

Министерство связи и массовых коммуникаций Российской Федерации

Государственное образовательное учреждение

высшего профессионального образования

ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ТЕЛЕКОММУНИКАЦИЙ И ИНФОРМАТИКИ

ЭЛЕКТРОННАЯ БИБЛИОТЕЧНАЯ СИСТЕМА

Самара

ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ

Государственное образовательное учреждение высшего профессионального образования

«Поволжский государственный университет телекоммуникаций и информатики»

Кафедра МСИБ

Методические разработки к лабораторной работе

Исследование протокола распределенного связующего дерева (STP)

для студентов специальностей 210406, 210400, 210403

Составители:

к.т.н., доц. Киреева Н. В.
асс. Буранова М.А.

Редактор:

д.т.н., проф. Зайкин В.П.

Рецензент:

д.т.н., проф. Васин Н.Н.

Самара 2010

Постановка задачи: Изучить работу протокола распределенного связующего дерева на коммутаторах Cisco Catalyst и произвести оптимизацию работы протокола, получение практических навыков работы с операционной системой коммутатора – *IOS*, использование анализатора *Wireshark*.

Приборы и материалы: Коммутаторы Cisco Catalyst, рабочие станции с ОС Windows, программное обеспечение Wireshark.

Необходимые знания и навыки:

- Отчётливое представление модели ВОС (OSI).
- Общие представления о коммутаторах и их соединениях.
- Общие понятия протокола распределенного связующего дерева, принципа работы, терминологию.
- Навыки работы с программой ping, telnet (HyperTerminal).
- Навыки работы с анализатором Wireshark или аналогичным.

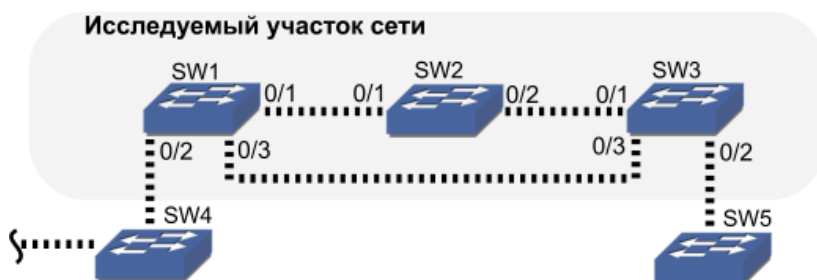
Ход работы

В данной работе производится исследование протокола распределенного связующего дерева (Spanning tree protocol). До начала работы следует изучить литературу по данной теме (список находится в приложении). Для выполнения работы студент должен представлять топологию исследуемой сети и все изменения, проводимые в ней, на момент выполнения заданий. Работа выполняется в одной из двух аудиториях с общей сетевой инфраструктурой, упрощенная топология представлена ниже:



Объектом исследования являются 3 коммутатора Cisco 2940 (SW1,SW2,SW3), работа остальных коммутаторов (SW4, SW5) не учитывается. Начиная работу, студент проверяет работоспособность сети командой ping и средствами ПО WireShark «захватывает» пакеты и наблюдает и анализирует их в окне программы.

Второй этап работы связан с изменением топологии сети. Администратор создает избыточные связи между исследуемыми коммутаторами. Топология сети на втором этапе:



После изменения топологии студент повторно проверяет работоспособность сети вышеописанными средствами.

На третьем этапе работа администратор включает протокол распределенного связующего дерева на исследуемых коммутаторах. Стоит отметить, что до этого момента поддержка данного протокола была отключена.

Студент выполняет в очередной раз проверку работоспособности сети и рассматривает кадр протокола STP, захваченный программой WireShark.

Каждый этап сопровождается выводами. Результатом выполнения лабораторной работы является оформленный отчет, сделанный в соответствии с пунктом «Выполнение работы».

Требования к отчёту.

Цель, приборы и материалы, топологии для всех этапов работы, таблицы и выводы полученные по ходу выполнения работы. Общий вывод о целесообразности применения STP, ограничениях и возможных проблемах его применения.

Выполнение работы.

Первый этап.

1. Получить от преподавателя IP-адреса соседних рабочих станций для последующего выполнения лабораторной работы.
2. Произвести проверку работоспособности локальной сети при помощи команды ping. Для этого
 - Открыть командную строку (cmd). «Пуск» – «Выполнить» - ввести «cmd», нажать «Ок», появится приветствие для ввода команд.
 - Выполнить команду **ping** ip-адрес. (подробнее о ping в приложении).

Пример выполнения:

```
C:\Documents and Settings\Пользователь>ping 192.168.0.111
```

Обмен пакетами с 192.168.0.111 по 32 байт:

Ответ от 192.168.0.111: число байт=32 время<1мс TTL=64

Ответ от 192.168.0.111: число байт=32 время<1мс TTL=64

Ответ от 192.168.0.111: число байт=32 время<1мс TTL=64

Ответ от 192.168.0.111: число байт=32 время<1мс TTL=64

Статистика Ping для 192.168.0.111:

Пакетов: отправлено = 4, получено = 4, потеряно = 0 (0% потерь),

Приблизительное время приема-передачи в мс:

Минимальное = 0мсек, Максимальное = 0 мсек, Среднее = 0 мсек

Программа послала 4 запроса (ICMP Echo-Request) и получила 4 ответа. Что свидетельствует о существовании соединения между рабочими станциями.

3. Загрузить Wireshark. Выбрать соответствующий сетевой адаптер из списка, для этого открыть меню «Capture», далее «Interfaces» и в открывшемся меню выбрать сетевой адаптер (например Atheros AR8121, название следует уточнить у преподавателя). Произвести действия аналогичные п.2. т.е. выполнить команду ping.

В основном окне программы можно увидеть строки (может отличаться):

No. .	Time	Source	Destination	Protocol	Info
265	63.080234	192.168.0.111	192.168.0.2	ICMP	Echo (ping) request
266	63.080252	192.168.0.2	192.168.0.111	ICMP	Echo (ping) reply

Source – отправитель пакета.

Destination – получатель пакета.

Protocol -протокол, которому принадлежит пакет.

Info – краткое пояснение о пакете.

Рассмотрим подробнее первый пакет. Он отправлен с рабочей станции с ip-адресом 192.168.0.11, адресат – 192.168.0.2, пакет принадлежит протоколу ICMP, в данном случае это пакет ping запрос.

Раскроем пункт Ethernet II.

```
 Ethernet II, Src: IntelCor_a9:43:1e (00:1c:c0:a9:43:1e), Dst: AsustekC_64:0f:14
  > Destination: AsustekC_64:0f:14 (00:22:15:64:0f:14)
  > Source: IntelCor_a9:43:1e (00:1c:c0:a9:43:1e)
  Type: IP (0x0800)
```

В нём указаны MAC-адреса участвующих в передаче рабочих станций.

Заполните таблицу, по ходу выполнения пп.2,3 для всех IP-адрес.

№ п /	IP-адрес отправ.	IP-адрес получателя	MAC адрес отправителя	MAC адрес получателя	Пакетов получено	Максимальный

п						вре мя при ёма- пере дач и, <i>ms</i>
1						
2						
3						

Сделать вывод о работоспособности сети.

4. В командной строке выполнить команду ping , где ip-адрес один из выше указанной таблицы.

ping ip-адрес -t

Второй этап.

5. Преподаватель изменяет топологию сети, внося избыточные связи и создавая физические петли, предварительно отключив протокол STP в коммутаторах командой

no spanning-tree

(подробнее о команде в приложении).

6. Наблюдая за отправкой-получением пакетов отметить момент, при котором после отправки пакетов не приходит ответ.
7. Осуществить проверку работоспособности сети как в первом этапе работы. Сделать соответствующие выводы.

Третий этап.

8. Преподаватель активирует протокол STP на коммутаторах командой

spanning-tree

(подробнее о команде в приложении).

9. Наблюдая за отправкой-получением пакетов отметить момент, после которого появляется ответ на запросы ping.
10. Закрыть окно командной строки.
11. Осуществить проверку работоспособности сети как в первом этапе работы. Сделать соответствующие выводы.
12. Рассмотрим кадр BPDU протокола STP. Для этого следует: Загрузить Wireshark. Выбрать соответствующий сетевой адаптер из списка, для этого открыть меню «Capture», далее «Interfaces» и в открывшемся меню выбрать сетевой адаптер (например Atheros AR8121, название следует уточнить у преподавателя). В основном окне программы будут отображаться «захваченные» кадры, выберем кадр принадлежащий

протоколу распределенного связующего дерева. (Protocol – STP) И рассмотрим основные его параметры.

No.	Time	Source	Destination	Protocol	Info
48	0.635160	Cisco_1a:14:82	PVST+	STP	conf. Root = 35296/00:12:43:1a:14:80 Cost = 0 Port = 0x8002

Source - отправитель кадра, в нашем случае это MAC-адрес того коммутатора, к порту которого подключен наш сетевой адаптер.

Destination – в данном случае здесь указывается не адрес получателя, а режим работы STP, в нашем случае это per VLAN Spanning Tree (отдельный экземпляр STP для каждого VLAN), что вполне стандартно для коммутатора Cisco 2940 series.

Protocol – протокол которому предназначен данный кадр.

Info – для удобства использования в нем могут быть помещены не только краткие уточнения или пояснения, как в случае с ICMP, но и отображены наиболее важные параметры. Conf.root, идентификатор корневого моста (Bridge ID), из двух составных частей приоритета и адреса разделенных символом /. Затем следует стоимость маршрута и номер порта, на который был отправлен данный кадр BPDU.

Заполните таблицу для «захваченного» кадра BPDU и поместите её в отчёт.

Отправитель кадра, MAC	Режим работы STP	Идентификатор корневого моста	Стоимость маршрута

13. Далее производится исследование и построение топологии сети. Для этого студенту предоставляется telnet доступ к каждому коммутатору. Студент выполняет команду

show spantree 1 active

на каждом коммутаторе, анализируя полученную информацию строит топологию сети (коммутаторов). К карте необходимо отобразить все задействованные порты, их состояния, MAC-адреса коммутаторов и их приоритеты, «цены» маршрутов. Окончательно необходимо отметить корневой мост (коммутатор). Подробное описание команды и комментарии к выводимой ей информации, а также примеры построения топологии со всеми обозначениями находятся в приложении

14. По результатам построения топологии нарисовать схему дерева – результат работы протокола STP (исключить заблокированные соединения).
15. Сделать выводы о работе протокола STM и о причинах выхода из строя сети при его отключении. Все пояснения аргументировать пактами выполненной лабораторной работы и построенной топологией и схемой.

Вопросы и задания.

1. Общепонимание алгоритма распределенного связующего дерева. Принадлежность к уровню ВОС.
2. Причины необходимости использования данного алгоритма и реализуемого им протокола.
3. Возможные последствия при отключенном протоколе в топологии сети, представленной в данной работе.
4. Общие понятия и термины, используемые в алгоритме. (Сегмент, корневой мост, корневой порт, назначенный порт их идентификаторы).
5. Первый этап конфигурирования STP.
6. Второй этап конфигурирования STP.
7. Третий этап конфигурирования STP.
8. Кадр, используемый протоколом STP. Наиболее важные параметры.
9. Понятие стоимости маршрута как метрики принятые стандарты и принцип расчета расстояния.
10. Основные настраиваемые параметры протокола.
11. Рекомендации по оптимизации работы STP.
12. Недостатки STP.
13. Краткое пояснение принципа работы протокола для топологии, используемой в данной работе.
14. Краткая характеристика исследуемых в данной работе коммутаторов.

Приложение.

1. Общие сведения об Catalyst 2940.



WS-C2940-8TF-S - это малый настольный управляемый коммутатор **Cisco Catalyst 2940-8TF-S** из серии Cisco Catalyst 2940. **Cisco Catalyst 2940** - серия малых, настольных, управляемых коммутаторов компании Cisco, с восьмью портами Fast Ethernet и с интегрированным uplink-портом Fast Ethernet или Gigabit Ethernet. Коммутаторы **Cisco Catalyst 2940** малошумные, легко устанавливаются и настраиваются благодаря таким функциям как Cisco Express Setup и настройке через Web. Коммутаторы **Cisco Catalyst 2940** предназначены для использования на рабочих местах конечных пользователей, например, в аудиториях, конференц-залах и небольших рабочих группах. В серию Cisco Catalyst 2940 входят: Catalyst 2940-8TT-S, Catalyst 2940-8TF-S

Основные характеристики коммутаторов серии Cisco Catalyst 2940

	Catalyst 2940-8TT-S	Catalyst 2940-8TF-S
Количество портов Fast Ethernet	8 (10/100TX)	8 (10/100TX) + 1 (100FX)
Количество портов Gigabit Ethernet	1 (10/100/1000T)	0
Количество портов Gigabit Ethernet SFP	0	1
Пропускная способность, Гбит/с	3,6	
Тип транков VLAN	802.1q	
Объем flash-памяти, Мб	8	
Объем ОЗУ, Мб	16	

Функциональные особенности серии коммутаторов Cisco Catalyst 2940:

- **Разработаны с учетом потребностей конечных пользователей:** коммутаторы серии Catalyst 2940 отличаются компактными размерами, бесшумностью работы, гибкими возможностями монтажа (есть как крепежные отверстия, так и магнитное крепление), повышенной физической безопасностью (благодаря металлическому корпусу и защитным фиксаторам корпуса коммутатора и кабелей Ethernet).
- **Высокая безопасность:** поддерживаются протокол 802.1x, функциональность Port Security, динамическое назначение VLAN при аутентификации на сервере VMPS, Private VLAN Edge, присутствует дуплексный порт SPAN.
- **Удобство установки и управления:** коммутатор имеет функции автоматической загрузки конфигурации по протоколу DHCP, функциональность ExpressSetup, поддержка SNMP v.1, 2, 3 и Telnet, RMON, SPAN, NTP, TFTP.

- **Функциональность качества обслуживания (QoS):** поддерживается классификация трафика по полю 802.1p (CoS), маркирование входящих пакетов, приоритетная очередность (Strict Priority) и очередность WRR для исходящих пакетов.

Производительность

- Максимальная пропускная способность - 3,6 Гб/с
- Неблокируемая коммутация на скорости 2,7 миллионов пакетов/с (размер пакетов 64 байта)
- Поддерживаются технологии и протоколы: Per VLAN Spanning Tree Plus (PVST+), VLAN Trunking Protocol (VTP) pruning, Internet Group Management Protocol (IGMP) Snooping, IGMP immediate-leave processing, Multicast VLAN Registration (MVR)

Программа ping.

С помощью отправки сообщений с эхо-запросом по протоколу ICMP проверяет соединение на уровне протокола IP с другим компьютером, поддерживающим TCP/IP. После каждой передачи выводится соответствующее сообщение с эхо-ответом. Ping - это основная TCP/IP-команда, используемая для устранения неполадки в соединении, проверки возможности доступа и разрешения имен.

Синтаксис

```
ping [-t] [-a] [-n счетчик] [-l размер] [-f] [-i TTL] [-v тип] [-r счетчик] [-s счетчик]
[{-j список_узлов | -k список_узлов}] [-w интервал]
[имя_конечного_компьютера]
```

-t Задаёт для команды ping отправку сообщений с эхо-запросом к точке назначения до тех пор, пока команда не будет прервана. Для прерывания команды и вывода статистики нажмите комбинацию CTRL-BREAK. Для прерывания команды ping и выхода из нее нажмите клавиши CTRL-C.

-a Задаёт разрешение обратного имени по IP-адресу назначения. В случае успешного выполнения выводится имя соответствующего узла.

-n *счетчик* Задаёт число отправляемых сообщений с эхо-запросом. По умолчанию — 4.

-l *размер* Задаёт длину (в байтах) поля данных в отправленных сообщениях с эхо-запросом. По умолчанию — 32 байта. Максимальный *размер* — 65527.

-f Задаёт отправку сообщений с эхо-запросом с флагом «Don't Fragment» в IP-заголовке, установленном на 1. Сообщения с эхо-запросом не фрагментируются маршрутизаторами на пути к месту назначения.

-i TTL Задаёт значение поля TTL в IP-заголовке для отправляемых сообщений с эхо-запросом. По умолчанию берётся значение TTL, заданное по умолчанию для узла.

-j список_узлов Указывает для сообщений с эхо-запросом использование параметра свободной маршрутизации в IP-заголовке с набором промежуточных точек назначения, указанным в *списке_узлов*.

-k список_узлов Указывает для сообщений с эхо-запросом использование параметра строгой маршрутизации в IP-заголовке с набором промежуточных точек назначения, указанным в *списке_узлов*.

-w интервал Определяет в миллисекундах время ожидания получения сообщения с эхо-ответом, которое соответствует сообщению с эхо-запросом.

2. Краткая теория STA.

В тех локальных сетях, где технологии и оборудование реализуют функции только первого и второго уровней модели ВОС (OSI), проблема использования альтернативных путей имеет свою специфику: базовые протоколы поддерживают только *древовидные*, то есть не содержащие замкнутых контуров, топологии связей.

Для автоматического перевода в резервное состояние всех альтернативных связей, не вписывающихся в топологию дерева, в локальных сетях используются **алгоритм покрывающего дерева** (Spanning Tree Algorithm, STA) и реализующий его **протокол покрывающего дерева** (Spanning Tree Protocol, STP).

Алгоритм покрывающего дерева, разработанный достаточно давно, в 1983 году, был признан IEEE удачным и включен в ту же спецификацию 802.1D, в которой описывается и сам алгоритм работы прозрачного моста. Хотя мосты, для которых был разработан алгоритм, сегодня уже относятся к практически «вымершему» виду коммуникационных устройств, STA широко применяется в наиболее массовых устройствах современных локальных сетей — коммутаторах. Алгоритм STA позволил без применения маршрутизаторов только на основе коммутаторов строить крупные локальные сети, за счёт избыточных связей обладающие высокой надёжностью.

Производители оборудования, как правило, реализуют алгоритм STA в коммутаторах, предлагаемых для тех участков сети, к которым предъявляются повышенные требования к надёжности, — в магистральных коммутаторах, а также в коммутаторах отделов и крупных рабочих групп.

Необходимые определения.

Алгоритм STA формализует сеть (рис. 1, а) в виде графа (рис. 1), вершинами которого являются коммутаторы и сегменты сети.

Сегмент — связная часть сети, не содержащая коммутаторов (и маршрутизаторов). Сегмент может быть разделяемым (во время создания алгоритма STA это был единственный тип сегмента) и включать устройства физического уровня: повторители/концентраторы, существование которых коммутатор, будучи устройством канального уровня, «не замечает». Сегодня сегмент часто представляет собой дуплексный двухточечный канал между смежными портами двух коммутаторов.

Алгоритм покрывающего дерева обеспечивает построение древовидной топологии связей с единственным путем минимальной длины от каждого коммутатора и от каждого сегмента до некоторого выделенного **корневого коммутатора** — корня дерева. *Единственность* пути гарантирует отсутствие петель, а *минимальность* расстояния — рациональность маршрутов следования трафика от периферии сети к ее магистрали, роль которой исполняет корневой коммутатор.

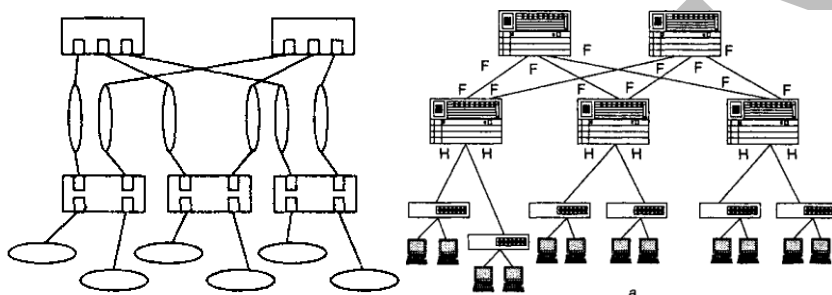


Рис. 1. Формализованное представление сети в соответствии с алгоритмом STA

Корневой порт коммутатора — порт, который имеет кратчайшее расстояние до корневого коммутатора (точнее, до любого из портов корневого коммутатора).

Идентификатором порта служит 2-байтовое число. Младший байт содержит порядковый номер данного порта в коммутаторе, а значение старшего байта задается администратором.

Назначенный порт — порт, который среди всех портов всех коммутаторов данного сегмента сети имеет минимальное расстояние до корневого коммутатора.

Назначенным коммутатором сегмента объявляется коммутатор, которому принадлежит назначенный порт данного сегмента.

Протокольными единицами данных моста (Bridge Protocol Data Unit, BPDU) называются специальные пакеты, которыми периодически

обмениваются коммутаторы для автоматического определения конфигурации дерева. Пакеты BPDU переносят данные об идентификаторах коммутаторов и портов, а также о расстоянии до корневого коммутатора. Интервал генерации пакетов BPDU, называемый в алгоритме *интервалом hello*, настраивается администратором и обычно составляет от 1 до 4 секунд.

Три этапа построение дерева.

Алгоритм STP определяет активную конфигурацию сети за три этапа.

Первый этап — определение корневого коммутатора, от которого строится дерево.

В соответствии с алгоритмом STP в качестве корневого коммутатора выбирается коммутатор с *наименьшим значением идентификатора*. Если администратор не вмешивается в этот процесс, корневым коммутатором будет выбран достаточно случайным образом — им станет устройство с минимальным MAC-адресом блока управления. Очевидно, что такой выбор может оказаться далеко не рациональным. Например, при выборе коммутатора 5 в качестве корневого значительная часть трафика проходила бы через большое количество транзитных сегментов и коммутаторов. Поэтому не стоит администратору пускать данный процесс «на самотек» — лучше в него вмешаться и назначить корневой коммутатор осознанно (за счет соответствующего конфигурирования старших байтов идентификаторов коммутатора), чтобы выбранный коммутатор действительно занимал центральное место в соединениях сегментов.

Второй этап — выбор корневого порта для каждого коммутатора.

Расстояние определяется по пакетам BPDU, поступающим от корневого коммутатора. На основании этих пакетов каждый коммутатор может определить минимальные расстояния от всех своих портов до корневого коммутатора. Каждый коммутатор анализирует и ретранслирует BPDU, увеличивая расстояние до корня, указанное в полученном пакете BPDU, на условное время того сегмента, из которого принят данный пакет. Тем самым в пакете BPDU по мере прохождения через коммутаторы наращивается расстояние до корневого коммутатора.

Ретранслируя пакеты, каждый коммутатор для каждого своего порта запоминает минимальное расстояние до корня, встретившееся во всех принятых этим портом пакетах BPDU. По завершении процедуры определения конфигурации покрывающего дерева каждый коммутатор находит свой корневой порт (с минимальным расстоянием до корня).

При равных метриках для разрешения неоднозначности к процедуре выбора минимального расстояния привлекаются значения идентификаторов коммутаторов и портов. Предпочтение отдается портам и коммутаторам с наименьшими идентификаторами.

Третий этап — выбор назначенных порта и коммутатора.

Из всех портов всех коммутаторов в пределах каждого сегмента сети выбирается назначенный порт и соответствующий данному порту назначенный

коммутатор сегмента. Аналогично выбору корневого порта здесь используется распределенная процедура. Каждый коммутатор сегмента прежде всего исключает из рассмотрения свой корневой порт (для сегмента, к которому он подключен, всегда существует другой коммутатор, который расположен ближе к корню). Для каждого из оставшихся портов выполняется сравнение принятых по ним минимальных расстояний до корня (еще до наращивания на условное время сегмента) с расстоянием до корня корневого порта данного коммутатора. Если все принятые на этом порту расстояния оказываются больше, чем расстояние от собственного корневого порта, значит, для сегмента, к которому подключен порт, кратчайший путь к корневому коммутатору проходит через него, и он становится назначенным. Коммутатор делает все свои порты, для которых такое условие выполняется, назначенными. Когда имеется несколько портов с одинаковым кратчайшим расстоянием до корневого коммутатора, выбирается порт с наименьшим идентификатором.

На выполнение всех трех этапов коммутаторам сети отводится по умолчанию 15 секунд. Предполагается, что за это время каждый коммутатор получит столько пакетов BPDU, сколько будет достаточно для определения состояния своих портов.

Все остальные порты, кроме корневых и назначенных, блокируются (на рисунке они перечеркнуты), и в результате завершается построение покрывающего дерева. Математически доказано, что при таком выборе активных портов в сети исключаются петли, и оставшиеся связи образуют *покрывающее дерево* (если оно вообще может быть построено при существующих связях в сети).

После построения покрывающего дерева коммутатор начинает принимать (но не продвигать) пакеты данных и на основе их адресов источника строить таблицу продвижения. Это обычный режим обучения прозрачного моста, который ранее нельзя было активизировать, так как порт не был уверен в том, что он останется корневым или назначенным и будет передавать пакеты данных. Состояние обучения по умолчанию также выдерживается в течение интервала 15 с. При этом порт продолжает участвовать в работе алгоритма STA, так что поступление пакетов BPDU с лучшими параметрами переводит его в заблокированное состояние.

И только после двукратной выдержки по таймеру порт переходит в состояние продвижения и обрабатывает пакеты данных в соответствии с построенной таблицей (которая продолжает модифицироваться, отражая изменения в структуре сети).

В процессе нормальной работы корневой коммутатор продолжает генерировать конфигурационные пакеты BPDU с интервалом hello, а остальные коммутаторы получают их через свои корневые порты и ретранслируют через назначенные порты. У коммутатора могут отсутствовать назначенные порты, как у коммутаторов 2 и 4, но он все равно участвует в работе протокола STA, так как корневой порт принимает служебные пакеты BPDU.

Если по истечении максимального времени жизни сообщения (по умолчанию — 20 с) корневой порт любого коммутатора сети не получает служебный пакет BPDU, то он инициализирует новую процедуру построения покрывающего дерева. При этом на все порты генерируется и передается пакет BPDU, в котором коммутатор указывает себя в качестве корневого. Аналогичным образом ведут себя и другие коммутаторы сети, у которых истек таймер максимального времени жизни сообщения, в результате чего выбирается новая активная конфигурация.

Недостатки и достоинства STA

Одним из основных достоинств алгоритма покрывающего дерева является то, что, в отличие от многих упрощенных алгоритмов, где переход в резервное соединение осуществляется исключительно при отказе соседнего устройства, он принимает решение о реконфигурировании с учетом не только связей с соседями, но и связей в отдаленных сегментах сети.

К недостаткам алгоритма можно отнести то, что в сетях с большим количеством коммутаторов время определения новой активной конфигурации может оказаться слишком большим. Если в сети используются заданные по умолчанию значения тайм-аутов, переход на новую конфигурацию может занять свыше 50 секунд: 20 секунд понадобится на констатацию факта потери связи с корневым коммутатором (истечение таймера — единственный способ узнать об этом событии в стандартном варианте STA) и еще 2x15 секунд потребуется для перехода портов в состояние продвижения.

Имеющиеся многочисленные нестандартные версии STA позволяют сократить время реконфигурирования за счет усложнения алгоритма, например добавления новых типов служебных сообщений. В 2001 году разработана стандартная ускоренная версия STA (спецификация IEEE 802.1w).

4. Команды коммутатора.

Cisco IOS (*Internetwork Operating System*) — программное обеспечение, используемое в маршрутизаторах Cisco, и некоторых сетевых коммутаторах. Cisco IOS это многозадачная операционная система выполняющая функции сетевой организации, маршрутизации, коммутации и передачи данных.

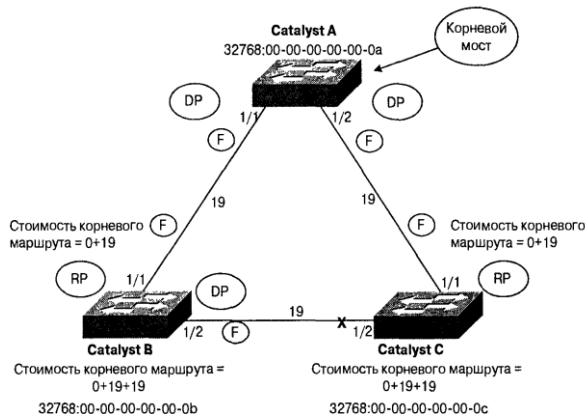
Cisco IOS имеет специфичный интерфейс командной строки (*command line interface, CLI*), который был скопирован многими другими сетевыми продуктами. Интерфейс IOS имеет набор многословных команд, доступные команды определены «режимом» и уровнем привилегий данного пользователя.

[no] spanning-tree [vlan vlan] (*режим глобальной конфигурации*)

Протокол STP стандартно включен в сети VLAN 1 и в любых недавно созданных VLAN-сетях. Если сеть VLAN не указана, то протокол STP включается или отключается во всех VLAN-сетях. Следует помнить о том, что

при отключенном механизме STP мостовые петли не обнаруживаются, а их возникновение не предотвращается.

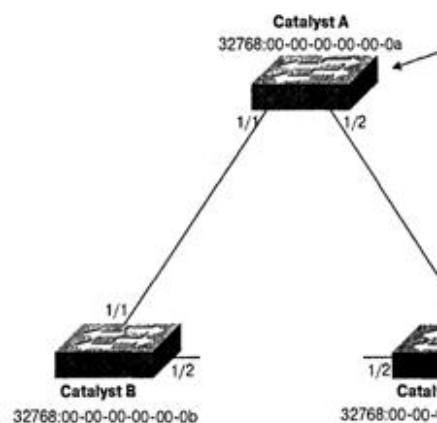
5. Построенной топологии сети.



Скорость порта	Стандартная стоимость порта в "сокращенном режиме"	Стандартная стоимость порта в "расширенном режиме"
10 Мбит/с	100	2 000 000
100 Мбит/с	19	200 000
1 Гбит/с	4	20 000

Корневые порты отмечаются литерами **RP**, назначенные порты – **DP**, метки **A** соответствуют портам в режиме передачи, а **X** – портам, которые находятся в состоянии блокировки.

6. Пример построенный схемы дерева.



10 Гбит/с	2	2000
-----------	---	------

Значения стоимости маршрутов

Стандартно порты коммутатора обладают определенными значениями стоимости маршрутов, которые указаны в таблице.

7. Навигация по топологии распределенного связующего дерева.

Указание местоположения корневого моста.

а) Выбор коммутатора, используемого в качестве начальной точки.

В идеале желательно начать с корневого моста на "вершине" STP-иерархии. Если неизвестно, какой коммутатор является корневым для данной VLAN-сети, то в качестве начальной точки можно выбрать любой.

б) Отображение корневого идентификатора (Root ID), локального BID-идентификатора и корневого порта.

```
show spanning-tree vlan vlan
```

```
Switch#show spanning-tree
VLAN0001
Spanning tree enabled protocol ieee
Root ID Priority 32769
Address 0001.C769.6BD6
Cost 19
Port 2(FastEthernet0/2)
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)
Address 0050.0F06.39C3
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 20
```

Interface	Role	Sts	Cost	Prio.Nbr	Type
Fa0/1	Desg	FWD	19	128.1	P2p
Fa0/2	Root	FWD	19	128.2	P2p
Fa0/3	Desg	FWD	19	128.3	P2p

в) Переход от корневого порта к корневному мосту.

Напомним, что в коммутаторе имеется только один корневой порт и этот порт ведет к корневному мосту. Коммутатор может иметь множество назначенных

портов, которые ведут от корневого моста. Нужно найти соседний коммутатор, который подключен к корневому порту.

```
Switch>en
Switch#show spanning-tree
VLAN0001
Spanning tree enabled protocol ieee
Root ID Priority 32769
    Address 0001.C769.6BD6
    This bridge is the root
    Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)
    Address 0001.C769.6BD6
    Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
    Aging Time 20

Interface Role Sts Cost Prio.Nbr Type
-----
Fa0/1 Desg FWD 19 128.1 P2p
Fa0/2 Desg FWD 19 128.2 P2p
Fa0/3 Desg FWD 19 128.3 P2p
```

Рассмотрим подробнее результаты выполнения команды.

```
VLAN 1
Spanning tree enabled
Spanning tree type ieee
Designated Root aa-aa-aa-aa-aa-aa
Designated Root priority 32768
Designated Root Cost 19
Designated Root port 1/1
Root Max Age 20 sec Hello Time 2 sec Forward Delay 15 sec
Bridge ID MAC ADDR bb-bb-bb-bb-bb-bb
Bridge ID priority 32768
Bridge ID Max Age 20 sec Hello Time 2 sec Forward Delay 15 sec
Port Vlan Port-State Cost priority Fast-Start Group-method
1/1 1 forwarding 19 32 disabled
1/2 2 forwarding 19 32 disabled
```

В первых двух строках статистики по корневому мосту указан идентификатор BID текущего корневого моста. Поля BID показаны отдельно: Designated Root (назначенное корневое устройство) — MAC-адрес моста (последние 6 байт BID), а Designated Root Priority (назначенный приоритет корневого моста) — значение приоритета моста (первые 2 байта BID). Общая стоимость маршрута к корневому мосту представлена в поле Designated Root Cost. Четвертая строка — Designated Root Port (назначенный корневой порт), указывает, какой порт в данный момент имеет статус корневого. Последняя строка статистики корневого моста указывает заданные на корневом мосту значения таймеров. Как уже говорилось в предыдущем разделе, значения таймеров имеют силу для всей сети (или по крайней мере для VLAN 1), что обеспечивает устойчивость структуры. Термин Designated (назначенный) в данном случае используется, чтобы указать принадлежность выводимых значений корневому порту. Следует отметить, что вследствие изменения топологии и задержек при распространении информации в течение времени, которое соответствует сходимости

алгоритма, информация на отдельном коммутаторе не соответствует достоверным характеристикам корневого моста.

Локальная статистика моста в первых двух строках содержит значение идентификатора VID локального моста, а в третьей строке представлены текущие значения таймеров.

Статистика по портам моста представлена в нижней части примера. В зависимости от количества портов коммутатора Catalyst, статистика может растянуться на несколько экранов, разделенных обозначением тоге (далее).

В такой статистике можно найти стоимости маршрутов для каждого порта. Значение стоимости маршрута каждого порта складывается со значением корневой стоимости, полученным в сообщениях BPDU. Другими словами, пусть порт 1/1 коммутатора Cat-B получает сообщения BPDU от корневого моста со значением стоимости маршрута, равным 0. Значение стоимости 19, записанное для порта 1/1, добавляется к нулевому значению, полученному от корневого моста, результирующая величина 19; сохраняется как значение Designated Root Cost. Коммутатор Cat-B отправляет в нижний сегмент сообщения BPDU со значением стоимости маршрута 19 — значение стоимости пути 19, записанное для порта 1/2, не добавляется к отправляемому значению.

Литература

1. В. Г. Олифер, Н. А. Олифер, «Компьютерные сети. Принципы, технологии, протоколы», учебник для вузов, издательство «Питер», 2007 г., 960 с.
2. Дэвид Хьюкаби, Стив Мак-Квери, «Руководство Cisco по конфигурированию коммутаторов Catalyst», издательство «Вильямс», 2004, 553 с.
3. Кеннеди Кларк, Кевин Гамильтон, «Принципы коммутации в локальных сетях Cisco (Cisco Lan Switching)», издательство «Вильямс», 2003, 654 с.
4. Официальный сайт Cisco systems, «Understanding Spanning-Tree Protocol», статья,
http://www.cisco.com/univercd/cc/td/doc/product/rtrmgmt/sw_ntman/cwsi2/cwsiug2/vlan2/stpapp.htm