

Лабораторная работа № 4

ИЗУЧЕНИЕ КОММУТИРУЕМОЙ СЕТИ ETHERNET

Цель работы

Изучить базовые принципы протокола канального уровня STP, научиться анализировать алгоритм работы протокола.

Современные компьютерные сети, основанные на технологии Ethernet, строятся применением коммутаторов – устройств, отвечающих за формирование физической телекоммуникационной среды, создание каналов и передачу по ним данных (фреймов). Все коммутаторы имеют ограниченное количество портов, к которым подключаются сетевые устройства (см. рисунок 1). Каждый порт имеет ограниченную пропускную способность (10/100/1000/10000 МБит/с).



Рисунок 1 – Коммутаторы

Для построения сетей, в которых требуется большое количество сетевых портов могут использоваться коммутаторы, объединяемые в единое логическое устройство - стек. В таких коммутаторах для их объединения используются специальные порты с пропускной способностью в разы или даже на порядки выше, чем пропускная способность рабочих интерфейсов.

На практике построение больших сетей, центральным элементом которых является один большой коммутатор, не является рациональным. Во-первых, в таких сетях все сетевые устройства должны быть напрямую подключены к одному коммутатору. В результате длинна кабельной подсистемы резко увеличивается. Во-вторых, стоимость коммутаторов, способных объединяться стек, значительно выше, т.к. в них требуется реализовать дополнительный функционал и создать специальные интерфейсы. В-третьих, в таких

сетях допускается использование коммутаторов только одного производителя (и, возможно, даже только одной модели).

Широкое распространение получил способ построения компьютерных сетей использованием нескольких коммутаторов, соединяемых между собой по обычным каналам (т.е. по тем же, что используются для подключения сетевых устройств) или по каналам с более высокой пропускной способностью (например, для сетевых устройств используются каналы 1 Гбит/с, а для соединения коммутаторов – 10 Гбит/с). В таких сетях каждый коммутатор функционирует как самостоятельное устройство и использует другие коммутаторы как промежуточные узлы для передачи данных.

В больших сетях, построенных на нескольких коммутаторах, принято выделять несколько административных уровней, например, доступа, распределения и ядра сети (см. рисунок 2). Непосредственное подключение сетевых устройств осуществляется на уровне доступа. Примером такого уровня может служить один подъезд жилого дома (или весь дом в целом). Коммутаторы уровня доступа соединяются с коммутаторами уровня распределения. Например, это могут быть коммутаторы одного района. А коммутаторы уровня доступа уже соединяются с коммутаторами уровня ядра сети. Обычно такие коммутаторы располагаются в дата-центрах провайдера. Очевидно, что производительность коммутаторов уровня распределения должна быть выше, чем коммутаторов уровня доступа. Аналогичная ситуация и с коммутаторами уровня ядра сети.

Следует отметить, что при построении сетей на базе технологии Ethernet с использованием нескольких коммутаторов должно выполняться условие – между двумя любыми узлами сети не должно быть больше 15 коммутаторов (ограничение на широкоэвентельный домен). В этом случае гарантируется, что скорость передачи данных во всей сети будет одинаковой. В противном случае из-за физических особенностей распространения сигналов и производительности самих коммутаторов скорость может резко снизиться.

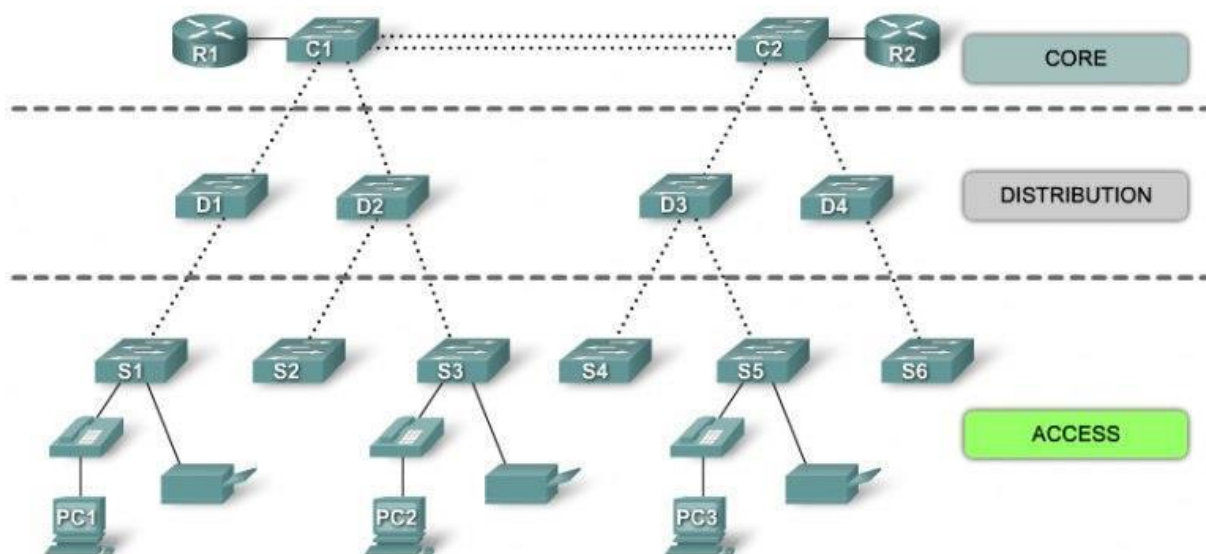


Рисунок 2 – Пример многоуровневой архитектуры локальной компьютерной сети

Цифровой (широковещательный) шторм

В сетях, в которых используется два и более коммутаторов, может возникнуть их циклическое соединение (так называемая петля). Такая ситуация может привести всю сеть в нерабочее состояние. Рассмотрим эту ситуацию подробнее.

Предположим, что в сети имеется два коммутатора, соединенных друг с другом двумя каналами связи (см. рисунок 3). Пусть ПК1, подключенный к коммутатору 1, инициирует передачу данных до ПК2, подключенного к коммутатору 2. В служебных таблицах коммутаторов ещё нет записей ни о ПК1, ни о ПК2. Получив фрейм от ПК1, коммутатор 1 анализирует свою таблицу MAC адресов и обнаруживает, что он «не знает» к какому из его интерфейсов подключен получатель этого фрейма (при этом запоминает, что отправитель подключен через соответствующий канал). В результате коммутатор 1 ретранслирует полученный фрейм во все свои интерфейсы, кроме того, через который этот фрейм был получен (к которому подключен ПК1). Фрейм от ПК1 будет передан по каналу 1 и по каналу 2 коммутатору 2 (т.е. в сети с этого момента существует уже два фрейма). Предположим, что коммутатор 2 сначала получит фрейм по каналу 1. Он также проверит свою таблицу MAC адресов и обнаружит, что никакой информации о получателе фрейма у него нет. Так же, как и коммутатор 1, он ретранслирует этот фрейм во все свои интерфейсы (канал 2 и канал до ПК2). Коммутатор 1, получив по каналу 2 фрейм опять проверит, что он «не знает» информации о получателе фрейма и ретранслирует его в канал 1 и канал до ПК1. В результате в сети появится

фрейм, который будет непрерывно циркулировать между коммутатором 1 и коммутатором 2. Аналогичная ситуация сложится и с фреймом, который поступит от коммутатора 1 в коммутатор 2 по каналу 2. Будет уже два фрейма, бесконечно циркулирующих в противоположных направлениях. Бесконечная передача фреймов прекратится после того, как ПК2 отправит хотя бы один фрейм в ответ ПК1. В этом случае в коммутаторе 2 появится запись о том, что ПК2 подключен соответствующему интерфейсу и все приходящие фреймы будут переданы уже только в один канал. Циркуляция фреймов прекратится. Аналогичным образом будет происходить обработка фреймов в сети с количеством коммутаторов больше 2 и их петлевым соединением (см. рисунок 4).

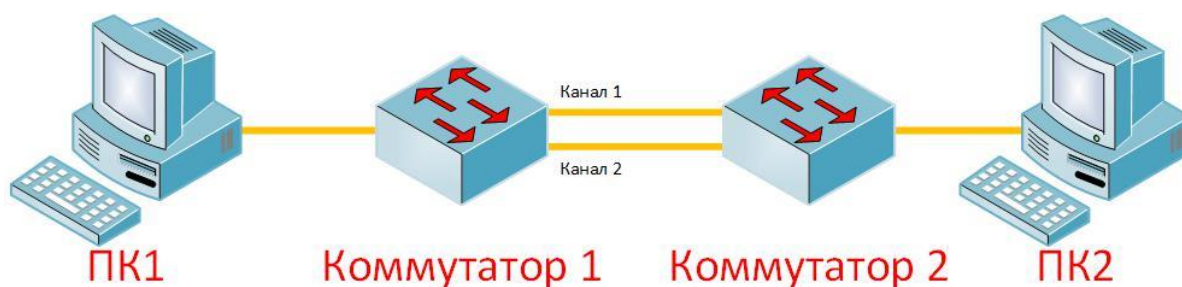


Рисунок 3 – Пример формирования петли в сети передачи данных между двумя коммутаторами

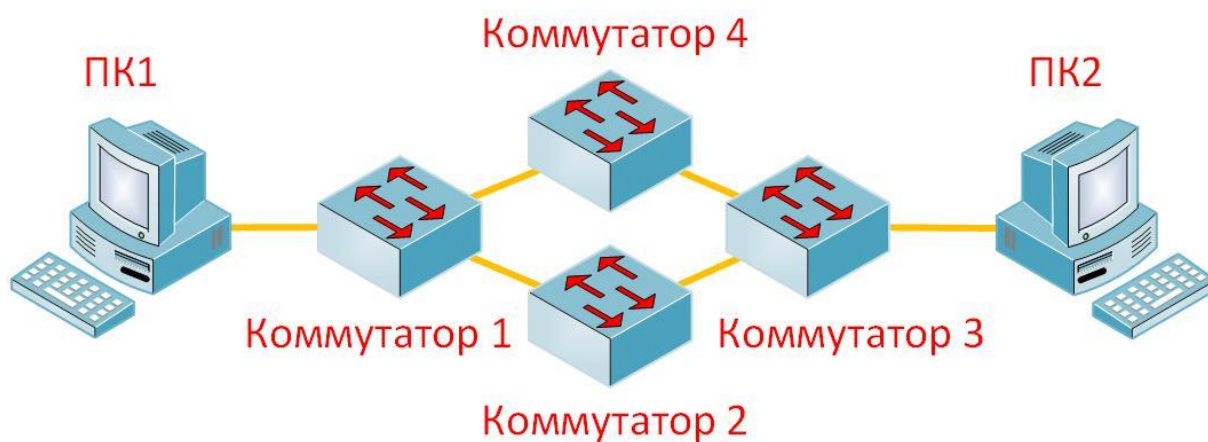


Рисунок 4 - Пример формирования петли в сети передачи данных более, чем двумя коммутаторами

А что произойдет в случае, если ПК1 отправит широковещательный фрейм? Его обработка будет производиться аналогичным образом, что и обработка фрейма, предназначенного ПК2. Отличие лишь будет в том, что в служебных таблицах коммутаторов никогда не появится запись о получателе такого фрейма, и он будет циркулировать по сети бесконечно.

Широковещательная передача данных активно используется всеми сетевыми устройствами (например, для разыменования IP адресов). В результате количество бесконечно циркулирующих фреймов в сети станет на столько

большим, что коммутаторы будут не способны обработать их. Такая ситуация называется «цифровой (широковещательный) шторм».

Алгоритм связующего дерева (Spanning-Tree Protocol)

В локальных компьютерных сетях, предъявляющих повышенные требования к надежности передачи данных, важной задачей является организация резервирования каналов. Для этих целей используется несколько независимых коммутаторов, которые, в свою очередь, могут соединяться в общем случае «каждый с каждым». А как же быть с появлением цифрового шторма в таких сетях?

Для защиты сети от появления в её структуре петель разработаны специальные алгоритмы, с помощью которых коммутаторы самостоятельно анализируют текущую конфигурацию сети и временно отключают резервные каналы. В случае выхода из строя какого-либо канала связи, коммутатор автоматически включает резервный канал и использует его для передачи данных.

Широко распространён протокол формирования связующего дерева (Spanning-Tree Protocol, STP), описанный в стандарте IEEE 802.1d. Автором этого протокола является Радья Перлман.

Согласно алгоритму STP в сети определённым образом один из коммутаторов выбирается корневым, а все остальные коммутаторы выстраивают относительно него дерево связей, временно отключая резервные каналы. Пока не будет построено такое дерево коммутаторы не будут передавать никакую информацию (кроме информации самого алгоритма STP) через свои интерфейсы. Это накладывает определённые ограничения на вхождение сети в рабочий режим, но гарантирует отсутствие в сети петель и её надежное функционирование.

Формат кадров BPDU (Bridge Protocol Data Units)

Коммутаторы сети с целью реализации протокола STP обмениваются информацией в виде специальных фреймов, которые называются BPDU (Bridge Protocol Data Units).

Фреймы STP имеет следующие поля (см. рисунок 5):

- Идентификатор используемого протокола (2 байта). Очевидно, что для нормальной работы протокола коммутаторы должны поддерживать одну и ту же его версию (о версиях STP будет сказано ниже).
- Версия протокола STP (1 байт).
- Тип BPDU (1 байт). Существует два типа: информирование о корневом коммутаторе и информирование о необходимости перестроения сети в результате изменения её конфигурации.
- Флаги (1 байт).

- Идентификатор корневого коммутатора (8 байт).
- Расстояние до корневого коммутатора (4 байта).
- Идентификатор коммутатора-отправителя фрейма (8 байт, о формате см. ниже).
- Идентификатор порта на коммутаторе-отправителе (2 байта, о формате см. ниже).
- Время жизни фрейма (2 байта). Измеряется в единицах по 0.5 сек, служит для выявления устаревших сообщений.
- Максимальное время жизни фрейма (2 байта).
- Интервал hello (2 байт), интервал через который корневой коммутатор посылает пакеты BPDU.
- Задержка смены состояний (2 байта). Минимальное время перехода интерфейсов коммутатора между состояниями.

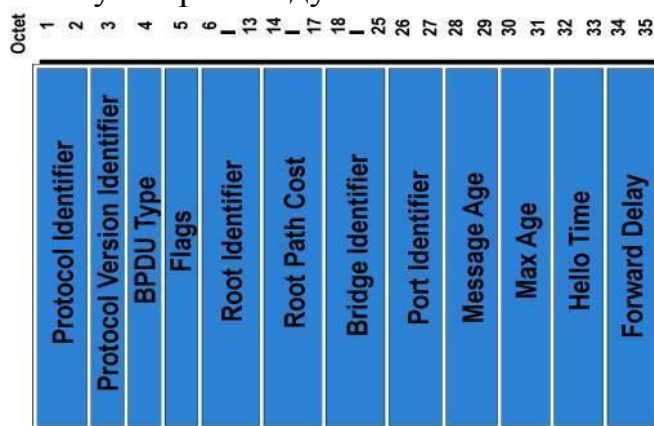


Рисунок 5 – Формат фрейма BPDU

Идентификация коммутаторов и их интерфейсов

В протоколе STP каждому коммутатору и каждому его интерфейсу задается приоритет – целое число. Приоритет коммутатора выбирается из диапазона от 0 до 65535. Приоритет порта задается от 0 до 255. Чем меньше выбранное значение, тем выше приоритет коммутатора или порта.

Идентификатором коммутатора является 64-х разрядное число, в младших 6-ти разрядах которого располагается MAC адрес, а в старших 2-х - его приоритет. Порт идентифицируется двухбайтовым числом, в котором в старшем байте приоритет порта, а в младшем его номер.

Роли, которые выполняют интерфейсы в результате работы STP

Цель работы STP – построить в сети структуру связей коммутаторов в виде дерева без циклов. Структура сети задается путем определения каждому интерфейсу одной из двух ролей (см. рисунок б): корневого (RP, от англ. Root port) или назначенного порта (DP, от англ. Distinguished port) или их блокирования. Корневой порт имеет самый короткий путь до корневого коммутатора. Назначенный порт – отвечает за передачу данных нижележащим (в

терминах дерева) коммутаторам. Очевидно, что корневой коммутатор имеет только назначенные порты. Остальные коммутаторы имеют как минимум один корневой порт и могут не иметь назначенных портов.

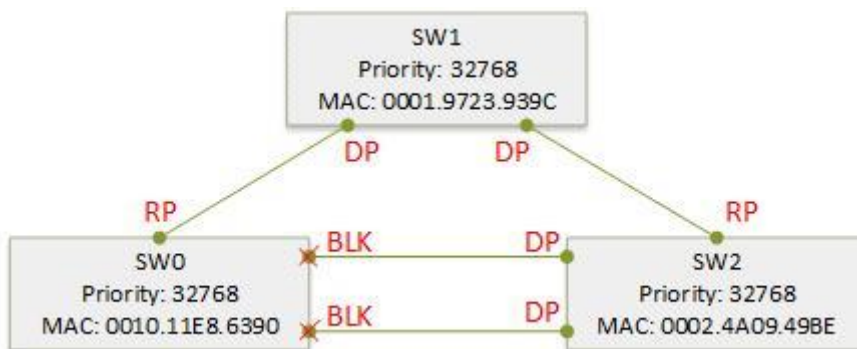


Рисунок 6 – Пример формирования структуры сети по алгоритму STP

Состояния интерфейсов коммутаторов, участвующих в STP

Все порты коммутатора, которые участвуют в реализации STP находятся в одном из 5 состояний (см. рисунок 7): блокирован (Blocking), слушает (Listening), изучает трафик (Learning), работает (Forwarding), отключен (Disabled).

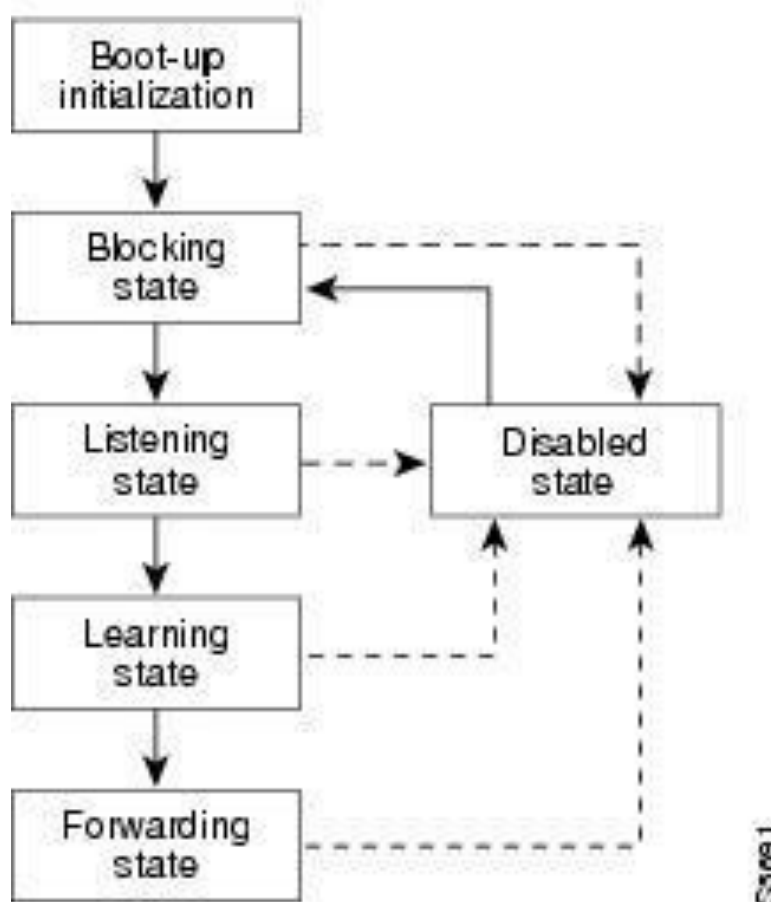


Рисунок 7 – Диаграмма переходов состояний портов в процессе работы STP.

Начальным для каждого порта коммутатора является состояние «блокирован». Передача фреймов через порт в этом режиме не производится. Принимаются и анализируются только фреймы BPDU. Из этого состояния порт может перейти в состояние «слушает» или «отключен».

В режиме «слушает» порт коммутатора принимает и передает служебные сообщения BPDU. Другие фреймы коммутатором не принимаются и не передаются. Таблица MAC-адресов коммутатора не изменяется. Из этого состояния порт может перейти в состояние «отключен» или «изучает».

В состоянии «изучает» порт продолжает принимать и передавать служебные сообщения BPDU. В этом состоянии порт принимает другие фреймы и использует информацию из их заголовков для формирования таблицы MAC-адресов (готовясь, тем самым, к выполнению обязанностей по коммутированию кадров). Из этого состояния порт может перейти в состояние «заблокирован» или «работает».

В режиме «работает» порт выполняет функции по получению и ретрансляции всех кадров в обычном режиме.

По умолчанию, порт находится в состоянии «блокирован» в течение промежутка времени, определённого параметром «max age». В состояниях «слушает» и «изучает» - в течение промежутка времени, заданного параметром forward delay. В результате, с момента начала работы алгоритма до момента начала передачи данных через порт должно пройти минимум max_age + forward delay сек. (по умолчанию этот минимум равен 35 с). Именно из-за этого свойства порты коммутатора, которые подключены к оконечному оборудованию исключают из работы STP (подробнее об этом ниже).

Определение расстояния между коммутаторами сети

Формирование структуры сети связей коммутаторов производится исходя из расстояний до корневого коммутатора. Под «расстоянием» понимается сумма весов каналов, через которые необходимо передать данные, чтобы достичь корневого коммутатора. Вес канала определяется скоростью передачи по нему: 0 – 1 Мбит/с = 256, 4Мбит/с = 200, 10 Мбит/с = 100, 16 Мбит/сек = 62, 100 Мбит/с = 19, 1 Гбит/с = 4, 10 Гбит/с = 2.

Алгоритм построения связующего дерева

Алгоритм построения связующего дерева в сети предполагает выполнение трех фаз:

- выбор корневого коммутатора;
- назначение корневого порта;
- определение назначенных и блокирование оставшихся портов.

1 Выбор корневого коммутатора

Процедура выборов корневого коммутатора продолжается в течение $max\ age$ секунд. В самом начале работы каждый коммутатор считает себя корневым. Корневой коммутатор сети через $hello$ секунд генерирует сообщение BPDU, в котором указывает свой идентификатор, значения своих параметров и отправляет его через все свои порты.

Получив BPDU коммутатор сравнивает значение указанного в нем идентификатора корневого коммутатора с имеющимся у него значением. Корневым назначается коммутатор, имеющий меньшее значение идентификатора.

Потеряв статус корневого коммутатора генерирует BPDU сообщения только после получения соответствующих сообщений от корневого коммутатора. Сообщения BPDU именно генерируются, а не ретранслируются коммутаторами, поскольку сообщения формируются для каждого порта в отдельности и в каждом из них рассчитывается значение поля «расстояние до корневого коммутатора».

В результате, с течением времени, в сети остается только один коммутатор с наименьшим значением идентификатора, который генерирует сообщения BPDU через интервал в $hello$ секунд.

2 Назначение корневого порта

Получив сообщение BPDU коммутатор запоминает его как лучшее для порта, а затем передает его на обработку. Лучшим считается сообщение, пришедшее от текущего корневого коммутатора и имеющее меньшее значение расстояния до него.

Корневой порт выбирается по результатам анализа сохранённых значений BPDU. Рассматриваются порты, в которых были получены сообщения от корневого коммутатора и имеющие наименьшее значение расстояния до него. Корневым признается порт, через который получено сообщение BPDU от соседнего коммутатора, имеющего наименьшее значение идентификатора и использовавший для передачи порт с наименьшим значением приоритета.

3 Определение назначенных и блокирование всех остальных портов

После определения корневого порта необходимо определить какие порты будут использоваться для передачи данных (т.е. будут назначенными), а какие из них необходимо перевести в состояние «блокирован».

Порты коммутатора, через которые не было получено сообщений от корневого коммутатора, сразу определяются назначенными (поскольку явно в этом направлении нет петель). Далее рассматриваются только порты, через которые такие сообщения были получены, но они не назначены корневыми.

Рассматривается каждый порт в отдельности. Порт получает роль «назначенного», если расстояние от текущего коммутатора до корневого меньше, чем расстояние от соседнего коммутатора до корневого. Если расстояния одинаковые, то назначенным остается порт коммутатора с меньшим идентификатором. В противном случае порт получает состояние «блокирован».

Перестроение конфигурации сети.

В случае если какой-либо коммутатор обнаруживает ситуацию, которая приведет к необходимости перестроить конфигурацию сети, то он формирует специальное сообщение BPDU – TCN (англ. Topology Change Notification) и отправляет его через свой корневой порт. Получив такое сообщение все промежуточные коммутаторы изменяют таймеры своих MAC таблиц и переправляют полученное сообщение через свои корневые порты. Корневой коммутатор получив сообщение TCN выставляет флаг TCA (англ. Topology Change Acknowledgement) в своих сообщениях BPDU, которые он генерирует каждые hello секунд. Получив такое сообщение каждый коммутатор начинает процедуру формирования связующего дерева.

Изменение таймеров в таблицах MAC необходимо для того, чтобы перестроение таблиц адресов произошло как можно быстрее (конфигурация сети ведь изменилась).

Перестроение конфигурации сети может потребоваться в следующих ситуациях:

- в коммутаторе один или несколько интерфейсов изменили свое состояние;
- коммутатором получено сообщение BPDU со значением корневого коммутатора меньшим, чем значение текущего корневого коммутатора;
- коммутатор не получал сообщений BPDU от корневого коммутатора в течение заданного промежутка времени.

Первая ситуация может возникнуть если к коммутатору подключили новое или отключили ранее подключенное сетевое устройство. В этой ситуации коммутатор переводит все свои интерфейсы в состояние «блокирован», отправляет TCN и ждет начала процедуры перестроения конфигурации.

Вторая ситуация возможно если в процессе функционирования какого-либо коммутатора администратор изменил его приоритет.

Пример формирования конфигурации сети по алгоритму связующего дерева

Рассмотрим сеть, представленную на рисунке 8. В ней присутствует 6 коммутаторов, обозначенных прямоугольниками. Внутри каждого прямо-

угольника указано имя коммутатора, его приоритет, MAC-адрес и номер каждого интерфейса. Красным цветом указаны роли портов. Цифрами – расстояние до корневого коммутатора. Предполагается, что все каналы связи имеют пропускную способность в 100 Мбит/с. Рассмотрим, как получилась такая конфигурация сети.

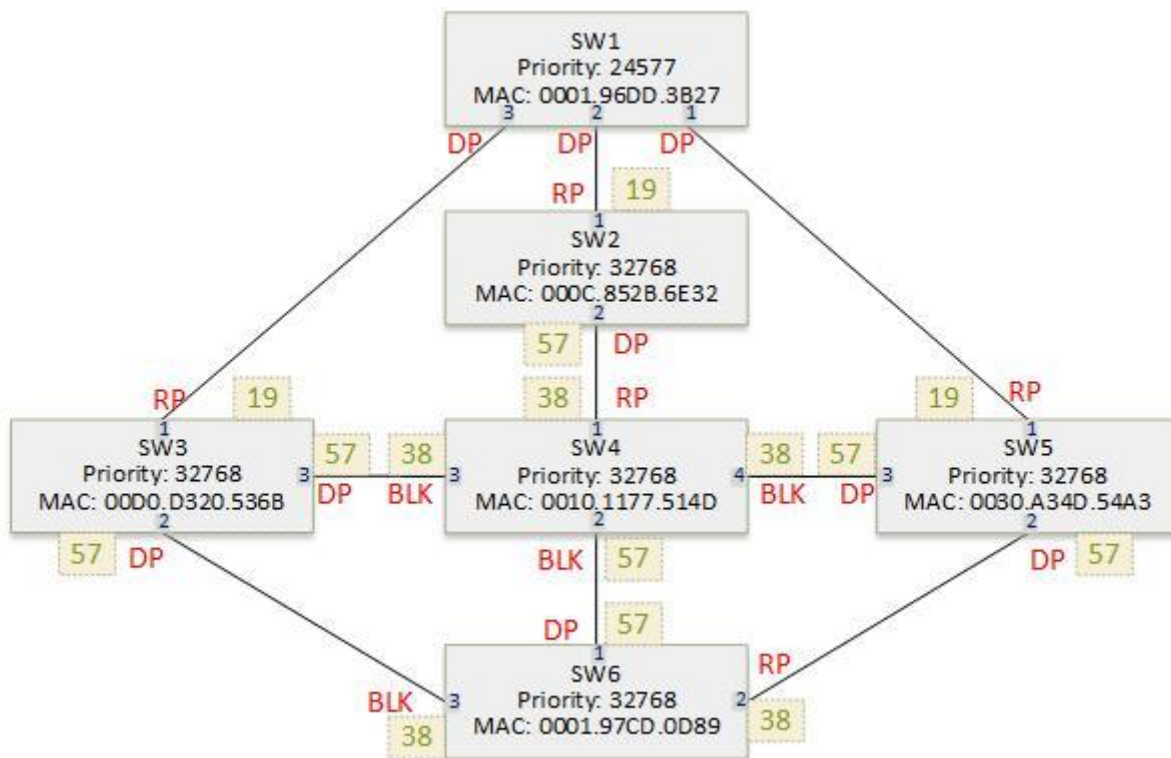


Рисунок 8 – Пример формирования конфигурации сети по алгоритму связующего дерева

Корневым коммутатором сети избран SW1, т.к. у него самое маленькое значение приоритета. Всем его портам определена роль «назначенный». Рассылая BPDV через порты 1-3, корневой коммутатор указывает свой идентификатор и расстояние, равное 0. Получив сообщение от корневого коммутатора через интерфейсы с номер 1, коммутаторы SW2, SW3, SW5 записывают, что расстояние до него будет равно 19 (т.е. будет использован только один канал в 100 Мбит/с.). Далее эти коммутаторы генерируют BPDV со значением расстояния равным 19 и рассылают его через все свои интерфейсы.

Коммутатор SW4, получив сообщение от SW2 определяет для интерфейса 1 расстояние до корневого коммутатора равным 38 (19 пришло от SW2 + 19 канал между SW4 и SW2). И рассылает BPDV со расстоянием в 38 через порты 2-4. Коммутаторы SW3 и SW5, получив от SW4 BPDV запоминают, что через канал 3 расстояние до корневого коммутатора будет равным 57. Далее коммутаторы действуют аналогичным образом. Расчет расстояний представлен на рисунке.

На этапе определения корневых портов, коммутаторы SW3, SW2, SW5 выбирают интерфейсы с номером 1, как имеющие минимальные значения расстояний до корневого коммутатора.

У коммутатора SW4 три порта имеют одинаковое расстояние – 1, 3 и 4. Рассматривая соседей SW2, SW3, SW5 можно увидеть, что коммутатор SW2 имеет идентификатор с меньшим значением. Поэтому корневым портом SW4 будет выбран порт номер 1. У коммутатора SW6 аналогичная ситуация: имеется два порта 3 и 2 с одинаково минимальным расстоянием до корневого коммутатора. Рассматривая соседей, определяем, что корневым будет порт с номером 2.

Следующим шагом необходимо определить какие порты будут назначенными, а какие необходимо заблокировать. У коммутаторов SW2 и SW1 выбор назначенных портов очевиден. У коммутатора SW6 осталось два порта – 1 и 3, через которые были получены BPDU от корневого коммутатора. SW4 и SW6 имеют одинаковые расстояния до корневого коммутатора, поэтому назначенным остается порт у коммутатора с меньшим значением идентификатора (SW4). У второго коммутатора порт блокируется. Рассматривая порт 3 коммутатора SW4 видно, что у SW3 расстояние до корневого коммутатора меньше, следовательно, порт 3 блокируется. Аналогично рассматриваются все остальные порты у все остальных коммутаторов.

Другие версии протокола связующего дерева

Классический протокол формирования связующего дерева требует определенного времени для построения конфигурации сети и восстановления её после сбоя. На практике это время оказалось значительным и были предприняты попытки сократить его. Кроме того, с появлением технологии VLAN потребовалось учитывать её при формировании конфигурации сети. Чтобы решить эти проблемы были предложены новые версии STP: Rapid STP, PVSTP, RPVST, PVST+, RPVST+. Кроме того, были предложены технологии, расширяющие протокол STP и позволяющие повысить защиту этого протокола от атак, а также обеспечить поддержку портов, к которым чаще всего подключены оконечные устройства. Рассмотрим некоторые из предложенных версий протоколов и расширений протокола STP.

1 Протокол Per-VLAN STP (PVST)

Компанией CISCO предложена версия протокола STP, работающая в сети, использующей VLAN. Этот протокол позволяет настраивая конфигурацию сети оперировать не физическими интерфейсами, а логическими портами, соответствующими виртуальным сетям. В рамках данного протокола в каждом VLAN работает отдельный экземпляр STP.

2 Расширение System-ID

Чтобы обеспечить построение в каждой VLAN своей конфигурации сети используя различные настройки коммутаторов (разные идентификаторы) была предложена технология, «скрывающая» номер VLAN внутри идентификатора коммутаторов. В результате приоритет коммутатора формируется из двух частей: самого приоритета (4 старших бита) и номера VLAN (12 младших бит).

3 Rapid STP

Используя протокол STP порт переходит в рабочий режим минимум через 30 секунд после начала построения дерева связей. Перестроение конфигурации сети также требует значительного времени. Чтобы исключить эти недостатки была разработана версия протокола, названная «быстрой» - Rapid STP. В этом протоколе сокращено количество состояний, через которые проходит порт, выстраивая конфигурацию сети до трех – «изучает», «передает» и «отключен». Добавлены статусы отключенных портов: «альтернативный» и «запасной». Компания CISCO для своего оборудования предложила протокол Rapid PVSTP.

4 Режим работы порта PortFast.

В процессе работы протокола STP порт переходит в рабочее состояние не раньше, чем через 29 секунд. Чтобы сократить это время порт можно исключить из работы STP. Сделать это можно только в том случае, если есть уверенность, что к этому порту не будет подключен коммутатор, который может организовать петлю в сети. Чтобы обеспечить быстрый переход в режим передачи кадров и иметь возможность отслеживать появление петель предложена технология PortFast, которая гарантирует, что порт не будет проходить состояния «слушает» и «изучает», а сразу перейдет в состояние «работает».

Увеличение пропускной способности каналов передачи данных (протокол LACP)

Любой канал (как и его соответствующая физическая среда) имеет ограниченную пропускную способность. Повысить скорость передачи данных по каналу возможно двумя способами: увеличивая пропускную способность канала (за счет расширения частоты передачи, применения различных кодирований и т.п.) или использовать для передачи данных сразу несколько каналов. Во втором случае пропускная способность канала формируется из суммарной пропускной способности объединенных каналов. Объединение каналов для передачи называется *агрегирование*.

Эта же технология используется для обеспечения отказоустойчивого функционирования канала передачи данных – некоторые каналы назначаются резервными и включаются только при необходимости.

Создать агрегированный канал можно двумя способами: статически и динамически. В первом случае каналы объединяются путем ручного конфигурирования оборудования на обоих концах канала. Второй способ предполагает механизм, позволяющий оборудованию «договориться» об агрегировании каналов. Если переговоры провести не удастся, то каналы продолжают функционировать независимо в обычном режиме.

Для того, чтобы коммутаторы восприняли несколько каналов как единое целое используется специальный протокол – LACP (Link Aggregation Control Protocol). Согласно этому протоколу интерфейс, включенный в агрегированную группу, отправляет в канал специальный кадр LACPDU. Получив такой кадр устройство, поддерживающее LACP, формирует ответ и тем самым согласовывает создание агрегированного канала. Кадр LACPDU содержит следующие поля: номер интерфейса, номер системы, параметры порта, текущий статус.

Агрегированные каналы распределяют передаваемые кадры между физическими каналами исходя из заданных правил: по MAC-адресу отправителя или MAC-адресу получателя, по IP адресу отправителя или получателя, по номеру порта отправителя и получателя. Исходя из значений указанных параметров выбирается один из интерфейсов коммутатора и по нему передаются данные.

Конфигурирование протокола STP в коммутаторах CISCO

В коммутаторах CISCO по умолчанию включен протокол STP (PVSTP) и используется он на всех интерфейсах. Посмотреть текущее состояние конфигурации сети можно в привилегированном режиме с использованием команды `show spanning-tree` (см. рисунок 10). Дополнительно команде можно задать параметры:

- `Active` – вывести информацию только о созданных конфигурациях сетей (VLAN). Этот параметр используется «по умолчанию».
- `Detail` – Вывести детальную информацию о состоянии протокола STP.
- `Inconsistentports` – вывести информацию о портах, в которых возникла какая-либо ошибка в процессе функционирования STP.
- `Interface` – вывести информацию о работе STP на указанном интерфейсе. Здесь же можно узнать работает ли порт в режиме `portfast`.
- `Summary` – выдать краткую информацию о работе STP

- Vlan – выдать информацию о конфигурации, построенной для указанной vlan.

```
Switch#show spanning-tree
VLAN0001
Spanning tree enabled protocol ieee
Root ID    Priority    24577
           Address    0001.96DD.3B27
           This bridge is the root
           Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID  Priority    24577 (priority 24576 sys-id-ext 1)
           Address    0001.96DD.3B27
           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
```

Рисунок 9 – Пример вывода информации о работе SPT

На рисунке 9 показана информация о работе STP. Информация выдается трех типов: информация о корневом коммутаторе сети (Root ID), информация о текущем коммутаторе и информация о портах. О коммутаторах выдается информация: приоритет, адрес, значения таймеров (параметров BPDUs). О каждом интерфейсе можно узнать роль, режим, стоимость канала, к которому подключен интерфейс, приоритет порта и тип соединения.

На рисунке 10 показан пример вывода детальной информации о работе STP. Информация показывается для каждой VLAN в отдельности. Сначала приводится информация сформированной конфигурации сети и настройках коммутатора: идентификатор коммутатора, значения таймеров, приоритет корневого коммутатора, значения флагов перестроения конфигурации и статистика работы STP. Затем выводится информация о каждом интерфейсе и наилучшей полученной BPDUs: роль, которую выполняет порт; стоимость канала, к которому подключен порт; приоритет порта и его идентификатор, информация о корневом коммутаторе и коммутаторе, от которого получена BPDUs⁹, значение таймеров и статистика работы STP.

```

Switch#show spanning-tree detail

VLAN0001 is executing the ieee compatible Spanning Tree Protocol
  Bridge Identifier has priority of 24576, sysid 1, 0001.96DD.3B27
  Configured hello time 2, max age 20, forward delay 15
  Current root has priority 24577
  Topology change flag not set, detected flag not set
  Number of topology changes 0 last change occurred 00:00:00 ago
    from FastEthernet0/1
  Times: hold 1, topology change 35, notification 2
    hello 2, max age 20, forward delay 15
  Timers: hello 0, topology change 0, notification 0, aging 300

Port 1 (FastEthernet0/1) of VLAN0001 is designated forwarding
  Port path cost 19, Port priority 128, Port Identifier 128.1
  Designated bridge has priority 24577, address 0001.96DD.3B27
  Designated port id is 128.1, designated path cost 19
  Timers: message age 16, forward delay 0, hold 0
  Number of transitions to forwarding state: 1
  Link type is point-to-point by default

```

Рисунок 10 – Пример вывода детальной информации о работе SPT

Следует отметить, что в системе моделирования Cisco packet tracer версии 6.0.1 в выводе информации по каждому интерфейсу есть путаница: вместо информации о корневом коммутаторе выводится информация о корневом интерфейсе текущего коммутатора, а вместо информации о соседнем коммутаторе выводится информация о корневом коммутаторе.

Получить информацию об используемом протоколе STP и состояние различных расширений можно с помощью команды `show spanning-tree summary` (см. рисунок 11). На рисунке видно, что текущий коммутатор выполняет протокол версии PVST и использует расширение `extended system-id`. По умолчанию все порты работают в обычном режиме (`portfast default – disabled`).

```

Switch#show spanning-tree summary
Switch is in pvst mode
Root bridge for: default
Extended system ID      is enabled
Portfast Default        is disabled
PortFast BPDU Guard Default is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default      is disabled
EtherChannel misconfig guard is disabled
UplinkFast              is disabled
BackboneFast            is disabled
Configured Pathcost method used is short

Name                    Blocking Listening Learning Forwarding STP Active
-----
VLAN0001                0          0          0          3          3
-----
1 vlans                 0          0          0          3          3

```

Рисунок 11 – Пример вывода статистики по STP

Настройка протокола SPT производится в режиме конфигурирования командой `spanning-tree` (см. рисунок 12). Этой командой можно задать тип используемого протокола (`mode`), включить/отключить режим `portfast` «по

умолчанию», включить/отключить использования STP в каждом VLAN и задать приоритет коммутатора (vlan [vlan] priority) или использовать автоматическое средство назначения текущего коммутатора корневым.

```
Switch#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#spanning-tree mode pvst
Switch(config)#no spanning-tree portfast default
Switch(config)#spanning-tree vlan 1 priority 4096
Switch(config)#spanning-tree vlan 1
```

Рисунок 12 – Пример конфигурирования протокола STP

В режиме конфигурирования интерфейса можно задать параметры порта командой `spanning-tree` (см. рисунок 13). Здесь можно: запретить/разрешить прием фреймов BPDU через порт (`spanning-tree bpduguard enable|disable`); указать, что если этот порт не будет выбран корневым, то он должен быть заблокирован (`spanning-tree guard root`); определить тип линии связи (допускается ли наличие каких-либо промежуточных устройств (например, концентраторов) или эта прямая связь с другим коммутатором); запретить/разрешить использование режима `portfast`; задать приоритет порта для каждого VLAN.

```
Switch(config)#interface fastEthernet 0/1
Switch(config-if)#spanning-tree bpduguard disable
Switch(config-if)#no spanning-tree guard
Switch(config-if)#spanning-tree link-type shared
Switch(config-if)#spanning-tree vlan 1 port-priority 16
```

Рисунок 13 – Пример конфигурирования порта для работы в STP

Конфигурирование протокола LACP в коммутаторах CISCO

В коммутаторах CISCO допускается создание от 1 до нескольких групп интерфейсов, объединяемых в логические группы. Определить алгоритм балансировки трафика в группе можно в привилегированном с помощью команды `port-channel` (см. рисунок 14). Для создания группы в неё необходимо добавить хотя бы один интерфейс. Сделать это можно в режиме конфигурирования интерфейса командой `channel-group`. В качестве параметров команды указывается номер группы и режим работы интерфейса: активный (т.е. коммутатор будет инициировать создание группы отправляя LACPDU в канал), пассивный (интерфейс будет включен в группу только после получения от соседнего коммутатора LACPDU и согласования работы агрегированного канала), а также определить использование технологии PAgP (проприетарной технологии CISCO, эта технология не рассматривается в рамках лабораторной работы). Используемый протокол агрегирования задается командой `channel-group protocol`. В результате в системе появится новый интерфейс `port-channel`.

```

Switch(config)#port-channel load-balance ?
dst-ip      Dst IP Addr
dst-mac     Dst Mac Addr
src-dst-ip  Src XOR Dst IP Addr
src-dst-mac Src XOR Dst Mac Addr
src-ip      Src IP Addr
src-mac     Src Mac Addr
Switch(config)#port-channel load-balance dst-mac
Switch(config)#interface fastEthernet 0/17
Switch(config-if)#channel-protocol lacp
Switch(config-if)#channel-group 1 mode passive

```

Рисунок 14 – Пример конфигурирования агрегированного канала на коммутаторе CISCO

Ход выполнения работы

Требуется соединить физическую сеть в соответствии со схемой сети или построить соответствующий проект в Cisco Packet Tracer (рисунок 15). Сразу после схемы сети в таблице указана схема адресация (таблица 1).



Рисунок 15 – Топология сети

Таблица 1 – Таблица адресации

Устройство	Интерфейс	IP адрес	Маска подсети
SW1	VLAN1	192.168.0.N	255.255.255.0
SW2	VLAN1	192.168.0.N+1	255.255.255.0
SW3	VLAN1	192.168.0.N+2	255.255.255.0
SW4	VLAN1	192.168.0.N+3	255.255.255.0

N – номер по журналу

Последовательность выполняемых действий.

1. Выполнить конфигурацию сетевых параметров для Sw1, Sw2, Sw3, Sw4.

```
Switch(config)#interface vlan 90
Switch(config-if)#
%LINK-5-CHANGED: Interface Vlan90, changed state to up

Switch(config-if)#ip add
Switch(config-if)#ip address 192.168.90.1 255.255.255.0
```

2. Проверить правильность настройки утилитой Ping

```
Switch>ping 192.168.0.4

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.0.4, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/1 ms
```

3. Сохранить конфигурацию

```
Switch#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

4. Анализирование конфигурации STP

```
Switch#show spanning-tree
VLAN0001
Spanning tree enabled protocol ieee
Root ID    Priority    32769
           Address    0001.4371.7577
           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    0001.4371.7577
           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

Рисунок 16 – Вывод команды show spanning-tree на SW2

Как видно исходя из рисунка 16, после ввода команды show spanning tree, становится понятно, что S2 является root bridge на основании меньшего VID. Его интерфейсы Fa0/1, Fa0/2, Fa0/3 являются назначенными. Далее согласно алгоритма STP получается 3 заблокированных порта в сети и тем самым отсутствие сетевых петель.

```

Spanning tree enabled protocol ieee
Root ID    Priority    32769
          Address    0001.4371.7577
          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    0060.5C9D.A270
          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
Fa0/3          Desg FWD 19        128.3    P2p
Fa0/2          Root FWD 19        128.2    P2p
Fa0/4          Desg FWD 19        128.4    P2p

```

Рисунок 17 – Вывод команды show spanning-tree на SW1

Исходя из рисунка 17, видно, что порт Fa0/1 – является альтернативным и временно заблокирован для предотвращения образования петель. Порт Fa0/3 и Fa0/4 является назначенным, порт Fa0/2 является корневым.

```

Switch#show spanning-tree
VLAN0001
Spanning tree enabled protocol ieee
Root ID    Priority    32769
          Address    0001.4371.7577
          Cost      19
          Port      1(FastEthernet0/1)
          Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID  Priority    32769 (priority 32768 sys-id-ext 1)
          Address    00E0.F9BA.6635
          Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
          Aging Time 20

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

```

Рисунок 18– Вывод команды show spanning-tree на SW3

На рисунке 18 видно, что порт Fa0/1 является корневым, а порты Fa0/2 и Fa0/3 являются альтернативными и временно заблокированы.

```

Switch#show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
            Address    0001.4371.7577
            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    0003.E46A.E06E
            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        Root FWD 19        128.3    P2p

```

Рисунок 19 – Вывод команды show spanning-tree на SW4

На рисунке 19 видно что порты Fa0/1 и Fa0/2 являются назначенными. порт Fa0/3 является корневым.

5. Изменить приоритет коммутатора, находящегося по диагонали от корневого командой: spanning-tree vlan 1 priority 4096.

Подождать пока сеть перенастроится. Посмотреть настройки коммутаторов командой: show spanning-tree

Контрольные вопросы

1. Что такое «цифровой шторм»? При каких условиях он возникает в компьютерных сетях?
2. Что такое «стек коммутаторов»?
3. В чем основная идея алгоритма «Spanning-Tree»?
4. Алгоритм определения корневых и назначенных портов коммутаторов по STA?
5. Объясните, как связаны между собой роли и режимы портов при реализации STP?
6. Что такое идентификатор коммутатора? Почему по умолчанию приоритет коммутатора в CPT можно задать только как множитель на 4096?
7. Зачем используется режим PortFast?
8. Что такое агрегирование каналов? Зачем оно используется?
9. Что можно использовать для обеспечения резервирования каналов: алгоритм SP или агрегирование каналов?
10. Чем отличается статический и динамический режим агрегирования каналов?
11. Какие алгоритмы применяются для балансировки трафика в агрегированных каналах?