC5 и PC6 в соответствии с планом адресации.
   2.2. Используя команду ping, убедиться в наличии связи между PC1 и PC4, PC2 и PC5, а также между PC3 и PC6.
3. **Настроить VTP на свитчах.**
   3.1. Используя команду **show vtp status**, проверить текущие настройки коммутаторов.
   3.2. Коммутаторы S1, S2 и S3 находятся в режиме Server. Это является настройками по умолчанию для большинства коммутаторов Catalyst.
   3.3. Настроить **operating mode**, **domain name**, и **VTP password** на всех коммутаторах.

Коммутатор	VTP mode	Domain name	VTP password
S1	Server	VtpLab	Cisco
S2	Client	VtpLab	Cisco
S3	Transparent	VtpLab	Cisco

   3.4. Сконфигурировать **native VLAN** для trunk портов на всех коммутаторах.
   3.5. Для упрощения процесса необходимо использовать команду interface range.
   3.6. Настроить port security для access layer коммутаторов S2 и S3.
   3.7. Для портов fa0/6, fa0/11 и fa0/18 установить ограничение в одно подключение. Аппаратный адрес подключенного интерфейса должен быть получен динамически.
   3.8. Настроить VLAN на VTP сервере:
   VLAN 99 (management)
   VLAN 10 (Green)

VLAN 20 (Amber)
VLAN 30 (Red).

3.9. Проверить распространение созданных VLAN на S1, S2 и S3.
3.10. Используя команду **show vlan brief** на **S2** и **S3**, убедится в том, что созданные на VTP-сервере VLAN-ы были распределены на все коммутаторы.
3.11. Создать новый VLAN 88 на S2 и S3.
3.12. Удалить VLAN 88 на S3.
3.13. Настроить VLAN на S3:
VLAN 99 (management)
VLAN 10 (Green)
VLAN 20 (Amber)
VLAN 30 (Red).
3.14. Настроить IP-адреса для интерфейсов управления на S1, S2 и S3.
3.15. Назначить порты соответствующим VLAN согласно таблице назначения портов.

4. Настроить VTP Pruning

4.1. VTP Pruning настраивается в global configuration mode командой vtp pruning для коммутаторов в Server VTP Operating Mode.

Содержание отчета

Отчет о лабораторной работе должен содержать перечень команд, а также результатов их выполнения для каждого из пунктов порядка выполнения.

ЛАБОРАТОРНАЯ РАБОТА № 4

Базовая настройка протокола STP

4.1. Цель работы

Цель работы - ознакомить с протоколом связующего дерева STP и закрепить полученные навыки о VLAN.

4.2. Теоретическая часть

STP (Spanning Tree Protocol) — сетевой протокол (или семейство сетевых протоколов), предназначенный для автоматического удаления циклов (петель коммутации) из топологии сети на канальном уровне в Ethernet-сетях. Первоначальный протокол STP описан в стандарте 802.1D. Позже появилось несколько новых протоколов (RSTP, MSTP, PVST, PVST+), отличающихся некоторыми особенностями в алгоритме работы, в скорости, в отношении к VLAN и в ряде других

вопросов, но в целом решающих ту же задачу похожими способами. Все их принято обобщённо называть STP-протоколами.

В настоящее время протокол STP (или аналогичный) поддерживается почти всеми Ethernet-коммутаторами - как реальными, так и виртуальными, за исключением самых примитивных.

Описание протокола

Протокол работает на канальном уровне. STP позволяет делать топологию избыточной на физическом уровне, но при этом логически блокировать петли. Достигается это с помощью того, что STP отправляет сообщения BPDU и обнаруживает фактическую топологию сети. А затем, определяя роли коммутаторов и портов, часть портов блокирует так, чтобы в итоге получить топологию без петель.

Для того чтобы определить, какие порты заблокировать, а какие будут передавать данные, STP выполняет следующее:

- выбор корневого моста (Root Bridge);
- определение корневых портов (Root Port);
- определение выделенных портов (Designated Port).

Выбор корневого моста

Корневым становится коммутатор с наименьшим идентификатором моста (Bridge ID).

Только один коммутатор может быть корневым. Для того чтобы выбрать корневой коммутатор, все коммутаторы отправляют сообщения BPDU, указывая себя в качестве корневого коммутатора. Если коммутатор получает BPDU от коммутатора с меньшим Bridge ID, то он перестает анонсировать информацию о том, что он корневой, и начинает передавать BPDU коммутатора с меньшим Bridge ID.

В итоге только один коммутатор останется корневым и будет передавать BPDU.

Изначально Bridge ID состоял из двух полей.

Приоритет — поле, которое позволяет административно влиять на выборы корневого коммутатора. Размер — 2 байта.

MAC-адрес — используется как уникальный идентификатор, который, в случае совпадения значений приоритетов, позволяет выбрать корневой коммутатор.

Так как MAC-адреса уникальны, то и Bridge ID уникален, так что какой-то коммутатор обязательно станет корневым.

Определение корневых портов

Порт коммутатора, который имеет кратчайший путь к корневому коммутатору, называется корневым портом. У любого некорневого коммутатора может быть только один корневой порт. Корневой порт выбирается на основе меньшего Root Path Cost - это общее значение стоимости всех линков до корневого коммутатора. Если стоимость

линков до корневого коммутатора совпадает, то выбор корневого порта происходит на основе меньшего Bridge ID коммутатора. Если и Bridge ID коммутаторов до корневого коммутатора совпадает, то тогда корневой порт выбирается на основе Port ID.

Определение назначенных портов

Коммутатор в сегменте сети, имеющий наименьшее расстояние до корневого коммутатора, называется назначенным коммутатором (мостом). Порт этого коммутатора, который подключен к рассматриваемому сегменту сети, называется *назначенным портом*. Так же, как и корневой порт, выбирается на основе:

- меньшего Root Path Cost;
- меньшего Bridge ID;
- меньшего Port ID.

Команда show spanning-tree

Команда show spanning-tree возвращает отчет и информацию о состоянии связующего дерева для данного коммутатора. Выделенные строки кода демонстрируют, что выбранным корневым портом является порт 8. Если бы коммутатор был корневым мостом, то в этой строке находилась бы запись We are the root of the spanning tree, так как на корневом мосту корневого порта нет. Почему? Потому что корневой порт — это всегда порт, соединяющий коммутатор с корневым мостом, а на самом корневом мосту его быть не может.

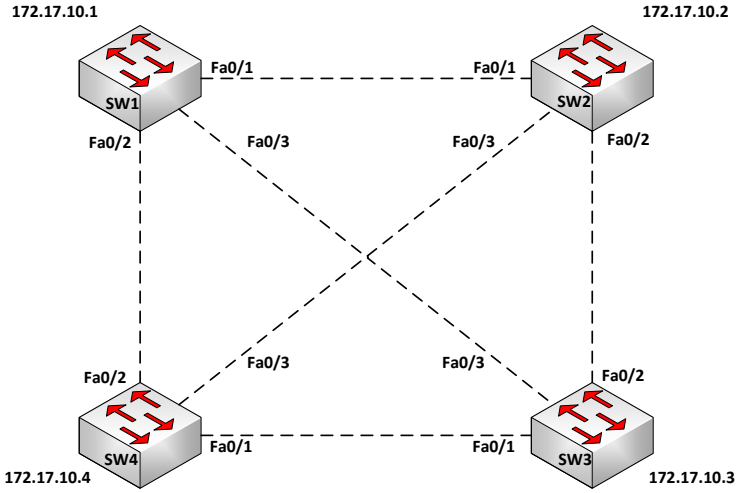
Перечисление интерфейсов говорит нам, что интерфейс Interface Fa0/1 находится в состоянии forwarding (передача); также вы можете увидеть состояния disabled (отключение) и blocking (блокировка). Наконец, еще одним важным элементом является счетчик отправленных и полученных BPDU-сообщений на данном интерфейсе.

```
S2#show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID Priority 32769
          Address cc00.0ca0.0000
          Cost 19
          Port 3(FastEthernet0/3)
          Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
  Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)
          Address cc02.0ca0.0000
          Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
          Aging Time 20
Interface          Role Sts  Cost      Prio.Nbr  Type
---------------- ---- --- --------- -------- --------------------------
Fa0/1              Altn BLK  19        128.1    P2p
Fa0/2              Desg FWD  19        128.2    P2p
Fa0/3              Root FWD  19        128.3    P2p
```

4.3. Практическая часть

Для начала выполнения лабораторной работы необходимо соединить физическую сеть в соответствии со схемой сети или построить соответствующий проект в Cisco Packet Tracer. Сразу после схемы сети в таблице указана схема адресации, которую нужно применять только тогда, когда это будет явно указано в тексте лабораторной работы.

Используемая топология



План адресации

Device	Interface	IP Address	Subnet Mask	Default Gateway
S1	VLAN 1	172.17.10.1	255.255.255.0	N/A
S2	VLAN 1	172.17.10.2	255.255.255.0	N/A
S3	VLAN 1	172.17.10.3	255.255.255.0	N/A
S4	VLAN 1	172.17.10.4	255.255.255.0	N/A

Порядок выполнения работы

1. **Базовая конфигурация оборудования.**
   - 1.1. Настроить hostname.
   - 1.2. Отключить DNS lookup.
   - 1.3. Установить пароль для EXEC mode.
   - 1.4. Настроить message-of-the-day banner.

1.5. Установить пароль для console.
1.6. Активировать пользовательские порты на каждом из коммутаторов.
1.7. На каждом из коммутаторов настроить trunk-порты.
1.8. Настроить IP-адреса для интерфейсов управления VLAN1 на S1, S2, S3 и S4.

2. Настроить Spanning Tree.

2.1. На каждом из коммутаторов (S1, S2, S3 и S4) проверить текущую конфигурацию командой **show spanning-tree.**

3. Наблюдение поведения оборудования при изменении топологии.

3.1. На каждом из коммутаторов (S1, S2 и S3) включить режим отладки протокола STP командой **debug spanning-tree events**.
3.2. На корневом коммутаторе (в данном примере S4) умышленно отключить один из trunk-портов.

Содержание отчета

Отчет о лабораторной работе должен содержать перечень команд, а также результатов их выполнения для каждого из пунктов порядка выполнения.

ЛАБОРАТОРНАЯ РАБОТА № 5
Базовая настройка статической и динамической маршрутизации протокола RIP

5.1. Цель работы

Целью данной лабораторной работы является изучение процессов настройки статических маршрутов динамической маршрутизации протокола RIP на маршрутизаторах Cisco.

5.2. Теоретическая часть

Работа современных компьютерных сетей требует организации надежных и производительных механизмов построения и прокладки маршрутов из одной сети в другую. Протокол IP является маршрутизируемым, то есть для связи сетей по данному протоколу необходимы маршруты. В сетях большого масштаба требуется максимально автономная прокладка маршрутов, для чего применяются различные протоколы маршрутизации.

Маршрутизация (routing) — процесс определения маршрута следования информации в сетях связи. Маршруты сетевого устройства содержатся в специальной таблице маршрутизации. Но способы заполнения таблицы маршруты разделяют на статические и динамические. Маршруты, задаваемые вручную администратором сети

называются статическими. Динамические маршруты — маршруты, которые были получены как результат вычисления алгоритмов маршрутизации на основе данных о топологии сети и ее состоянии. Информация о сети, предоставляется различными протоколами маршрутизации. Как правило, в компьютерных сетях функция маршрутизации выполняется специальными программно-аппаратными средствами — маршрутизаторами.

Маршрутизатор (router, роутер) — сетевое устройство третьего уровня модели OSI, обладающее как минимум двумя сетевыми интерфейсами, которые находятся в разных сетях. Причем в сетях могут использовать различные технологии физического и канального уровней. Таблица маршрутизации хранится в памяти устройства. Маршрутизаторы на рынке различаются по области их применения. Высокопроизводительные промышленные маршрутизаторы выпускаются в 19-дюймовых Unit корпусах для монтажа в серверную стойку. Их функциональность, как правило, можно расширить с помощью установки дополнительных модулей. Такие маршрутизаторы отличает высокая стоимость. Маршрутизаторы для дома и небольшого офиса имеют малый размер, просты в настройке. Зачастую в них встроены коммутатор и беспроводная точка доступа. В рамках данной лабораторной работы будет рассмотрена настройка как статической, так и динамической маршрутизации по протоколам RIP1/RIP2.

Специальные термины и понятия

Метрика — числовой коэффициент, влияющий на выбор маршрута в компьютерных сетях. Как правило, определяется количеством хопов (ретрансляционных переходов) до сети назначения или параметрами канала связи. Чем метрика меньше, тем маршрут приоритетнее.

Административное расстояние (administrative distance) — коэффициент надежности маршрута, используемый на маршрутизаторах компании Cisco. Приоритет имеет тот маршрут, который обладает меньшим административным расстоянием (см. табл. 5.1). В случае двух одинаковых маршрутов с одинаковым административным расстоянием рассматривается метрика маршрута. Таким образом, использование административных расстояний позволяет резервировать маршруты.

Примечание. Значения административного расстояния для маршрутов не рассылаются протоколами маршрутизации и используются только локально.

Таблица 5.1

Административные расстояния для некоторых видов маршрутизации.

Вид	**Административное расстояние**
Напрямую подключенная сеть (directly connected)	0
Статический маршрут на интерфейс/следующую сеть	1
EIGRP	90
OSPF	110
RIP	120

Шлюз по умолчанию (default gateway), шлюз последней надежды (last resort gateway) — адрес маршрутизатора, на который отправляется трафик для которого не нашлось отдельных записей в таблице маршрутизации. Для устройств, подключенных к одному маршрутизатору (как правило, это рабочие 5 станций), использование шлюза по умолчанию — единственная форма маршрутизации. Шлюз последней надежды применяется обычно в устройствах (маршрутизаторах), где ситуация, в которой не найдется отдельного маршрута, является исключительной.

Шаблонная маска (wildcard mask) — маска, указывающая на количество хостов сети. Является дополнением для маски подсети. Вычисляется по формуле для каждого из октетов маски подсети как 255-маска подсети. Например, для сети 192.168.1.0 и маски подсети 255.255.255.242 шаблонная маска будет выглядеть как 0.0.0.13. Шаблонная маска используется в настройке некоторых протоколов маршрутизации, а также является удобным параметром ограничений в списках доступа.

Автономная система — группа маршрутизаторов, обменивающаяся маршрутной информацией с помощью одного протокола.

Петля маршрутизации — явление, возникающее, когда маршрутизатор отсылает пакет на неверный адрес назначения. Получивший такой пакет маршрутизатор возвращает его обратно. Таким образом получается петля. Для борьбы с подобными петлями в TCP/IP предусмотрен механизм TTL. Протоколы маршрутизации также предлагают способы борьбы с петлями.

Конфигурирование протоколов маршрутизации

Командная оболочка имеет встроенные средства поддержки конечного набора протоколов IP-маршрутизации. Поддерживается ряд протоколов внешней (BGP) и внутренней маршрутизации (RIP, OSPF, EIGRP и т.д.). Наряду с этим, естественной является также поддержка статической маршрутизации на основе таблиц, наполнение которых производится статическими правилами пересылки пакетов между

сетями. Для конфигурирования работы протоколов динамической маршрутизации необходимо использовать подрежим конфигурирования маршрутизатора (router). Следует напомнить, что данный подрежим поддерживает работу и настройку протоколов маршрутизации, и, чтобы войти в него, необходимо использовать команду router, а также имя протокола в качестве аргумента, например, *router rip*. Команды, используемые в данном подрежиме, имеют строгую зависимость от того, какой именно протокол маршрутизации подвергается конфигурированию.

Статические записи вносятся в таблицу маршрутизации с помощью перечисления их в командной оболочке в режиме глобального конфигурирования (configure terminal). Учитывая любое изменение в логической или физической топологии сети, становится необходимым производить перманентное переконфигурирование таблиц маршрутизации.

Статическая маршрутизация

Преимущества использования статической маршрутизации в ограниченных по размеру и глубине сетях очевидны. Скорость развертывания, производительность протоколов внутренней маршрутизации, требования к аппаратному обеспечению и административной поддержке не сравнимы с трудозатратами, необходимыми на организацию и поддержку работоспособности протоколов внешней маршрутизации. Статические записи маршрутизации протокола IP вносятся в конфигурацию маршрутизатора в режиме глобального конфигурирования. Для этого необходимо использовать команду ip route, в качестве аргументов которой необходимо передать адрес сети, маску сети и шлюз, через который будет осуществляться маршрутизация к указанной сети.

Например, статический маршрут к сети 10.10.11.0/28 на маршрутизаторе R2 на рис. 5.1, может выглядеть следующим образом:

R2 # conf t

R2 (config)# ip route 10.10.11.0 255.255.255.240 172.17.117.1

R2 (config)# ^ Ctrl - Z

То есть указывается шлюз 172.17.117.1, через который будет осуществляться пересылка пакетов, предназначенных для сети 10.10.11.0/28. Используя статическую маршрутизацию, становится возможным указывать маршрутизатору маршрут по умолчанию, который используется в том случае, когда системе неизвестен конкретный шлюз до системы назначения.

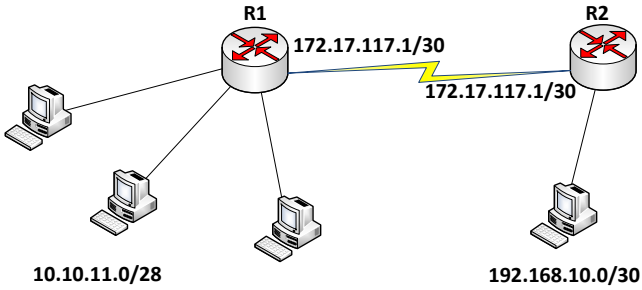


Рис. 5.1. Сеть со статической маршрутизацией отправителя

Статическая запись маршрута по умолчанию приведена ниже:

name#conf t

name(config)# ip route 0.0.0.0 0.0.0.0 10.1.1.1

name(config)# ^ Ctrl - Z.

Такая конфигурация подразумевает, что все пакеты, к узлу получателя которых нет прямого (известного) маршрута, будут передаваться шлюзу 10.1.1.1. Данному узлу делегированы полномочия по дальнейшей пересылке пакетов к сети узла получателя.

Если конфигурируемый маршрутизатор сам выполняет роль шлюза, как это показано на рис. 5.2 (R1), то в таблице маршрутизации вместо IP-адреса шлюза по умолчанию можно указать интерфейс, который, например, непосредственно подключен к сети доступа поставщика услуг Internet.

//Определение интерфейса для маршрута по умолчанию

name#conf t

name(config)# ip route 0.0.0.0 0.0.0.0 serial 0/0

name(config)# ^ Ctrl – Z.

Существует еще одна из плоскостей применения статической маршрутизации. Командная оболочка позволяет использовать специальный программный интерфейс null0 в качестве приемника для любого типа трафика. Любой трафик, перенаправляемый на данный интерфейс, будет безвозвратно уничтожаться.

Динамическая маршрутизация

Рассматриваемые в данном разделе протоколы внутренней маршрутизации предназначены для автоматизированного сбора сведений о логической и физической топологии сети и наполнении корректными записями таблицы маршрутизации. Любое изменение сети через определенный интервал времени, необходимый на обработку и рассылку новой информации о маршрутах, будет зафиксировано протоколом динамической маршрутизации и отображено в таблице маршрутизации. Кроме этого, любые системы,

участвующие в обмене маршрутной информацией, также будут уведомлены о произошедших изменениях.

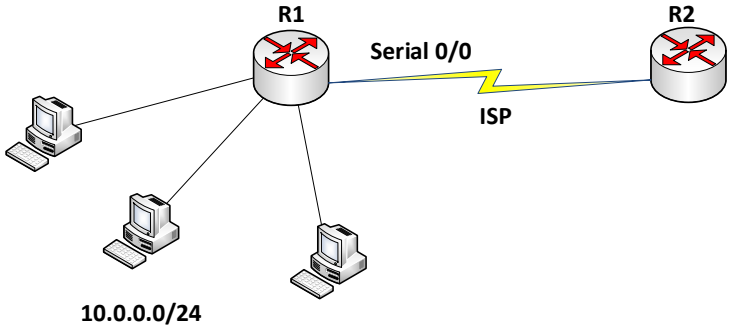


Рис. 5.2. Статическая маршрутизация через заданный интерфейс

Как уже было несколько раз отмечено ранее, конфигурирование работы протоколов динамической маршрутизации происходит в подрежиме конфигурирования маршрутизатора (router). Переход в данный подрежим осуществляется с помощью команды ***router***, которой передается имя протокола в качестве аргумента.

Например:

name > enable
name # configure terminal
name (config)# router rip
name (config - router)#

Протокол RIP

Несмотря на то, что протокол RIP, по меркам сетевого мира, является достаточно пожилым программным обеспечением, он находит широкое применение в сегментах сетей с ограниченным адресным пространством. Как правило, такие сети представляют собой совсем небольшие по территориальным размерам объединения предприятий и офисов, в которых основным козырем протокола динамической маршрутизации становится вменяемость (сложность) настройки и поддержки, а также требования к аппаратным ресурсам.

Улучшенная версия протокола (RIPv2) поддерживает сетевые маски переменной длины и бесклассовую маршрутизацию. Дополнительно к этому вторая версия протокола поддерживает вычисление метрики расстояния на основе критерия пропускной способности канала передачи данных. Однако так же, как и в предыдущей версии, протокол RIPv2 не поддерживает длину маршрута, превышающую 16 транзитных переходов. Следует также отметить, что исполняемый процесс протокола RIP в качестве

информационной обменной единицы, курсирующей между узлами, обслуживающими сеть с помощью данного протокола, использует полную таблицу маршрутизации. Другими словами, протокол RIP осуществляет широковещательную рассылку полной таблицы маршрутизации. Этим свойством, в некоторых случаях значительно сказывающемся на доступной пропускной способности и загруженности сети, он значительно отличается от более современных протоколов, в которых применяется частичное обновление маршрутных записей.

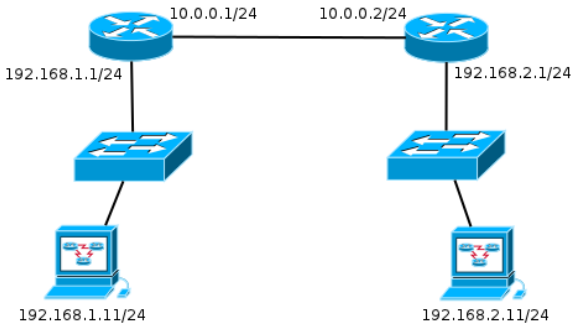
Командная оболочка имеет набор встроенных команд для поддержки протоколов RIP и RIPv2, которые используются в подрежиме конфигурации маршрутизатора (router).

Команда *router rip* позволяет перейти в подрежим конфигурирования протокола RIP, при этом приглашение командной оболочки изменится, например, на *R1(config-router)#.* Находясь в подрежиме конфигурирования протоколов динамической маршрутизации (в данном случае RIP), можно использовать специальные команды для непосредственной конфигурации программных свойств протокола.

По умолчанию командная оболочка использует первую версию протокола **RIP**. Для того, чтобы включить поддержку **RIPv2**, необходимо явно указать версию протокола с помощью команды *version 2*.

Настройка RIPv2 в программе Cisco Packet Tracer

Пример настройки RIP2 (без указания масок подсети) для сети, изображенной ниже:



Подразумевается, что все необходимые сетевые интерфейсы на каждом устройстве предварительно настроены.

Router>enable (вход в привилегированный режим)

Router#configure terminal (вход в консоль настройки)

Router(config)#router rip (вход в режим конфигурации протокола RIP)

Router(config-router)#network 192.168.1.0 (добавление непосредственно подключенной к роутеру сети)

Router(config-router)#network 10.0.0.0 (добавление другой непосредственно подключенной к роутеру сети)

Router(config-router)#version 2 (использовать RIP2)

Router(config-router)#exit (выход из режима конфигурации протокола RIP)

Router(config)#exit (выход из консоли настройки)

Router#write memory (сохранение настроек в память устройства)

Building configuration...

[OK]

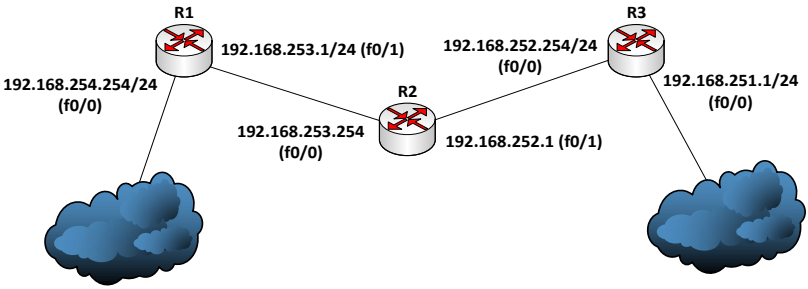
Аналогичные действия проделываются и для второго маршрутизатора, но вместо сети 192.168.1.0 прописывается непосредственно подключенная к нему сеть 192.168.2.0. Обратите внимание на то, что не указывается маска подсети. Она берется автоматически из настроек интерфейса.

5.3. Практическая часть

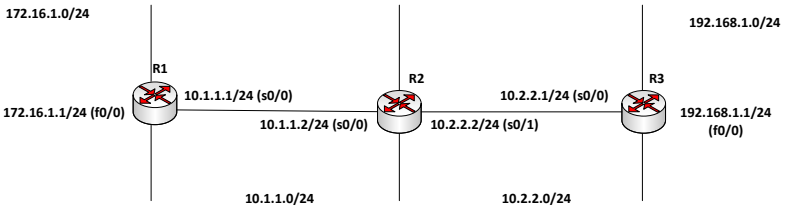
Данная лабораторная работа может быть выполнена на реальном оборудовании или в Cisco Packet Tracer. Все необходимые действия указаны по порядку их выполнения. Для начала выполнения лабораторной работы необходимо соединить физическую сеть в соответствии со схемой сети или построить соответствующий проект в Cisco Packet Tracer. Сразу после схемы сети в таблице указана схема адресации, которую нужно применять только тогда, когда это будет явно указано в тексте лабораторной работы.

Используемая топология

Для задания 1



Для задания 2



Порядок выполнения работы

Задание 1

1.1. Настроить сетевые интерфейсы и hostname на роутерах R1, R2, R3.

1.2. Настроить статическую маршрутизацию для сетей 192.168.254.0/24 и 192.168.251.0/24 на R2.

1.3. Настроить шлюз по умолчанию на R1 и R2.

1.4. Проверить сделанные настройки show ip route и утилитой ping.

Задание 2

2.1. Провести начальную конфигурацию маршрутизаторов R1, R2 и R3.

2.2. Настроить интерфейсы маршрутизаторов R1, R2 и R3.

2.3. Активировать RIP версии 2 и указать подключенные сети на маршрутизаторах R1, R2 и R3.

2.4. Проверить сделанные настройки.

- R1#ping 192.168.1.1
- R3#ping 172.16.1.1

Успешное прохождение ICMP-пакетов из одного конца сети в другой говорит об успешно выполненном конфигурировании маршрутизаторов.

Содержание отчета

Отчет о лабораторной работе должен содержать перечень команд, а также результатов их выполнения для каждого из пунктов порядка выполнения.

ЛАБОРАТОРНАЯ РАБОТА № 6

Базовая настройка протокола EIGRP

6.1. Цель работы

Целью данной лабораторной работы является изучение процессов настройки протокола EIGRP на маршрутизаторах Cisco.

6.2. Теоретическая часть

EIGRP

Протокол Enhanced IGRP, как и IGRP, является собственным протоколом компании Cisco, защищенным патентом; маршрутизаторы других производителей не поддерживают EIGRP, но это, пожалуй, единственный недостаток данного протокола. EIGRP обеспечивает отличную производительность, легко настраивается, поддерживает VLSM, а также IPX и AppleTalk. Это — протокол на основе вектора расстояния, обладающий некоторыми характеристиками протокола на основе состояния линии связи. EIGRP использует ту же составную метрику, что и его предшественник - протокол IGRP, но, в отличие от протокола IGRP, для которого характерна проблема маршрутных петель, EIGRP очень эффективно предотвращает их появление. Самой удивительной чертой EIGRP можно назвать его сдвоенный конечный автомат (dual finite state machine), обеспечивающий исключительно малое время схождения; кроме того, этот протокол поддерживает частичные обновления таблиц маршрутизации (на обновление маршрутной информации тратится меньше пропускной способности и ресурсов процессора), обеспечивает автоматическое обнаружение соседей и характеризуется улучшенной масштабируемостью.

Типы маршрутов EIGRP

EIGRP поддерживает несколько типов маршрутов, у каждого типа маршрута своё значение administrative distance:

- **internal** -- внутренние маршруты EIGRP. AD – 90;
- **external** -- маршруты, перераспределенные в процесс EIGRP из других источников. AD – 170;
- **summary** -- суммарные маршруты EIGRP. AD -- 5 (по умолчанию может быть любое - задается администратором при написании команды суммирования).

Базовые настройки протокола EIGRP

Включение протокола

Создание процесса EIGRP:

name(config)# router eigrp <AS>

Номер автономной системы должен быть одинаковым на всех маршрутизаторах, которые должны обмениваться информацией по протоколу EIGRP.

Для того чтобы маршрутизаторы начали обмениваться информацией, необходимо включить EIGRP на интерфейсах.

Включение EIGRP на интерфейсах:

name(config)# router eigrp 100

name(config-router)# network <network> [wildcard mask]

Параметры команды network:

- <network> — непосредственно присоединенная сеть к маршрутизатору;
- [wildcard mask] — маска, которая указывает с помощью нулей, какая часть из указанной сети должна совпадать, а с помощью 1 - какая часть сети может быть произвольной. Этот параметр опциональный. Если его не задать, то будет использоваться классовая маска для указанной сети.

Команда **network** делает следующее:

- включает EIGRP на интерфейсе, IP-адрес которого совпадает с указанной сетью и маской;
- анонсирует сеть этого интерфейса через другие интерфейсы, на которых включен EIGRP.

Сеть интерфейса анонсируется, только если интерфейс в состоянии up/up.

Команда network включает EIGRP на текущих интерфейсах и на всех следующих, которые появятся и совпадут с сетью, которая указана в команде network.

Включение режима объединения маршрутов на конкретном интерфейсе

Объединение маршрутов можно контролировать на уровне интерфейса. На интерфейсе ethernet0 можно было бы применить следующую команду для объединения любых выбранных маршрутов:

interface ethernet0

ip surrmary-address eigrp 100 10.101.1.0 255.255.255.0

При объявлении через интерфейс ethernet0 объединенному маршруту 10.101.1.0 присваивается административное расстояние 5, благодаря чему он замещает любые другие EIGRP-маршруты.

Метрики протокола EIGRP

EIGRP-метрики идентичны IGRP-метрикам, включая значения *K* в уравнении метрики. Единственное различие состоит в том, что EIGRP-метрика умножается на 256, превращаясь в 32-битное целое число вместо 24-битного.

Суммирование маршрутов

Суммарные маршруты EIGRP

- По умолчанию включено автоматическое суммирование маршрутов.
- Минимальная метрика (лучшая метрика) из всех маршрутов, которые суммируются, используется в качестве метрики суммарного маршрута.

- Когда последний специфический маршрут, который был объединен в суммарный, пропадет из таблицы маршрутизации (на маршрутизаторе, на котором выполняется суммирование), пропадет и суммарный маршрут.
- При создании суммарного маршрута маршрутизатор автоматически добавляет в таблицу маршрутизации этот суммарный маршрут с next-hop, указывающим на null0.

Отключение автоматического суммирования маршрутов:

name(config-router)#no auto-summary

После отключения автоматического суммирования на локальном маршрутизаторе появляются такие сообщения:

name(config-router)#no auto-summary

name(config-router)#

**Sep 10 04:10:41.989: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 192.168.2.1 (FastEthernet0/0) is resync: summary configured*

**Sep 10 04:10:41.993: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 192.168.4.5 (FastEthernet2/0) is resync: summary configured*

name(config-router)#

На соседе этого маршрутизатора:

name(config-router)#

**Sep 10 04:27:19.917: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 192.168.4.3 (FastEthernet0/0) is resync: peer graceful-restart*

name(config-router)#

Суммарный маршрут настраивается на интерфейсе:

dyn3(config-if)#ip summary-address eigrp <AS-number> <address> <mask> [admin-distance]

По умолчанию у суммарного маршрута EIGRP administrative distance — 5. AD суммарного маршрута используется для того, чтобы определить, помещать ли маршрут в null0 для суммарного маршрута в таблицу маршрутизации. AD является локальным параметром и не передается соседям. У соседей суммарный маршрут будет с AD = =internal = 90.

Если необходимо, чтобы маршрут в null0 не помещался в таблицу маршрутизации, то необходимо указать AD = 255.

Пример задания суммарного маршрута:

dyn3(config-if)#ip summary-address eigrp 1 172.16.0.0 255.255.0.0

Команды show для протокола EIGRP

Существует несколько команд show для протокола EIGRP, которые недоступны для прочих протоколов. Рассмотрим наиболее полезные.

Команда show ip eigrp neighbors

Команда show ip eigrp neighbors выводит список соседних EIGRP-маршрутизаторов, о которых известно данному маршрутизатору. В нашей сети маршрутизатор 1 «видит» маршрутизаторы 2 и 3. В столбце Interface указано, какой интерфейс получил приветствие от соседа. В столбце Hold представлено время удержания (в секундах), в течение которого маршрутизатор ожидает сообщения от соседа перед тем, как объявить его отключенным. Время в столбце Uptime — это время, прошедшее с момента, когда маршрутизатор впервые узнал о соседе. Значение в столбце SRTT (Smooth Round Trip Time — среднее время оборота) — это число миллисекунд, за которое маршрутизатор может отправить EIGRP-пакет соседу и получить ответ. В столбце RT0 показано значение тайм-аута для повторной передачи, то есть время ожидания (в миллисекундах) до повторной отправки пакета соседу. Значение в столбце Q Cnt — это число ожидающих отправки EIGRP-пакетов, находящихся в очереди маршрутизатора. Наконец, в столбце Seq Num представлен порядковый номер последнего пакета, полученного от соседа:

*Router1#**show ip eigrp neighbors***
IP-EIGRP neighbors for process 100
H Address Interface Hold Uptime SRTT RTO Q Seq
(sec) (ms) Cnt Num
1 192.168.1.14 Seo 11 15:40:05 32 1164 0 7
0 192.168.1.6 Sel 10 15:40:22 434 2604 0 9

Счетчики соседей EIGRP-маршрутизатора можно сбросить следующей командой:

clear ip eigrp neighbors.

Режим ведения журнала включается такой командой:

eigrp log-neighbor-changes.

Все изменения соседей записываются в файл журнала. Это означает, что если режим ведения журнала включен, то на маршрутизаторе должна быть допустимая конфигурация системного журнала.

Команда show ip eigrp topology

Команда show ip eigrp topology позволяет увидеть представление маршрутизатора о сетевой EIGRP-топологии. С каждой записью в топологии связывается состояние: Passive (пассивное), Active (активное), Update (обновление), Query (очередь), Reply (ответ) или Reply Status (состояние ответа). Passive означает, что никакие вычисления протоколом EIGRP не используются. Active означает, что EIGRP выполняет вычисления для данной точки назначения. Update, Query и Reply просто сообщают, что пакет указанного типа был отправлен точке назначения. Запись Reply Status говорит о том, что

был отправлен ответный пакет и маршрутизатор ожидает ответа.

Последнее, что необходимо знать для расшифровки данной таблицы, — это достижимое расстояние (Feasible Distance, FD). Для каждой записи за FD-значением следует косая черта (/) и еще одно число, представляющее подтвержденное расстояние до соседа. Если подтвержденное расстояние меньше достижимого, то данный путь становится следующим участком маршрута.

*Router1**#show ip eigrp topology***
IP-EIGRP Topology Table for process 100
Codes: P - Passive. A - Active. U - Update. Q - Query. R - Reply,
r - Reply status
P 192.168.1.8/30. 2 successors. FD is 21504000
via 192.168.1.14 (21504000/20992000). Serial0
via 192.168.1.6 (21504000/20992000). Serial1
P 192.168.1.12/30. 1 successors. FD is 20992000
via Connected. Serial0
P 192.168.1.4/30. 1 successors. FD is 20992000
via Connected. Serial1
P 172.16.1.0/24. 1 successors. FD is 281600
via Connected. Ethernet0
P 172.16.2.0/24. 1 successors. FD is 21017600
via 192.168.1.6 (21017600/281600). Serial1
P 172.16.3.0/24. 1 successors. FD is 21017600
via 192.168.1.14 (21017600/281600). Serial0

Команда show ip eigrp traffic

Команда show ip eigrp `traffic` просто выводит сведения о пакетах вызова, обновлениях, запросах, ответах и подтверждениях, отправленных процессом маршрутизации протокола EIGRP. Для каждого типа пакета первое число — это количество отправленных пакетов, второе число — количество полученных пакетов.

*Router1**#show ip eigrp traffic***
IP-EIGRP Traffic Statistics for process 100
Hellos sent/received: 24728/24704
Updates sent/received: 23/19
Queries sent/received: 1/1
Replies sent/received: 1/1
Acks sent/received: 12/15

Перераспределение маршрутов с других протоколов на протокол EIGRP

В многопротокольной сети необходимо определить метрики по умолчанию для обработки маршрутов, перераспределяемых с других протоколов на протокол EIGRP.

RIP

Для того чтобы включить режим перераспределения с RIP на EIGRP, необходимо просто определить метрику по умолчанию для входящих RIP-маршрутов. Пример:

Определение RIP-процесса
router rip
network 192.168.1.0
Определение EIGRP-процесса и добавление RIP-маршрутов
router eigrp 100
network 10.0.0.0
default-metric 1000 250 255 1 1500
redistribute rip

Этот пример практически идентичен другому, в котором ранее мы демонстрировали перераспределение маршрутов с RIP на IGRP. Это неудивительно, так как способ вычисления метрики в протоколах IGRP и EIGRP один и тот же. В данном примере осуществляется перераспределение только в одну сторону: с RIP на EIGRP. Необходимо соблюдать осторожность при перераспределении маршрутов в оба направления, так как это требует фильтрации маршрутов для предотвращения появления маршрутных петель.

IGRP

При одновременном использовании в маршрутизаторе протоколов IGRP и EIGRP перераспределение осуществляется автоматически, если процессы маршрутизации протоколов IGRP и EIGRP обладают одинаковыми номерами процессов. Если же номера процессов различаются, то необходимо использовать команду redistribute. В следующем примере перераспределение происходит автоматически, потому что номера процессов обоих протоколов маршрутизации совпадают (и равны 100).

Определение процесса маршрутизации для протокола IGRP:
router igrp 100
network 10.0.0.0.

Определение IGRP с таким же номером, что и у IGRP, чтобы между ними автоматически осуществлялось перераспределение:

router eigrp 109
network 10 .0.0.0.

В следующем примере номера процессов маршрутизации различаются, поэтому перераспределения маршрутов автоматически не происходит — приходится добавить команду redistribute. Так как оба протокола — IGRP и EIGRP — применяют одну и ту же метрику, команда default-metric не нужна.

Определение процесса маршрутизации для протокола IGRP:
router igrp 100

network 10.0.0.0.

Определение процесса маршрутизации для протокола IGRP-маршрутов:

router eigrp 109

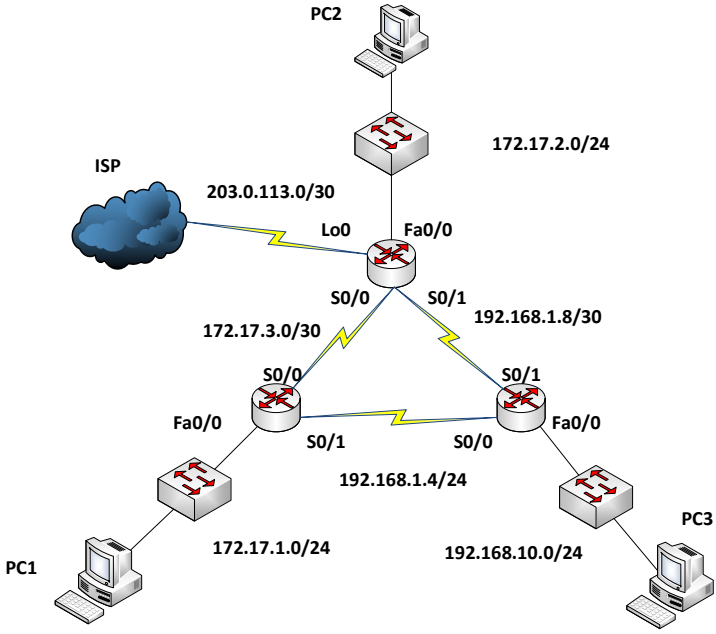
network 10.0.0.0

redistribute igrp 100.

6.3. Практическая часть

Данная лабораторная работа может быть выполнена на реальном оборудовании или в Cisco Packet Tracer. Все необходимые действия указаны по порядку их выполнения. Для начала выполнения лабораторной работы необходимо соединить физическую сеть в соответствии со схемой сети или построить соответствующий проект в Cisco Packet Tracer. Сразу после схемы сети в таблице указана схема адресации, которую нужно применять только тогда, когда это будет явно указано в тексте лабораторной работы.

Используемая топология



План адресации

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	Fa0/0	172.17.1.1	255.255.255.0	N/A
	S0/0	172.17.3.1	255.255.255.252	N/A
	S0/1	192.168.1.5	255.255.255.252	N/A
R2	Fa0/0	172.17.2.1	255.255.255.0	N/A
	S0/0	172.17.3.2	255.255.255.252	N/A
	S0/1	192.168.1.9	255.255.255.252	N/A
	Lo1	203.0.113.1	255.255.255.252	N/A
R3	Fa0/0	192.168.10.1	255.255.255.0	N/A
	S0/0	192.168.1.6	255.255.255.252	N/A
	S0/1	192.168.1.10	255.255.255.252	N/A
PC1	NIC	172.17.1.10	255.255.255.0	172.17.1.1
PC2	NIC	172.17.2.10	255.255.255.0	172.17.2.1
PC3	NIC	192.168.10.10	255.255.255.0	192.168.10.1

Порядок выполнения работы

1. **Базовая конфигурация оборудования.**
   1.1. Настроить hostname на маршрутизаторах.
   1.2. Отключить DNS lookup.
   1.3. Установить пароль для EXEC mode.
   1.4. Настроить message-of-the-day banner.
   1.5. Установить пароль для console.
2. **Настроить адресацию оборудования согласно плану.**
   2.1. Настроить интерфейсы на **R1**, **R2** и **R3** согласно плану адресации.
   2.2. Проверить выполненные настройки командой **show ip interface brief.**
3. **Настроить протокол динамической маршрутизации EIGRP на маршрутизаторе R1.**
   3.1. Активировать **EIGRP** на **R1,** используя **Process ID 1**.
   3.2. Используя команду **network**, сконфигурировать маршрутизатор **R1** для анонсирования сети 172.17.0.0.
   3.3. Настроить анонсирование сети 192.168.1.4/30.
4. **Настроить протокол динамической маршрутизации EIGRP на маршрутизаторе R2.**
5. **Настроить протокол динамической маршрутизации EIGRP на маршрутизаторе R3.**
6. **Проверить правильность выполненных настроек.**

6.1. Используя команды **show ip eigrp neighbors** вывести таблицу соседних **EIGRP**-маршрутизаторов.
6.2. Вывести детальную информацию по используемому протоколу динамической маршрутизации командой **show ip protocols.**

7. Анализ EIGRP маршрутов.
7.1. Просмотр таблицы маршрутизации.

8. Настройка EIGRP-Metrics.
8.1. Получение значений, составляющих **EIGRP-Metric** для интерфейса **Serial 0/0** маршрутизатора **R1** из вывода команды **show interfaces Serial0/0**.
8.2. Установка значений **bandwidth** для **Serial**-интерфейсов маршрутизаторов **R1, R2** и **R3**.
8.3. Проверить выполненные изменения параметра **bandwidth**.

9. Отключение автоматического суммирования маршрутов EIGRP.
9.1. Получение информации о суммарных маршрутах в таблице маршрутизации **R3**.
9.2. Отключить автоматическое суммирование маршрутов.
9.3. Проверить результат изменения конфигурации в таблице маршрутизации **R3**.

Содержание отчета

Отчет о лабораторной работе должен содержать перечень команд, а также результатов их выполнения для каждого из пунктов порядка выполнения.

ЛАБОРАТОРНАЯ РАБОТА № 7
Базовая настройка протокола OSPF

7.1. Цель работы

Целью данной лабораторной работы является изучение процессов настройки протокола EIGRP на маршрутизаторах Cisco.

7.2. Теоретическая часть

Протокол OSPF (Open Shortest Path First — выбор кратчайшего пути первым) относится к типу протоколов, работающих на основе состояния линии связи. Это хороший вариант для сетей, в которых требуется более сложный протокол маршрутизации, чем RIP, и применяются маршрутизаторы разных производителей, из-за чего использование EIGRP становится невозможным. У протокола OSPF есть несколько достоинств (большая их часть характерна также для протокола EIGRP): экономное использование пропускной способности, поддержка VLSM, быстрая сходимость при

изменении состояния сети. Самым большим преимуществом OSPF по сравнению с EIGRP является то, что это — открытый стандарт, поддерживаемый практически всеми производителями маршрутизаторов.

Протокол OSPF заслужил репутацию очень сложного протокола, и до некоторой степени такое мнение о нем оправданно. Однако в небольших сетях настройка протокола OSPF не занимает много времени. В данной книге мы не планируем всесторонне рассмотреть этот протокол и его возможности.

Как вы увидите, применение OSPF заставляет выбрать определенную топологию сети. OSPF делит сеть на области: область 0 — это магистраль, к которой подключаются все остальные области. Хотя такой дизайн подходит ко многим, возможно, даже к большинству сетей, вы можете обоснованно воспротивиться использованию протокола маршрутизации, потенциально ограничивающего гибкость вашей сети. Однако для некоторых людей этот аспект применения протокола OSPF является скорее достоинством, поскольку заставляет задействовать испытанные приемы построения сетей.

Единственным потенциальным недостатком протокола OSPF может быть то, что он отнимает очень много процессорного времени маршрутизатора. Однако захват процессора можно контролировать, ограничивая число маршрутизаторов в области, — еще один хороший прием из практики построения сетей.

Концепции

Перед тем как говорить что-либо о протоколе OSPF, важно познакомиться с его основными «строительными блоками». Как и раньше, не пытаясь объяснить все в деталях, просто представим некоторые важные концепции, которые помогут вам приступить к работе.

Области

Область (area) — это группа маршрутизаторов; хороший дизайн предполагает, что одна область содержит не более 50 маршрутизаторов (или 100 интерфейсов). Каждой области назначается номер, начиная с нуля (0). Магистральной областью является область 0 — она должна присутствовать в любой OSPF-сети. На рис. 7.1 показано, как области сети упорядочиваются в топологической композиции.

Область 0 на рисунке — это магистраль, к ней подключаются все остальные области. Эта концепция позволяет объединять сетевые адреса, уменьшая размер таблиц маршрутизации. Чем меньше таблицы маршрутизации, тем быстрее сходимость, тем меньше

затраты пропускной способности на передачу информации протокола маршрутизации и тем лучше распознавание маршрутов. Однако использование большого числа относительно маленьких областей также может привести к усложнению конфигурации.

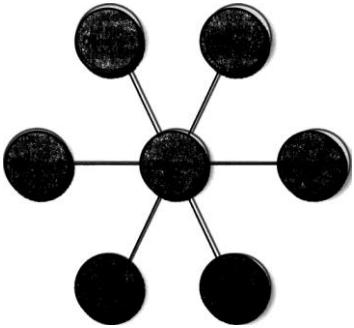


Рис. 7.1. Топология областей OSPF-сети

Типы маршрутизаторов

OSPF-маршрутизатор можно отнести к одному из следующих типов.

Граничный маршрутизатор автономной системы (Autonomous System Border Router, ASBR)

Маршрутизатор, один или несколько интерфейсов которого подключены к внешней сети или к сети с другим номером локальной автономной системы.

Граничный маршрутизатор области (Area Border Router, ABR)

Маршрутизатор, один или несколько интерфейсов которого подключены к другим областям. Информация обо всех областях, к которым подключен маршрутизатор, хранится внутри устройства.

Внутренний маршрутизатор области (area-internal router)

Маршрутизатор, все интерфейсы которого находятся в одной и той же области.

Магистральный маршрутизатор (backbone router)

Маршрутизатор, один или несколько интерфейсов которого подключены к магистральной OSPF-области. Магистральная область - это область 0.

Объявления о состоянии связей

Перед тем как приступать к обсуждению различных типов OSPF-областей, давайте выясним, каким образом между областями

рассылаются обновления, называемые объявлениями о состоянии связей (Link-State Advertisement, LSA. LSA — это групповая рассылка в OSPF-сети, рассказывающая об изменении или обновлении маршрутов другим маршрутизаторам или областям).

Существует шесть разных типов LSA-объявлений, определяющих тип объявляемого маршрута и способ обработки извещений. LSA-объявления отправляются каждые 30 минут или при каждом изменении состояния связи. Далее перечислены типы LSA-объявлений.

Тип 1 — связь маршрутизаторов *(router link)*

LSA-объявления типа 1 рассылаются всем маршрутизаторам области (то есть заполняют область). Они содержат всю информацию относительно состояния связей.

Тип 2 — сеть *(network)*

LSA-объявления типа 2 содержат сведения о сети. Уполномоченный маршрутизатор OSPF-сети (об уполномоченных маршрутизаторах см. далее) рассылает это объявление всем маршрутизаторам областей.

Тип 3 — внутренняя сводка *(internal summary)*

LSA-объявления типа 3 содержат информацию о маршрутах для внутренних сетей. Эта информация рассылается ABR-маршрутизатором всем магистральным маршрутизаторам.

Тип 4 — внешняя сводка ***(external summary)***

LSA-объявления типа 4 содержат информацию о маршрутах для ASBR-маршрутизаторов.

Тип 5 — автономная система (autonomous system)

LSA-объявления типа 5 содержат информацию о маршрутах, относящихся к внешним сетям. Такие объявления рассылаются только ASBR-маршрутизаторами.

Тип 6 — групповая рассылка протокола OSPF ***(Multicast OSPF, MOSPF)***

С помощью LSA-объявления типа 6 осуществляется групповая рассылка специфической информации. Маршрутизатор Cisco игнорирует объявления такого типа и при получении генерирует запись в системном журнале. Чтобы запретить создание сообщений в системном журнале, используйте команду ignore Isa mospf.

Тип 7 — внешние LSA-объявления для NSSA-областей ***(NSSA External LSA)***

LSA-объявления типа 7 используются в областях, называемых NSSA (Not- So-Stubby Area — не совсем тупиковая область).

Типы областей

В OSPF-сети могут присутствовать области нескольких типов.

Магистральная область (backbone area)

Любая OSPF-сеть требует наличия магистральной области, объединяющей воедино несколько областей. Магистральная область — это всегда область 0.

Стандартная область (standard area)

Стандартная область подключается к магистральной области и получает как внутренние, так и внешние LSA-объявления.

Тупиковая область (stub area)

Тупиковой области не нужны все маршруты, получаемые другими областями; ей необходимы лишь маршрут по умолчанию и LSA-сводки. Она не получает информации о внешних маршрутах. Тупиковая область обычно содержит не больше 50 маршрутизаторов, и ее адреса можно легко представлять находящимся выше областям в объединенном виде. Все области, за исключением области 0, могут быть тупиковыми, если не содержат ASBR-маршрутизаторов. Для создания тупиковой сети используется команда stub, например: *area 1 stub.*

Полностью тупиковая область (totally stubby area)

Полностью тупиковая область не получает внешних LSA-объявлений и LSA- сводок. Она практически идентична тупиковой области, за исключением того, что не получает совершенно никакой сводной информации, только маршрут по умолчанию. Для настройки полностью тупиковой области добавьте к командам конфигурирования области команду no-summary, например: area 1 stub no-summary.

Не совсем тупиковая область *(Not-So-Stubby Area, NSSA)*

Не совсем тупиковая область напоминает тупиковую область, но поддерживает обмен маршрутной информацией с внешней сетью, использующей другой протокол маршрутизации. Иными словами, это обычная тупиковая область, но имеющая ASBR-маршрутизатор. Удаленная сеть становится NSSA-областью в нашей OSPF-сети, что устраняет необходимость встраивания в сеть другого протокола маршрутизации. ASBR-маршрутизатор NSSA-области предоставляет маршруты, перераспределенные с другого протокола маршрутизации, и передает их обратно в магистральную область 0. Для определения сети в качестве NSSA-сети используйте команду nssa, например:

area 1 nssa.

Полностью, но не совсем тупиковая область *(totally stubby not-so-stubby area)*

Не пугайтесь, вы прочитали название правильно. Более того, благодаря такому названию названия остальных областей кажутся вполне нормальными. По существу, чтобы создать такую область, после команды **nssa** просто добавляется команда **no-summary.** Делая так, мы блокируем в NSSA-области LSA-объявления типов 3 и 4. То

есть целиком команда может выглядеть приблизительно так:

area 1 nssa no summary.

Идентификатор маршрутизатора

У каждого маршрутизатора в OSPF-сети должен быть уникальный идентификатор. По умолчанию идентификатором маршрутизатора является адрес его петлевого интерфейса. Если адрес петлевого интерфейса не определен, то в качестве идентификатора маршрутизатора выбирается высший IP-адрес среди всех активных интерфейсов. Помните, что петлевой интерфейс маршрутизатора Cisco по определению всегда включен, а его IP-адрес уникален в сети (то есть это не может быть адрес 127.0.0.1).

Уполномоченный маршрутизатор

Чтобы сегменты могли обмениваться маршрутной информацией, в каждом сегменте должен быть уполномоченный маршрутизатор (Designated Router, DR). Протокол OSPF выбирает DR-маршрутизатор в каждом сегменте с множественным доступом. Когда широковещательная OSPF-рассылка поступает на DR-маршрутизатор, его задача состоит в том, чтобы разослать обновления на все маршрутизаторы в своей области. Это позволяет сократить OSPF-трафик до минимума, поскольку каждый маршрутизатор обменивается данными с единственным DR-маршрутизатором, от которого получает маршрутную информацию. В противном случае в широковещательную рассылку были бы вовлечены все маршрутизаторы, что, в свою очередь, требовало бы вновь и вновь организовывать широковещательную рассылку до тех пор, пока каждый маршрутизатор не получил бы сообщение. Другими словами, DR-маршрутизатор обеспечивает отношение «один ко многим» вместо «многие ко многим». Благодаря наличию DR-маршрутизатора обновления отправляются только на одно устройство, и единственный маршрутизатор отвечает за обновление всех маршрутизаторов внутри сегмента.

Также необходимо выделить резервный уполномоченный маршрутизатор (Backup Designated Router, BDR), который возьмет на себя функции DR-маршрутизатора, если тот станет недоступным. При недоступности и DR-, и BDR-маршрутизаторов новые уполномоченные маршрутизаторы выбираются автоматически.

Включение протокола OSPF

Протокол OSPF настраивается практически так же, как и остальные протоколы маршрутизации: используйте команду **router** для задания протокола и номера процесса, а также команду network, чтобы сообщить маршрутизатору, за какие сети отвечает этот протокол. Однако здесь легко запутаться: для протокола OSPF в команде network необходимо указать шаблонную маску, а не маску

подсети.

Пример:

router ospf 99 network 10.10.1.0 0.0.0.255 area 0

Здесь мы задаем процесс маршрутизации протокола OSPF с идентификатором процесса 99, ответственный за сеть 10.10.1.0/24, принадлежащую области 0. Следовательно, этот маршрутизатор является частью магистральной OSPF-области. Не путайте идентификатор процесса (99) с идентификатором области (0). OPSF работает на всех интерфейсах, соответствующих команде network.

Примеры OSPF-конфигураций

На рис. 7.2 показана сеть, в которой магистраль состоит из одного маршрутизатора. Также присутствуют три области, в каждой из которых по одному маршрутизатору. В этой конфигурации маршрутизатор 1 является магистральным, а маршрутизаторы 2, 3 и 4 - граничными маршрутизаторами области (ABR) с интерфейсами в других областях. Один из интерфейсов каждого из этих маршрутизаторов соединяется с магистральной областью.

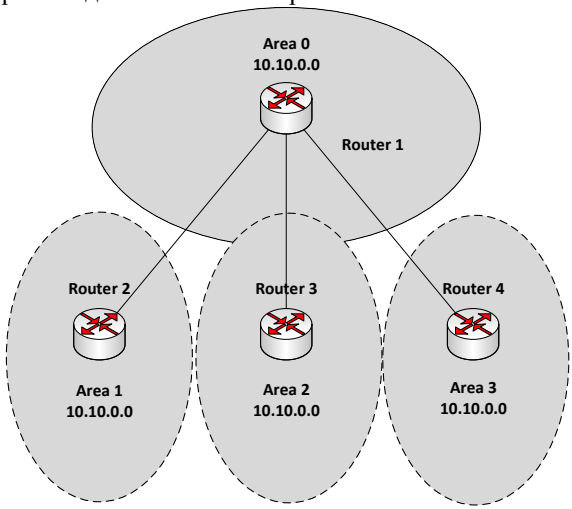


Рис. 7.2. OSPF-сеть с одним магистральным маршрутизатором

Конфигурирование маршрутизатора 1:

router ospf 99
network 10.10.0.0 0.0.255.255 area 0

Конфигурирование маршрутизатора 2:

router ospf 99
network 10.10.0.0 0.0.255.255 area 0
network 10.11.0.0 0.0.255.255 area 1

Конфигурирование маршрутизатора 3:

router ospf 99
network 10.10.0.0 0.0.255.255 area 0
network 10.12.0.0 0.0.255.255 area 2

Конфигурирование маршрутизатора 4:

router ospf 99
network 10.10.0.0 0.0.255.255 area 0
network 10.13.0.0 0.0.255.255 area 3

При конфигурировании каждого маршрутизатора для всех областей, в которые он входит, используется команда **network**. Маршрутизатору 1 команда **network** нужна только для области 0, так как у него нет интерфейсов в области 1**,** 2 или 3, его единственной областью является сеть 10.10.0.0 (область 0). Важно понимать, что областям принадлежат не маршрутизаторы, а интерфейсы.

Все остальные маршрутизаторы (2, 3 и 4) являются граничными маршрутизаторами области, поэтому для них необходимо указать по две команды **network**, задающие их участие в двух областях. В каждой команде **network** указывается подсеть, связанная с областью. Таким образом, в конфигурации маршрутизаторов 2, 3 и 4 для области 0, которая для них является магистральным подключением, указана подсеть 10.10.0.0 0.0.255.255 и также присутствуют записи для собственных областей (1, 2 и 3). Вместе в тем обратите внимание, что у всех маршрутизаторов один и тот же идентификатор OSPF-процесса (99). В отличие от номеров локальных автономных систем в протоколах IGRP и EIGRP идентификатор процесса за пределами маршрутизатора ни на что не влияет, и если бы в данном примере идентификаторы процесса были разными, то маршрутизация все равно работала бы.

Теперь немного изменим эту конфигурацию и рассмотрим вариант, в котором у нас нет уполномоченного магистрального маршрутизатора. На рис. 7.3 все три маршрутизатора принадлежат области 0, что делает их граничными маршрутизаторами области (магистральный маршрутизатор отсутствует).

Несмотря на это, магистраль в примере есть, просто у нее нет собственного маршрутизатора. Данная конфигурация более надежна, так как здесь мы отказались от единственного слабого звена.

Далее представлены команды конфигурирования устройств. Они практически не изменились, за исключением того, что из конфигурации был удален маршрутизатор 1, а маршрутизаторы 2, 3 и 4 теперь объединены одним сетевым сегментом.

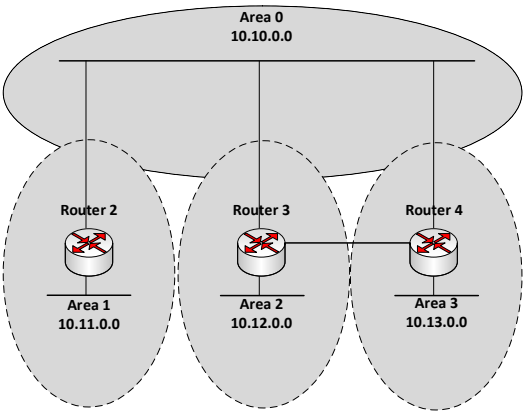


Рис. 7.3. OSPF-сеть с магистралью, проходящей через три ABR-маршрутизатора

Конфигурирование маршрутизатора 2:

```
router ospf 99
  network 10.11.0.0 0.0.255.255 area 1
  network 10.10.0.0 0.0.255.255 area 0
```

Конфигурирование маршрутизатора 3:

```
router ospf 99
  network 10.12.0.0 0.0.255.255 area 2
  network 10.10.0.0 0.0.255.255 area 0
```

Конфигурирование маршрутизатора 4:

```
router ospf 99
  network 10.13.0.0 0.0.255.255 area 3
  network 10.10.0.0 0.0.255.255 area 0
```

Объединение маршрутов в OSPF

Как и в случае с любым протоколом маршрутизации, объединение протоколов помогает уменьшить размер таблицы маршрутизации. В протоколе OSPF различают два типа объединения: внутреннее и внешнее по отношению к области.

Внутреннее объединение

Внутреннее объединение — это объединение ABR-маршрутизатором маршрутов в определенной области. ABR-маршрутизатор может объединять маршруты внутри своей области и за ее пределами при условии, что все подсети смежные (поразрядно) и допускают объединение. Тот факт, что области являются соседними, не означает, что адресация в них поддерживает объединение.

Внешнее объединение

Внешнее объединение осуществляется на ASBR-маршрутизаторах, причем объединяется вся сеть. Внешнее объединение можно применять при внесении в OSPF-сеть внешних маршрутов.

Виртуальные магистральные связи

Иногда у вас может не быть возможности создать непрерывную магистраль, например, по каким-то политическим или конструктивным причинам. Хотя протокол OSPF требует наличия одной подключенной магистрали, он также позволяет создать магистраль из двух отдельных областей. Такая магистраль называется виртуальной связью (virtual link).

7.3. Базовые настройки OSPF

Выбор идентификатора маршрутизатора (Router ID)

Router ID можно назначить административно выполнив команду:

name(config-router)#router-id <ip-address>.

Если RID не был назначен административно, то он выбирается автоматически, в зависимости от настроек маршрутизатора, по таким правилам.

1. Настроен один loopback-интерфейс и несколько интерфейсов с различными адресами:
   - адрес, присвоенный loopback-интерфейсу, будет Router ID.
2. Настроены несколько loopback-интерфейсов с несколькими IP-адресами в каждом:
   - наибольший IP-адрес, присвоенный любому из loopback-интерфейсов, будет Router ID.
3. Настроены несколько интерфейсов с IP-адресом на каждом:
   - наибольший IP-адрес из всех активных интерфейсов будет Router ID.

Перезапустить процесс OSPF можно командой:

name # clear ip ospf process.

Включение OSPF

Включить OSPF на интерфейсах в соответствующих сетях:

name (config)# router ospf <process-id>

name (config-router)# network <network> <wildcard mask> area <area-id>

Параметры команды **network**:

- <network> — непосредственно присоединенная сеть к маршрутизатору;
- <wildcard mask> — маска, которая указывает с помощью 0, какая часть из указанной сети должна совпадать, а с помощью 1- какая часть сети может быть произвольной;

– <area-id> — идентификатор зоны, в которой будет работать интерфейс маршрутизатора. Интерфейс попадет в эту зону при условии, что его IP-адрес совпадает с сетью, указанной с помощью network и wildcard mask. Для небольших сетей этот параметр можно указывать равным 0, но для больших сетей необходимо соблюдать иерархический дизайн зон в OSPF. Все обновления OSPF, которые передаются между различными зонами, должны проходить через зону 0.

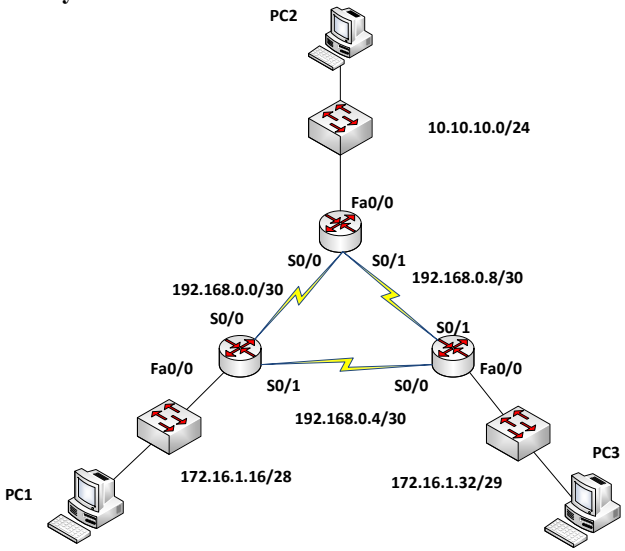
Команда network делает следующее:

– включает OSPF на интерфейсе, IP-адрес которого совпадает с указанной сетью и маской;
– анонсирует сеть этого интерфейса через другие интерфейсы, на которых включен OSPF.

7.4. Практическая часть

Данная лабораторная работа может быть выполнена на реальном оборудовании или в **Cisco Packet Tracer**. Все необходимые действия указаны по порядку их выполнения. Для начала выполнения лабораторной работы необходимо соединить физическую сеть в соответствии со схемой сети или построить соответствующий проект в **Cisco Packet Tracer**. Сразу после схемы сети в таблице указана схема адресации, которую нужно применять только тогда, когда это будет явно указано в тексте лабораторной работы.

Используемая топология



План адресации

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	Fa0/0	172.16.1.17	255.255.255.240	N/A
	S0/0	192.168.10.1	255.255.255.252	N/A
	S0/1	192.168.10.5	255.255.255.252	N/A
R2	Fa0/0	10.10.10.1	255.255.255.0	N/A
	S0/0	192.168.10.2	255.255.255.252	N/A
	S0/1	192.168.10.9	255.255.255.252	N/A
R3	Fa0/0	172.16.1.33	255.255.255.248	N/A
	S0/0	192.168.10.6	255.255.255.252	N/A
	S0/1	192.168.10.10	255.255.255.252	N/A
PC1	NIC	172.16.1.20	255.255.255.240	172.16.1.17
PC2	NIC	10.10.10.10	255.255.255.0	10.10.10.1
PC3	NIC	172.16.1.35	255.255.255.248	172.16.1.33

Порядок выполения работы

1. **Выполнить базовую конфигурацию R1,R2 и R3.**
   1.1. Настроить hostname.
   1.2. Отключить DNS lookup.
   1.3. Настроить privileged EXEC mode password.
   1.4. Настроить password для console 0.
   1.5. Настроить password для VTY 0 – 15.
   1.6. Базовые настройки оборудования Cisco описаны в предыдущих лабораторных работах.
2. **Выполнить конфигурацию сетевых интерфейсов R1,R2 и R3 согласно плану адресации.**

Для того чтобы быстро проверить сделанные настройки, можно воспользоваться командой Cisco CLI **show ip interface brief**, которая покажет установленные на интерфейсах ip-адреса и статус интерфейсов.

3. **Выполнить конфигурацию OSPF.**
4. **Проверить конфигурацию OSPF.**
5. **Изменить OSPF Router ID.**

Изменить OSPF Router ID можно с помощью добавления loopback интерфейса с адресом, значения которого больше, чем любого из назначенных интерфейсов маршрутизатора.

R1

R1(config)#interface loopback 0

R1(config-if)#ip address 10.1.1.1 255.255.255.255

R2

R2(config)#interface loopback 0

R2(config-if)#ip address 10.2.2.2 255.255.255.255

R3

R3(config)#interface loopback 0

R3(config-if)#ip address 10.3.3.3 255.255.255.255

Для того, чтобы новые параметры вступили в силу, нужно перезапустить **OSPF**. Наиболее простой способ сохранить конфигурацию маршрутизатора и выполнить команду **reload**.

Кроме того, OSPF **Router ID** может быть установлен принудительно.

R1(config)#router ospf 1

R1(config-router)#router-id 10.4.4.4

R1(config-router)#end

Для того, чтобы изменения вступили в силу, нужно перезапустить **OSPF**. Это можно сделать с помощью команды **clear.**

*R1#**clear ip ospf process***

Проверить сделанные настройки можно через **show ip protocols**

Отменить предыдущие настройки

R1(config)#router ospf 1

R1(config-router)#no router-id 10.4.4.4

R1(config-router)#end

Содержание отчета

Отчет о лабораторной работе должен содержать перечень команд, а также результатов их выполнения для каждого из пунктов порядка выполнения.

ЛАБОРАТОРНАЯ РАБОТА № 8

Базовая настройка протокола BGP

8.1. Цель работы

Целью данной работы является изучение принципов организации междоменной маршрутизации и основы функционирования протокола BGP-4, овладение практическими навыками базовой настройки протокола.

8.2. Теоретическая часть

Автономные системы

Крупные сети, такие как Intermet, организованы как множество доменов или автономных систем (autonomous system - AS). Такое разделение позволило логически провести административные и политические границы между различными организациями. Автономная система представляет собой набор маршрутизаторов, имеющих единые правила маршрутизации и управляемых одной технической администрацией. Для всей остальной сети AS является конечным простейшим элементом и воспринимается как единое целое. Реестром

сети Intemet либо провайдером услуг каждой AS назначается уникальный идентификационный номер, однако существует также диапазон номеров для частных AS: (64512-65535).

Маршрутизация между различными AS осуществляется с помощью протоколов внешнего шлюза. В настоящее время стандартом для организации междоменной маршрутизации в сети Internet является протокол BGP-4.

Принято различать два типа автономных систем: тупиковые (stub-network), или одноканальные (single-homed), и многоканальные (multi-homed), которые могут быть транзитными (рис. 8.3) или не-транзитными (рис. 8.2).

Автономная система называется тупиковой (рис.8.1), если все маршруты из нее в другие сети проходят через один канал. Весь внешний трафик, исходящий из такой AS, может отправляться только через одного провайдера.

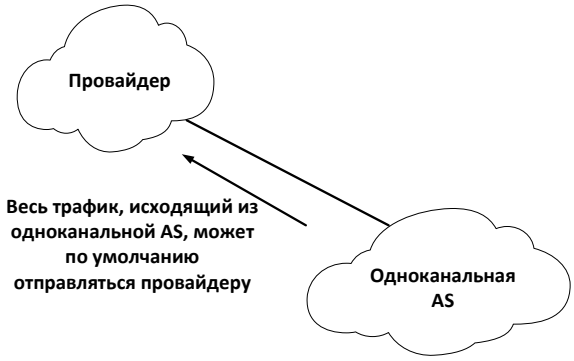


Рис. 8.1. Одноканальная (тупиковая) AS

Автономная система называется многоканальной, если в ней имеется более одной точки выхода во внешние сети. Автономная система может быть многоканальной по отношению к одному или к нескольким провайдерам.

В нетранзитных многоканальных AS не разрешается сквозной трафик через автономную систему. Транзитным (сквозным) по отношению к многоканальной AS является любой трафик, отправитель и получатель которого не принадлежат к данной AS.

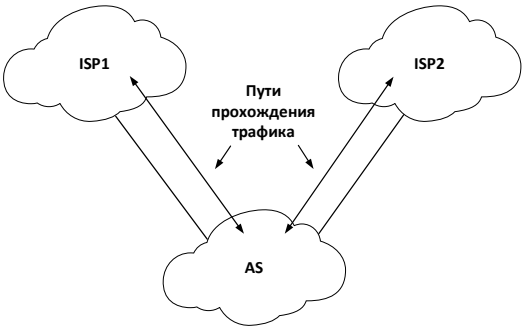


Рис. 8.2. Многоканальная нетранзитная AS

Многоканальные транзитные AS могут использоваться для транзита трафика в интересах других AS, а следовательно, BGP должен взаимодействовать с внутренними протоколами маршрутизации, действующими в пределах транзитных AS.

Рис. 8.3. Многоканальная транзитная AS

Протокол BGP

BGP - Border Gateway Protocol - является протоколом вектора маршрута и используется для обмена маршрутной информацией между автономными системами. Термин вектор маршрута (path vector) происходит из самого принципа действия BGP: маршрутная информация содержит последовательности номеров AS, через которые прошел пакет с заданным префиксом сети. Маршрутная информация, связанная с префиксом, используется для профилактики образования петель в маршрутах.

Протокол BGP не предъявляет никаких требований к топологии сети. Принцип его действия предполагает, что маршрутизация внутри автономной системы выполняется с помощью внутренних протоколов маршрутизации (IGP).

В качестве транспортного протокола BGP используют протокол TCP (порт 179). Таким образом, вся надежность доставки (включая повторную передачу) возлагается на протокол TCP и не требует отдельной реализации в самом BGP.

Рис. 8.4. Внешний и внутренний BGP

Маршрутизаторы, которые работают с протоколом BGP, называют спикерами BGP (BGP speakers). Два спикера BGP, образующих TCP соединение друг с другом для обмена маршрутной информацией, называют соседними (neighbors), или взаимодействующими (peers). Спикеры BGP могут находиться как в одной AS, в этом случае говорят о внутреннем BGP (Initemal BGP - IBGP), так и в разных AS, в этом случае говорят о внешнем BGP (Extemal BGP - EBGP) (рис. 8.4).

Одно из основных положений протокола BGP состоит в том, что взаимодействующие узлы устанавливают между собой сеансы связи. Если этот этап по каким-либо причинам не был выполнен, то обмен маршрутной информацией или ее обновление не произойдет. Переговоры с соседними узлами основаны на успешном установлении соединения по протоколу TCP, успешной обработке сообщения OPEN и периодическом обмене сообщениями UPDA TE и KEEP ALIVE.

Всего в ПРОТОКОЛЕ BGP есть 4 типа сообщений:

- OPEN (открытие соединения);
- UPDATE (обновление маршрутной информации);
- NOTIFICATION (уведомление об ошибке);
- KEEPALIVE (проверка состояния соединения).

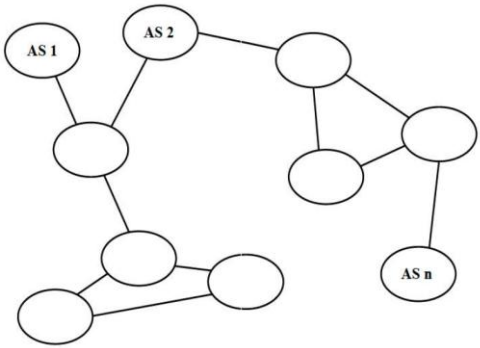


Рис. 8.5. Граф автономных систем

После установки сеанса связи между спикерами BGP ведется обмен всеми известными им маршрутами, которые могут далее использоваться в работе по протоколу BOP. Протоколом BGP на основе информации, полученной от различных маршрутизаторов, выстраивается граф автономных систем (рис. 8.5) со всеми связями между AS (такой граф иногда называют деревом), где каждой AS соответствует уникальный номер.

После того как соединение установлено и проведен начальный обмен маршрутами, по сети рассылается лишь информация об изменениях в таблице маршрутизации - так называемые инкрементные обновления (incremental updates) и сообщения поддержания соединения (keepalive). Пакеты KEEPALIVEE (длиной 19 байт каждый) не создают практически никакой нагрузки на процессор маршрутизатора и полосу пропускания, так как им требуется очень незначительная полоса пропускания (один 152-битовый пакет рассылается каждые 60 секунд, т.е. около 2,5 байт/с).

В случае если маршрут становится недействительным, т.е. по нему невозможно достичь пункта назначения, спикер BGP информирует об этом своих соседей и удаляет недействительный маршрут.

Кроме того, в памяти каждого BGP-маршрутизатора хранится номер версии таблицы BGP. Номер версии увеличивается на 1 всякий раз, когда BGP обновляет таблицу при изменении информации маршрутизации и совпадает у всех одноранговых узлов BGP. Быстрорастущие номера таблиц свидетельствуют о наличии нестабильных участков в сети.

Выбор маршрутов

Перед тем как переходить к конфигурированию протокола BGP, необходимо понять, какие метрики маршрутизации применяет этот

протокол. Выбор маршрутов в BGP основывается на большем объеме информации, чем в любом другом протоколе маршрутизации. Далее перечислены наиболее важные параметры, участвующие в выборе маршрута.

- **Вес (weight)**

Вес обозначает всего лишь локальный показатель предпочтительности маршрута. Вес назначается маршруту на определенном маршрутизаторе и используется только на этом устройстве. Вес никогда не назначается чужим маршрутам. Чем больше значение веса для маршрута, тем маршрут лучше. Вес можно настроить и применять для выбора лучшего среди нескольких маршрутов.

- **Локальное предпочтение (local preference)**

Локальное предпочтение - это еще один показатель, позволяющий выбрать лучший маршрут. В отличие от веса iBGP-маршрутизаторы обмениваются значениями локального предпочтения, однако эти значения не передаются на внешние BGP-маршрутизаторы. Значение по умолчанию для локального предпочтения равно 100. Как и с весом, чем больше значение, тем лучше маршрут.

- **Селектор выхода (Multi-Exit Discriminator, MED)**

MED-значения описывают маршруты внешним маршрутизаторам. В отличие от предпочтения и веса MED-значения реально передаются за пределы сети, чтобы сообщить соседним маршрутизаторам, по каким связям мы хотим обмениваться с ними информацией. В отличие от других метрик, чем меньше MED-значение, тем лучше маршрут. По умолчанию MED-значение равно нулю (О).

Название •селектор выхода• - неудачное и только запутывает. Разработчики протокола BGP рассуждали с точки зрения поставщика услуг Интернета: какой выход из ISP-сети следует выбрать, чтобы достичь нашей сети? С точки зрения нашей сети гораздо удобнее было бы назвать метрику селектором входа, то есть этой метрикой мы сообщаем поставщику услуг Интернета, какой из нескольких входов в нашу сеть ему использовать. MED-значения применяются только в том случае, если вы работаете с одним поставщиком услуг Интернета, но используете несколько физических линий связи.

- **AS-путь (AS path)**

BGP-маршрутизация основывается на списке автономных систем (AS), которые необходимо пройти, чтобы достичь точки назначения. Этот список и называется AS-путем. Предпочтительными являются более короткие AS-пути, однако существует множество способов

фильтрации маршрутов на базе их AS-путей. AS пути позволяют протоколу BGP распознавать маршрутные петли.

Вложение маршрутной информации в BGP

Есть несколько способов добавления маршрутной информации в протокол BGP: перераспределение IGP маршрутов в BGP; перераспределение статических маршрутов; командой **network.**

Динамические и статические маршруты могут быть автоматически перераспределены в протокол BGP. Кроме того, с помощью команды **network** можно выборочно подставить необходимые маршруты (как динамические, так и статические) в BGP. При этом предполагается, что сети, заданные командой **network**, существуют, и протокол BGP проверяет их доступность через таблицу маршрутизации.

Если протокол BGP не находит точного соответствия для заявленных сетей, то маршруты к ним не объявляются.

Команда **network** имеет следующий формат:

*Router(config-router)#**network** номер сети [**mask** маска сети]*

Для включения на маршрутизаторе протокола BGP необходимо в конфигурационном режиме вести команду **router bgp <AS>**, содержащую номер AS маршрутизатора. BGP-соседи указываются с помощью команды **neighbor <ip address> remote-as <AS>**, которая содержит номер AS, в которой находится соседний маршрутизатор, и ip-адрес его интерфейса. Для указания сетей, информация о которых должна включаться в ***update*** рассылку, в режиме конфигурирования BGP необходимо ввести команду:

network <ip address> mask <mask>, содержащую ip-адрес и маску сети.

Пример конфигурации BGP:

*Router# **configure terminal***

*Router(config)#**router bgp 100***

*Router(config-router)#**network** 192.168.42.0 mask 255 .255.255.0*

*Router(config-router)#**neighbor** 172.16.0.1 remote-as 200*

В табл. 8.1 представлены команды, используемые для мониторинга протокола BGP на маршрутизаторах Cisco.

Таблица 8.1
Команды мониторинга BGP

Router#**show ip bgp summary**	Вывод подробной информации о соединении с конкретным соседом
Router#**show ip bgp neighbors**	Вывод подробной информации о соединении с конкретным соседом
Router#**show ip bgp neighbors** *ip-address*	Вывод подробной информации о соединении с конкретным соседом
Router#**show ip bgp neighbors** *ip-address* **advertised-route**	Отображение всех сетей, анонсированных выбранному соседу
Router#**show ip bgp neighbors** *ip-address* **received-route**	Отображение всех сетей, полученных от выбранного соседа
Router#**show ip bgp**	Отображение всех полученных маршрутов
Router#**show ip bgp** *ip-address*	Отображение все маршрутов прохождение трафика до конкретного IP-адреса
Router#**debug bgp updates**	Режим отладки всех входящих и исходящих обновлений
Router#**debug bgp events**	Режим отладки всех событий BGP

8.3. Практическая часть

1. Соберите схему, изображенную на рис. 8.6, раздайте IP адреса и настройте виртуальные интерфейсы (табл. 8.2). Не забывайте о том, что изначально все физические интерфейсы роутеров находятся в выключенном состоянии и их необходимо включить с помощью команды no shutdown.

,

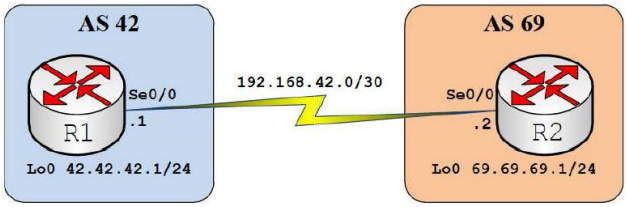


Рис. 8.6. Схема телекоммуникационной сети для выполнения лабораторной работы

Таблица 8.2
Исходные данные для выполнения лабораторной работы

Устройство	AS	Интерфейс	IP – адрес	Маска
R1	**42**	**Loopback 0**	**42.42.42.1**	**255.255.255.0**
		Serial 0/0	**192.168.42.1**	**255.255.255.252**
R2	**69**	**Loopback 0**	**69.69.69.1**	**255.255.255.0**
		Serial 0/0	**192.168.42.2**	**255.255.255.252**

2. С помощью команды **ping** проверьте правильность произведенной настройки.

3. Настройте протокол BGP на маршрутизаторах, R1 и R2 находятся в автономных системах 42 и 69 соответственно. Роутеры необходимо настроить таким образом, чтобы они рассылали маршрутную информацию о тех сетях, в которых находятся их loopback - интерфейсы.

*Rl#**configure terminal***
*Rl(config)#**router bgp 42***
*Rl(config-router)#**network 42.42.42.0 mask 255.255.255.0***
*Rl(config-router)#**neighbor 192.168.42.2 remote-as 69***
*R2#**configure terminal***
*R2(config)#**router bgp 69***
*R2(config-router)#**network 69.69.69.0 mask 255.255.255.0***
*R2(config-router)#**neighbor 192.168.42.1 remote-as 42***

Если всё настроено верно, по прошествии некоторого времени в консолях обоих роутеров появится системное сообщение, показывающее, что отношения соседства между роутерами установлены:

***Mar 1 00:07:28.667: %BGP-5- ADJCHANGE:neighbor 192.168.42.2 Up.**

4. С помощью команды **show ip bgp summary** проверьте верность произведенной конфигурации.

5. С помощью команды **show ip route bgp** проверьте наличие записей в таблице маршрутизации роутеров.

Содержание отчета

Отчет о лабораторной работе должен содержать перечень команд, а также результатов их выполнения для каждого из пунктов порядка выполнения.