

Министерство образования и науки РФ
Московский государственный технический университет
имени Н.Э. Баумана
Факультет: «Информатика и системы управления»
Кафедра «Компьютерные системы и сети» (ИУ6)

Лабораторная работа № 3
Технологии канального уровня

Учебное пособие

Москва 2013

СОДЕРЖАНИЕ

Введение	3
Теоретическая часть	4
Описание стенда	12
Практическая часть	16
Требования к отчету	22
Контрольные вопросы	23
Литература	24

Введение

Целью работы является - изучить и освоить возможности современных управляемых коммутаторов, позволяющие изменять, конфигурировать виртуальные локальные сети.

Теоретическая часть

Виртуальные сети (Virtual LAN)

Виртуальная ЛВС (VLAN, Virtual LAN) – логическая группа компьютеров в пределах одной реальной ЛВС, за пределы которой не выходит любой тип трафика (широковещательный [broadcast], многоадресный [multicast] и одноадресный [unicast]). За счет использования VLAN администратор сети может организовать пользователей в логические группы независимо от физического расположения рабочих станций этих пользователей.

Основные цели введения виртуальных сетей в коммутируемую среду:

- повышение полезной пропускной способности сети за счет локализации широко-вещательного (broadcast) трафика в пределах виртуальной сети;
- повышения уровня безопасности сети за счет локализации одноадресного (unicast) трафика в пределах виртуальной сети;
- формирование виртуальных рабочих групп из некомпактно (в плане подключения) расположенных узлов;
- улучшение соотношения цены/производительности по сравнению с применением маршрутизаторов.

Классический пример, отражающий суть виртуальных сетей, приведен на рисунке 4. К одному коммутатору гипотетической фирмы подключены как машины бухгалтеров, так и машины инженеров. При этом совершенно нет никакой необходимости во взаимодействии машин данных двух групп сотрудников. Поэтому машины бухгалтеров выделяются в одну виртуальную сеть, а машины инженеров в другую. При этом весь трафик (широковещательный, многоадресный и одноадресный) будет ограничен пределами своей виртуальной сети. Более того, повысится безопасность сети. Например, если на машине бухгалтера появится вирус «червь», то максимум он сможет заразить только машины бухгал-
теров.

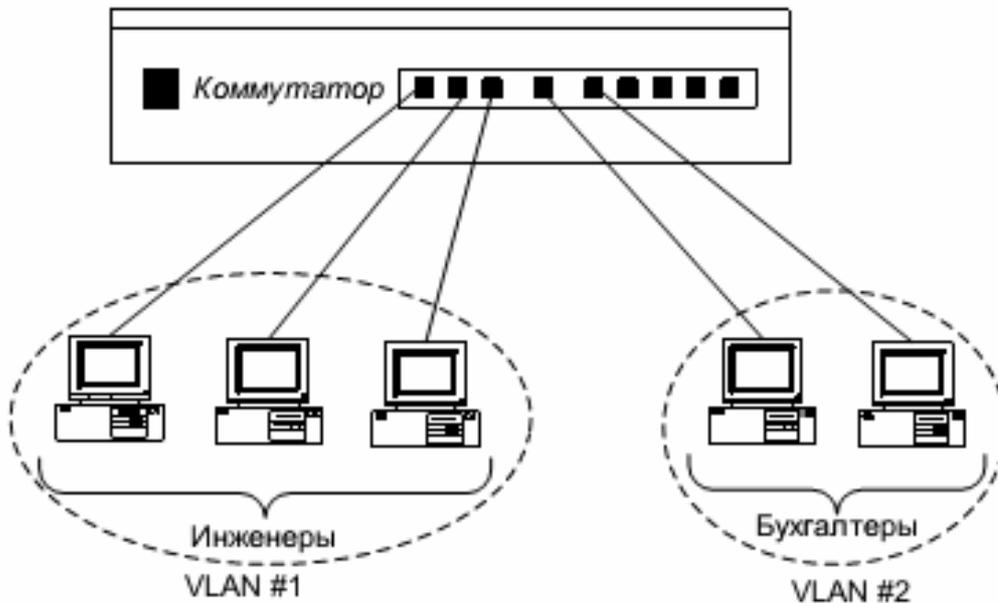


Рисунок 4. Пример использования технологии виртуальных сетей.

Сегодня существует достаточно много вариантов реализации VLAN.

Простые варианты

VLAN представляют собой набор портов коммутатора, более сложные реализации позволяют создавать группы на основе других критериев. В общем случае возможности организации VLAN тесно связаны с возможностями коммутаторов.

5.1. Сети на базе MAC-адресов (MAC-based VLANs)

Данный тип виртуальных сетей группирует устройства на основе их MAC-адресов. Для получения доступа в виртуальную сеть, устройство должно иметь MAC-адрес, который содержится в списке адресов данной виртуальной сети. Помимо прочего, отличительной особенностью данного типа виртуальных сетей является то, что они ограничивают только широковещательный трафик. Отсюда вытекает их название – широковещательные домены на базе MAC-адресов. Теоретически один MAC-адрес может являться членом нескольких широковещательных доменов, на практике данная возможность определяется функциональностью конкретной модели коммутатора.

Настройка виртуальной сети на основе MAC-адресов может отнять много времени. Представьте себе, что вам потребуется связать с VLAN адреса 1000 устройств. Кроме того, MAC-адреса "защиты" в оборудование и может потребоваться много

времени на выяснение адресов устройств в большой, территориально распределенной сети. Более того, существуют проблемы безопасности при работе с сетью на базе MAC-адресов. Злоумышленник может узнать MAC-адрес компьютера, входящего в ту или иную VLAN, назначить его своей сетевой карте (как минимум, это можно сделать стандартными средствами MS Windows XP) и успешно подключиться к сети.

С другой стороны, широковещательные домены на базе MAC-адресов позволяют физически перемещать станцию, позволяя, тем не менее, оставаться ей в одном и том же широковещательном домене без каких-либо изменений в настройках конфигурации. Это удобно в сетях, где станции часто перемещают, например, где люди, использующие ноутбуки, постоянно подключаются в разных частях сети.

5.2. Сети на базе портов (Port-based VLANs)

Устройства связываются в виртуальные сети на основе портов коммутатора, к которым они физически подключены. То есть каждый порт коммутатора включается в одну или более виртуальных сетей. К достоинствам данного типа виртуальных сетей можно отнести высокий уровень безопасности и простоту в настройке. К недостаткам можно отнести статичность данного типа виртуальных сетей. То есть при подключении компьютера к другому порту коммутатора необходимо каждый раз изменять настройки VLAN.

5.3. Сеть на базе маркированных кадров (IEEE 802.1Q VLANs)

В отличие от двух предыдущих типов виртуальных сетей VLAN на основе маркированных кадров могут быть построены на двух и более коммутаторах.

IEEE 802.1Q – открытый стандарт, который описывает процедуру тегирования трафика для передачи информации о принадлежности к VLAN. Кроме процедуры тегирования для передачи трафика разных VLAN в стандарте 802.1Q описаны:

- протокол GVRP;
- протокол MSTP;
- и др.

Однако чаще всего 802.1Q упоминается именно как стандарт, относящийся

к VLAN и процедуре тегирования. Коммутатор, на котором настроены виртуальные сети IEEE 802.1Q, помещает внутрь кадра тег, который передает информацию о принадлежности трафика к VLAN.

TPID	Priority	CFI	VLAN ID
------	----------	-----	---------

Размер тега составляет 4 байта. Он состоит из следующих полей:

Tag Protocol Identifier (TPID) – идентификатор протокола тегирования. Размер поля

– 16 бит. Указывает, какой протокол используется для тегирования.

Для 802.1q ис-

пользуется значение 0x8100.

Priority – приоритет. Размер поля – 3 бита. Используется

стандартом IEEE 802.1p

для задания приоритета передаваемого трафика.

Canonical Format Indicator (CFI) – индикатор канонического формата.

Размер поля –

1 бит. Указывает на формат MAC-адреса. 1 – канонический (кадр

Ethernet), 0 – не

канонический (кадр Token Ring, FDDI).

VLAN Identifier (VID) – идентификатор VLAN. Размер поля – 12 бит.

Указывает, ка-

кой виртуальной сети принадлежит кадр. Диапазон возможных

значений VID от 0

до 4094.

Тэг вставляется после поля «Адрес отправителя». Так как кадр изменился, то пересчитывается контрольная сумма (рисунок 5).

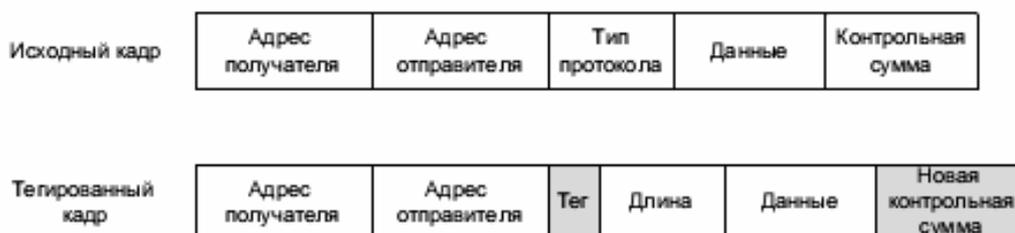


Рисунок 5.

Механизмы добавления тэга в заголовок кадра Ethernet

Тэг в заголовок кадра может быть добавлен:

явно, если сетевые карты поддерживают стандарт IEEE 802.1Q, и на этих картах

включены соответствующие опции, то исходящие кадры Ethernet от этих карт бу-

дут содержать тэги;

неявно, если сетевые адаптеры, подключенные к этой сети, не

поддерживают стандарт IEEE 802.1Q, то добавление маркеров выполняется на коммутаторе на основе группировки по портам.

Изменение формата кадра Ethernet приводит к тому, что сетевые устройства, не поддерживающие стандарт IEEE 802.1Q (такие устройства называют *ag- unaware*), не могут работать с кадрами, в которые вставлены метки. Поэтому для обеспечения совместимости с устройствами, поддерживающими стандарт IEEE 802.1Q (*ag-aware* устройства), коммутаторы стандарта IEEE 802.1Q должны поддерживать как традиционные Ethernet-кадры, то есть кадры без меток (*untagged*), так и кадры с метками (*tagged*).

Входящий и исходящий трафики, в зависимости от типа источника и получателя, могут быть образованы и кадрами типа *tagged*, и кадрами типа *untagged* – только в этом случае можно достигнуть совместимости с внешними по отношению к коммутатору устройствами. Трафик же внутри коммутатора всегда образуется пакетами типа *tagged*. Поэтому для поддержки различных типов трафиков и для того, чтобы внутренний трафик коммутатора образовывался из тегированных пакетов, на принимаемом и передающем портах коммутатора кадры должны преобразовываться в соответствии с predetermined правилами.

Конфигурирование виртуальной сети IEEE 802.1Q

Коммутаторы для ВЛС требуют предварительного конфигурирования (поставляются они обычно в состоянии, в котором ведут себя как обычные коммутаторы). Для конфигурирования удобно использовать внеполосное (*out of band*) управление через консольный порт, поскольку при внутриволновом (*in band*) по неосторожности или неопытности можно попасть в "капкан" в какой-то момент из-за ошибки конфигурирования консоль может потерять связь с коммутатором.

Портам коммутаторов, поддерживающих 802.1Q, и участвующим в формировании ВЛС, назначаются специфические атрибуты. Каждому порту назначается PVID (Port VLAN Identifier) – идентификатор ВЛС для всех входящих на него немаркированных кадров. Коммутатор маркирует каждый входящий к нему НЕМАРКИРОВАННЫЙ кадр (вставляет номер VLAN в соответствие с PVID и приоритет, пересчитывает FCS), а

маркированные оставляет без изменений. В результате внутри коммутатора все кадры будут маркированными.

Порты могут конфигурироваться как маркированные или немаркированные члены ВЛС. Немаркированный член ВЛС (untagged member) выходящие через него кадры выпускает без тега (удаляя его и снова пересчитывая FCS). Маркированный член ВЛС (tagged member) выпускает все кадры маркированными. Для каждой ВЛС определяется список портов, являющихся ее членами. Порт может быть членом одной или более ВЛС. При конфигурировании для каждой ВЛС каждый порт должен быть объявлен как немаркированный (U), маркированный (T) или не являющийся членом данной VLAN (.). Если используются магистральные линии (Port Trunking или LinkSafe), то с точки зрения ВЛС данные порты представляют единое целое.

Построение виртуальных сетей 802.1Q на одном коммутаторе

Рассмотрим следующий пример (рисунок 6), который помогает понять принцип работы

виртуальных сетей 802.1Q. Машины 1 и 4 принадлежат VLAN 1, машины 2 и 5 принадле-

жат VLAN 2, а машина 3 является общедоступным ресурсом и включена в VLAN 3.

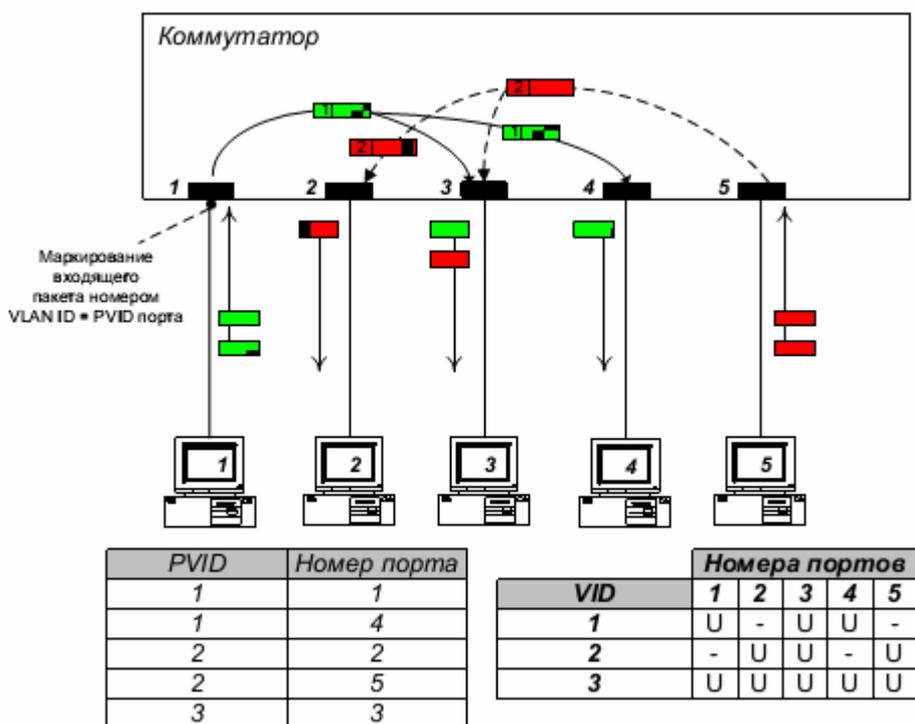


Рисунок 6. Пример настройки виртуальной сети 802.1Q на одном коммутаторе.

В соответствии с распределением по виртуальным сетям портам назначаются PVID как указано в таблице, то есть все немаркированные пакеты, поступающие на коммутатор из сети, будут тегироваться в соответствие с PVID порта.

Будем считать, что все машины не поддерживают стандарт IEEE 802.1Q. Таким образом, из всех пакетов, поступающих на конечные компьютеры, тег должен удаляться.

Так как к порту 3 подключен сервер, то он должен принимать пакеты от всех машин, то есть из всех виртуальных сетей. Таким образом, порт 3 является нетеггируемым для всех VLAN, то есть пакеты всех виртуальных сетей будут передаваться в сеть через этот порт немаркированными.

Оставшиеся порты включаются не только в свою VLAN, но и в VLAN 3, так как они должны принимать пакеты не только из своей VLAN, но и из VLAN сервера.

Построение виртуальных сетей 802.1Q на нескольких коммутаторах

Как уже отмечалось, виртуальные сети 802.1Q могут быть построены на нескольких коммутаторах и охватывать всю локальную сеть. Рассмотрим следующий пример (рисунок 7).

Коммутаторы SW2 и SW3 поддерживают 802.1Q, SW1 поддерживает только ВЛС по портам, SW4 – коммутатор без поддержки ВЛС. Для того, чтобы в обе ВЛС V1 и V2 попали

узлы, подключенные к коммутаторам SW1 и SW2, между этими коммутаторами приходится прокладывать отдельные линии и занимать по порту на каждую ВЛС.

Порты 1 и 2 коммутатора SW2 конфигурируются как немаркированные (U), один для ВЛС V1 (PVID=1),

другой для V2 (PVID=2). Порт 8 у SW2 и 1 у SW3 объявляются

маркированным (T) для

ВЛС V2 и V3. Порты SW2 и SW3, к которым подключаются компьютеры, объявляются не-

маркированными членами соответствующих ВЛС, у этих портов PVID принимает значе-

ния 1, 2 и 3 (в соответствии с номером ВЛС). Членам ВЛС V2 и V3

разрешаем доступ в

Интернет через маршрутизатор, подключенный к порту 7 коммутатора

SW3. Для этого порт 7 конфигурируется как немаркированный член V2 и V3, это обеспечит прохождение всех кадров от пользователей Интернет к маршрутизатору. Для того, чтобы ответные кадры могли дойти до пользователей, назначим порту 7 коммутатора SW3 PVID=9. Это будет дополнительная ВЛС для доступа к Интернет. Эта ВЛС должна быть "прописана" и во всех портах SW2 и SW3, к которым подключаются пользователи Интернет (порты SW2.8 и SW3.1 будут маркированными членами ВЛС 9, остальные - немаркированными). Под словом "прописана" подразумеваем указание на членство этих портов в VLAN 9, но никак не назначение им PVID=9.

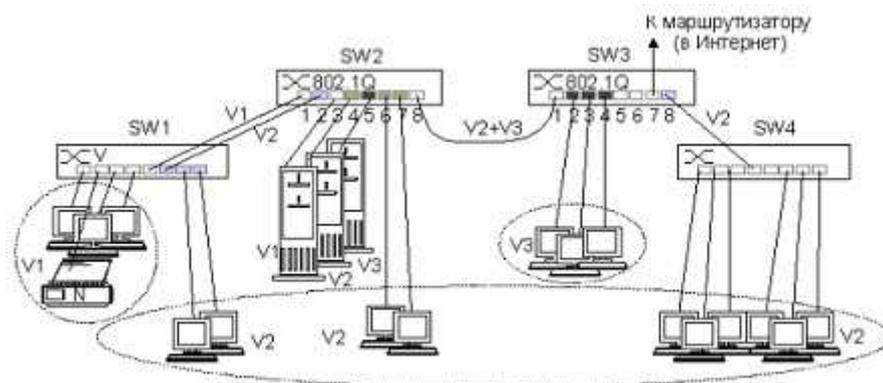


Рисунок 7. Сеть с распределенными ВЛС 802.1Q.

Реализации стандарта IEEE 802.1Q в сетевом оборудовании различных производителей

Стандарт IEEE 802.1Q определяет только формат используемых кадров Ethernet и минимальный набор требований к устройству, которые обязательны к реализации всеми производителями. Вместе с тем, очень большая область использования этой технологии отдается на откуп производителю оборудования. Современное сетевое оборудование от многих производителей по умолчанию позволяет производить демаркирование исходящих пакетов на текущем порту только для одной определенной VLAN. Например, пусть входящие пакеты для порта 1 маркируются PVID=1, то есть компьютеры на порту 1 входят в VLAN 1. Из исходящих с порта 1 кадров должен извлекаться тег, иначе кадр будет отброшен конечным компьютером (если тот не поддерживает IEEE 802.1Q). Так вот извлечь тег можно только с номером какой-то одной определенной VLAN. А что делать, если к этому порту подключены машины

из нескольких VLAN?

Для того, чтобы обойти это ограничение существует несколько способов:

1. Использование асимметричных виртуальных сетей. У некоторых коммутаторов есть возможность активировать данный тип VLAN (asymmetric vlan enabled/disabled). Это позволяет включать один порт в несколько VLAN.
2. Включение поддержки 802.1Q на оконечных устройствах – компьютерах.
3. Маршрутизация виртуальных сетей. Для этого необходимо, чтобы сетевой адаптер общедоступного ресурса (сервер, принтер и т.д.) поддерживал IEEE 802.1Q. Тогда схема будет следующей:
 - ✓ на сетевом адаптере общедоступного ресурса настраиваем несколько IP-подсетей. Каждую IP-подсеть привязываем к определённой VLAN.
 - ✓ компьютеры из каждой VLAN также помещаются в определённую IP-подсеть.

Если коммутатор не поддерживает асимметричных VLAN, то наиболее простой способ построения сети с общедоступными ресурсами – это организация поддержки 802.1Q на оконечных устройствах. При этом необязательно иметь сетевые адаптеры с данной поддержкой – существуют утилиты, которые позволяют запустить данный протокол даже на самых простых сетевых картах. Например, одной из таких утилит в Linux является vconfig. Правда если речь идёт о принтерах (то есть об оборудовании, ОС которого не допускает явного вмешательства), то остаётся только 3-ий способ.

Напоследок следует отметить, что в современных коммутаторах поддерживаются в лучшем случае два типа VLAN: по портам и IEEE 802.1Q.

6. Протоколы связующего дерева (Spanning Tree Protocols)

Для обеспечения надёжности работы сети зачастую необходимо использовать резервные линии связи. Базовые протоколы локальных сетей поддерживают только древовидные, то есть не содержащие замкнутых контуров, топологии связей. Это означает, что для организации альтернативных каналов требуются особые протоколы и технологии, выходящие

за рамки базовых, к которым относится Ethernet. Для автоматического перевода в резервное состояние всех альтернативных связей, не вписывающихся в топологию дерева, в локальных сетях используются алгоритм связующего дерева (Spanning Tree Algorithm, STA) и реализующий его протокол (Spanning Tree Protocol, STP) и расширения последнего.

6.1. Алгоритм связующего дерева (Spanning Tree Algorithm, IEEE

802.1d)

Алгоритм связующего (*примечание:* иногда покрывающего, распределяющего) дерева предназначен для изменения структуры, построенной на коммутаторах вычислительной сети, таким образом, чтобы в ней отсутствовали циклы, и чтобы при этом сеть сохранила связность. Изначально алгоритм был разработан компанией Digital Equipment Corporation. Впоследствии этот алгоритм был взят за основу при разработке комитетом IEEE спецификации 802.1d, в соответствии с которой обеспечивалась автоматическое изменение структуры сети, построенной на коммутаторах.

Для того чтобы алгоритм STA мог выполняться, необходимо чтобы в сети коммутаторов

были выполнены следующие предварительные установки:

- каждому коммутатору администратор должен назначить уникальный идентификатор, на основе которого будет получен приоритет коммутатора как комбинация MAC-адреса коммутатора и назначенного идентификатора;
- каждый порт коммутатора тоже должен иметь уникальный идентификатор;
- для каждого порта коммутатора должно быть определено значение относительной стоимости отправки кадров через данный порт, которое обычно обратно пропорционально пропускной способности подключенного канала.

После того, как были заданы указанные настройки, алгоритм STA предполагает выполнение нескольких последовательных этапов:

- выбор корневого моста;
- выбор корневых портов;
- выбор назначенных мостов;
- выбор назначенных портов;

изменение конфигурации сети.

Алгоритм выбора корневого моста

Корневым в алгоритме STA называется мост, расположенный в вершине дерева, в которое преобразуется существующая структура сети. Алгоритм STA предписывает использование в качестве критерия выбора значение приоритета моста, которое представляет собой MAC адрес данного моста. Для того чтобы системный администратор мог управлять процессом выбора корневого коммутатора, он может назначать для каждого коммутатора уникальный идентификатор, который и является приоритетом. На рисунке 8 представлена схема сети, которая построена с использованием коммутаторов и содержит несколько петель. В качестве корневого моста в данном случае будет выбран мост №1.

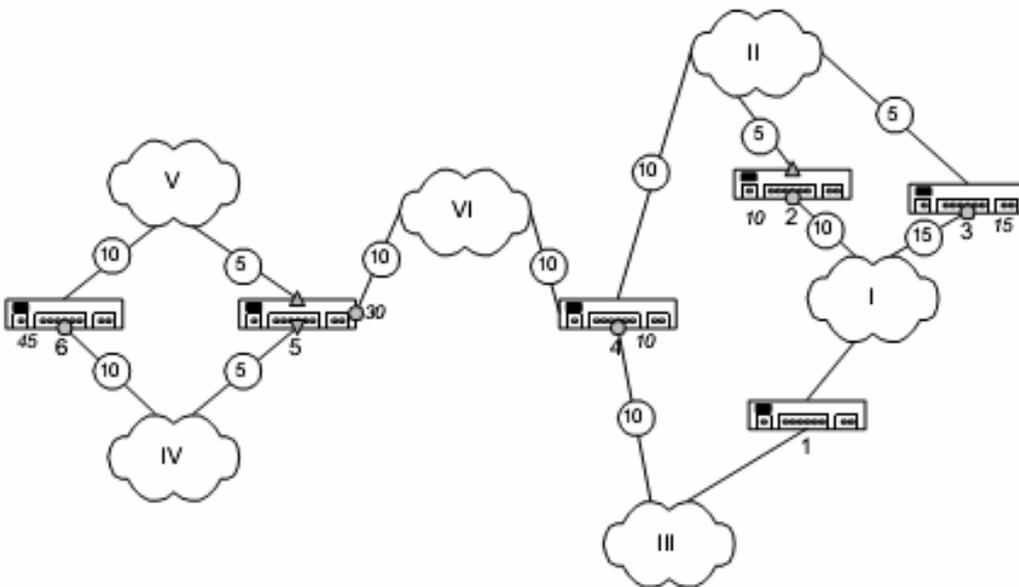


Рисунок 8. Пример работы алгоритма STA с сетью, содержащей петлевые структуры.

Алгоритм выбора корневых портов

После определения корневого моста все оставшиеся мосты выполняют процедуру определения корневого порта (root port). Корневым считается тот из портов моста, который связан наиболее дешевым путем с корневым мостом. В процессе определения корневого порта определяется также значение стоимости корневого маршрута (root path cost). На рисунке 7 корневые порты мостов отмечены точкой. Число, размещенное около порта, соответствует рассчитанному значению стоимости корневого маршрута.

Стоимость маршрута иногда называют расстоянием. Именно по этому критерию выбирается единственный порт, соединяющий каждый коммутатор с корневым, и единственный коммутатор, соединяющий каждый сегмент сети с корневым коммутатором. Стоимость маршрута определяется как суммарное условное время на передачу одного бита данных от порта данного коммутатора до порта корневого коммутатора. При этом временем внутренних передач (с порта на порт) пренебрегают, учитывается только время передачи по сегментам сети. Условное время сегмента рассчитывается как время, затрачиваемое на передачу одного бита информации в 10 наносекундных единицах между непосредственно связанными по сегменту сети портами. Для сегмента Ethernet это время равно 10 условным единицам.

В том случае, если несколько портов моста имеют одинаковое значение величины корневого маршрута, в качестве корневого выбирается порт, имеющий меньший порядковый номер, то есть наибольший приоритет.

Алгоритм определения назначенных мостов и портов

После определения корневых портов выполняется процедура определения назначенного моста для сети. На рисунке 7 сегменты сети обозначены римскими цифрами. В качестве назначенного для сегмента сети выбирается мост, который имеет минимальное значение стоимости корневого маршрута. В том случае, если несколько мостов будут иметь одинаковое значение величины корневого маршрута, в качестве назначенного будет выбран мост, который имеет минимальное значение идентификатора, то есть максимальный приоритет. В частности, для рассматриваемого варианта сети, назначенным мостом для сегмента II будет выбран мост №2, которому соответствует такое же значение этой величины, что и мосту №4. Аналогичным образом мост №5 будет выбран назначенным мостом для сегментов V и VI. Порт назначенного моста, который используется для подключения к выбранному сегменту

ту сети, называется назначенным портом. На рисунке 7 назначенные порты отмечены треугольником.

У корневого моста все порты являются назначенными, а их расстояние до корня полагается равным нулю. Корневого порта у корневого моста нет.

Изменение конфигурации сети по результатам выполнения алгоритма STA

После того, как были определены назначенные мосты и порты, может быть выполнена процедура изменения структуры сети. В соответствии с этой процедурой, все порты мостов, которые не получили статус корневых или назначенных, должны быть переведены в заблокированное состояние. В соответствии с этой процедурой должны быть заблокированы все порты мостов №3, 4 и 6 (рисунк 9). Также должны быть заблокированы порты, которые не получили статус корневых или назначенных, на корневом и назначенных мостах.

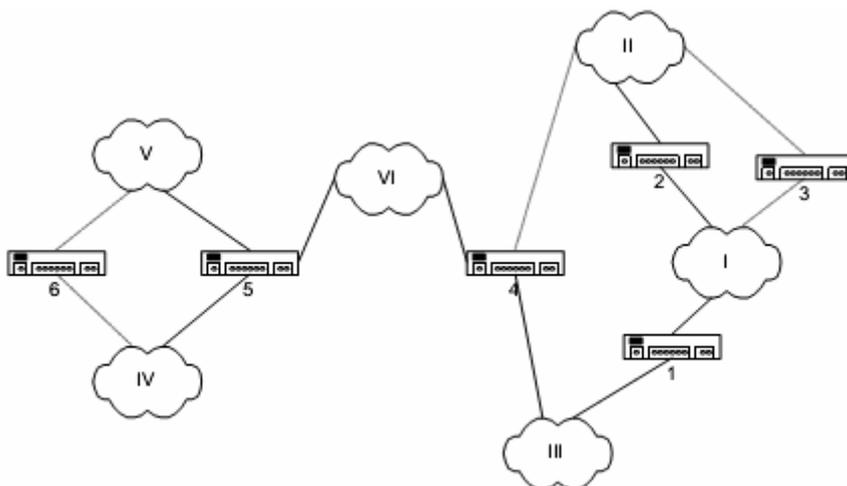


Рисунок 9. Топология сети после применения алгоритма связующего дерева.

Протокол связующего дерева (spanning Tree Protocol)

Для активизации протокола STP администратор должен активировать протокол, а затем задать следующие *обязательные настройки* на каждом коммутаторе в сети:

1. Уникальный идентификатор коммутатора, на основе которого будет получен приоритет коммутатора. Чем меньше значение идентификатора, тем выше будет приоритет коммутатора.
2. Уникальный идентификатор для каждого порта коммутатора. Чем меньше значение идентификатора, тем выше будет приоритет порта.

3. Стоимость отправки кадров через каждый порт коммутатора. Обычно коммутатор автоматически вычисляет стоимость для каждого собственного порта на основе пропускной способности подключенной к порту линии связи.

Также в зависимости от модели коммутатора администратор может дополнительно вручную задать значение следующих переменных:

1. Интервал приветствия Bridge Hello Time – это интервал между двумя посылками специального пакета BPDU (см. ниже), рассылаемого корневым коммутатором для оповещения остальных коммутаторов о том, что корневой коммутатор функционирует.
2. Максимальный возраст коммутатора bridge Max. Age . Если пакет BPDU не будет получен от корневого коммутатора по истечении максимального времени жизни, то ваш коммутатор будет считать, что корневой коммутатор перестал функционировать, то есть произошли изменения в конфигурации сети, а затем начнет посылать собственный BPDU всем другим коммутаторам для разрешения стать корневым мостом.

Замечание: необходимо на всех коммутаторах сети значение параметра Bridge

Max. Age выставлять большим значения параметра bridge Hello Time (см. формулы ниже), иначе постоянно будут происходить ошибки конфигурации.

3. Задержка смены состояния коммутатора Bridge Forward Delay – это время, которое затрачивается на перевод любого порта из режима блокировки (blocking state) в режим функционирования (forwarding state).

Для нормального функционирования протокола STP приведенные параметры должны удовлетворять следующему условию:

$$2 * (\text{Hello Time} + 1) \leq \text{Max Age} \leq 2 * (\text{Forward Delay} - 1)$$

После активации и осуществления необходимых настроек администратором протокол STP начинает функционировать. Чтобы выполнить все шаги, заложенные в алгоритме

STA, коммутаторы, работающие по протоколу STP, обмениваются между собой сообщениями в формате, определенном стандартом IEEE 802.1d. Эти сообщения называются блоками данных протокола мостов (BPDU, Bridge Protocol Data Unit) и содержат следующие поля:

Идентификатор версии протокола (2 байта). Коммутаторы должны поддерживать одну и ту же версию протокола STP, иначе может установиться активная конфигурация с петлями.

Тип сообщения (1 байт). Существует два типа сообщений BPDU – конфигурационный BPDU, то есть заявка на возможность стать корневым коммутатором, на основании которой происходит определение активной конфигурации, и BPDU уведомления о реконфигурации, которое посылается коммутатором, обнаружившим сбой, требующее проведения реконфигурации .отказ линии связи, отказ порта, изменение приоритетов коммутатора или портов.

Флаги (1 байт). Содержит два 1-битных флага:

- Бит 0. Topology Change (Изменение топологии) – указывает на то, что сообщение было послано для того, чтобы сигнализировать об изменении в составе сети (сообщение реконфигурации);
- Бит 7. Topology Change Acknowledgment (Подтверждение изменения топологии) – используется для подтверждения получения сообщения реконфигурации с установленным битом 0.

Приоритет корневого коммутатора (8 байт) содержит в качестве младших байт

MAC-адрес корневого коммутатора и в качестве старших 2 байт идентификатор

корневого коммутатора, назначенный администратором.

Стоимость корневого пути (2 байта).

Приоритет коммутатора (8 байт) содержит 6-байтовый аппаратный адрес моста,

пославшего сообщение, и 2-байтовое значение заданного этому мосту идентификатора.

Идентификатор порта (2 байта). Определяет порт, через который было послано сообщение.

Возраст сообщения (2 байта). Указывает на время, прошедшее с момента отправки сообщения. Измеряется в единицах времени, кратных 0,5 секунды. Каждый коммутатор добавляет ко времени жизни пакета свое время задержки.

Максимальный возраст сообщения (2 байта). Задаёт ограничение возраста, по достижении которого сообщение должно быть удалено.

Интервал приветствия `Bridge Hello Time` (2 байта). Фиксирует временной интервал между конфигурационными сообщениями корневого моста.

Задержка смены состояний `Bridge Forward Delay` (2 байта).

Специфицирует промежуток времени, в течение которого коммутаторы должны ожидать завершения работы алгоритма STA после изменения топологии сети. Если ещё не все коммутаторы закончили работу алгоритма, то преждевременные передачи данных могут вызвать появление циклов.

Сообщения BPDU инкапсулируются стандартными кадрами протокола Канального уровня. Желательно, чтобы все коммутаторы поддерживали общий групповой адрес, с помощью которого кадры, содержащие пакеты BPDU, могли одновременно передаваться всем коммутаторам сети. В противном случае пакеты BPDU рассылаются широковещательно.

Пакеты BPDU не перенаправляются в другие сети.

Жизненный цикл протокола STP можно представить в виде следующей блок-схемы.



Рисунок 10.

Подробно рассмотрим каждый из этапов цикла.

Процесс конфигурации

Выбор корневого моста. После инициализации каждый коммутатор сначала считает себя корневым. Поэтому он начинает через интервал приветствия bridge Hello Time генерировать через все свои порты сообщения BPDU конфигурационного типа. В них он указывает свой приоритет в качестве приоритета корневого коммутатора, расстояние до корня устанавливается в 0, а в качестве идентификатора порта указывается приоритет того порта, через который передается BPDU. Как только коммутатор получает BPDU, в котором имеется приоритет корневого коммутатора, больше его собственного, он перестает генерировать свои собственные кадры BPDU, а начинает ретранслировать только кадры нового претендента на звание корневого коммутатора. При ретрансляции кадров он наращивает расстояние до корня, указанное в пришедшем BPDU, на условное время сегмента, по которому принят данный кадр.

Выбор корневого порта. При ретрансляции кадров каждый коммутатор для каждого своего порта запоминает минимальное расстояние до корня. При завершении процедуры установления конфигурации покрывающего дерева, каждый коммутатор находит свой корневой порт – это порт, который ближе других портов находится к корню дерева.

Выбор назначенных мостов и портов. Кроме этого, коммутаторы выбирают для каждого сегмента сети назначенный порт. Для этого они исключают из рассмотрения свой корневой порт. Далее для всех своих оставшихся портов сравнивают принятые по ним минимальные расстояния до корня. Порт с минимальным расстоянием является назначенным портом. Когда имеется несколько портов с одинаковым кратчайшим расстоянием до корневого коммутатора, выбирается порт с наименьшим идентификатором (следовательно, с наибольшим приоритетом). Все порты, кроме назначенных и корневого, переводятся в заблокированное состояние и на этом построение покрывающего дерева заканчивается.

Функционирование сети

В процессе нормальной работы корневой коммутатор продолжает генерировать служебные пакеты BPDU через интервал bridge Hello Time , а остальные коммутаторы продолжают их принимать своими корневыми портами и ретранслировать назначенными.

Возникновение события, требующего проведение реконфигурации

В процессе функционирования сети с установившейся древовидной топологией следующие события приводят к инициализации новой процедуры построения покрывающего дерева:

- если по истечении максимального времени жизни корневой порт любого коммутатора сети не получит пакет BPDU от корневого коммутатора.
- если пакет BPDU, генерируемый корневым коммутатором, будет получен любым не корневым портом любого назначенного коммутатора.

Рассылка сообщений о реконфигурации

Коммутатор, обнаруживший одно из вышеперечисленных событий, в первую очередь оповещает об этом корневой коммутатор (рисунок 11).

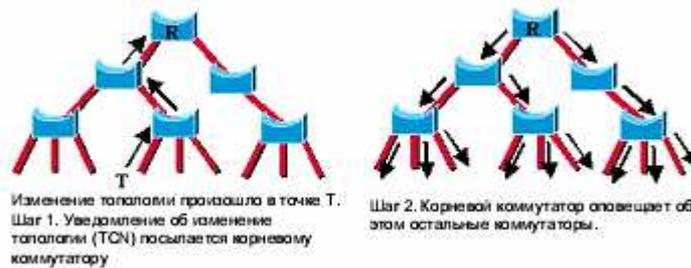


Рисунок 11.

Как только корень узнает о происшедшем изменении, он выставляет флаг TCV сообщениях BPDU, которые он рассылает. Затем данные сообщения распространяются по всей сети. В этом случае граф сети выстраивается заново в соответствии с вышеописанным алгоритмом.

У пакета BPDU с уведомлением о реконфигурации отсутствуют все поля, кроме первых двух.

Состояния портов коммутатора

Таким образом, в процессе построения топологии сети каждый порт коммутатора проходит несколько стадий:

Blocking (Заблокирован). При инициализации коммутатора все порты (за исключением отключенных) автоматически переводятся в состояние "Заблокирован". В

этом случае порт генерирует, принимает, обрабатывает и ретранслирует только пакеты BPDU. Все остальные пакеты не передаются.

Listening (Прослушивание). В начальный момент работы алгоритма STA порты коммутатора переходят в состояние "Прослушивание". В этот момент пакеты

BPDU от других коммутаторов еще не получены и коммутатор считает себя корневым, а все свои порты – назначенными. В этом режиме порт может находиться до истечения таймера смены состояний (Forwarding Timer). Значение таймера берет-

ся из значения переменной $bridge\ Forward\ Delay$ и может устанавливаться админи-

стратором (см. выше). В этом режиме порт продолжает генерировать, принимать,

обрабатывать и ретранслировать только пакеты BPDU. Если в течение этого вре-

мени порт получит BPDU с лучшими параметрами, чем собственные (расстояние, идентификатор коммутатора или порта), то он перейдет в состояние "Заблокирован". В противном случае порт переводится в состояние "Обучение". Learning (Обучение). Порт начинает принимать пакеты и на основе адресов источника строить таблицу коммутации. Порт в этом состоянии все еще не продвигает пакеты. Порт продолжает участвовать в работе алгоритма STA, и при поступлении BPDU с лучшими параметрами переходит в состояние Blocking "Заблокирован". Forwarding (Продвижение). Только после двукратной выдержки по таймеру порт переходит в состояние —Продвижение и обрабатывает пакеты данных в соответствии с построенной таблицей Disable (Отключен). В это состояние порт переводит администратор. Отключенный порт не участвует ни в работе протокола STP, ни в продвижении пакетов данных. Порт можно также вручную включить и он сначала перейдет в состояние Blocking.

Временная диаграмма состояния портов приведена ниже.

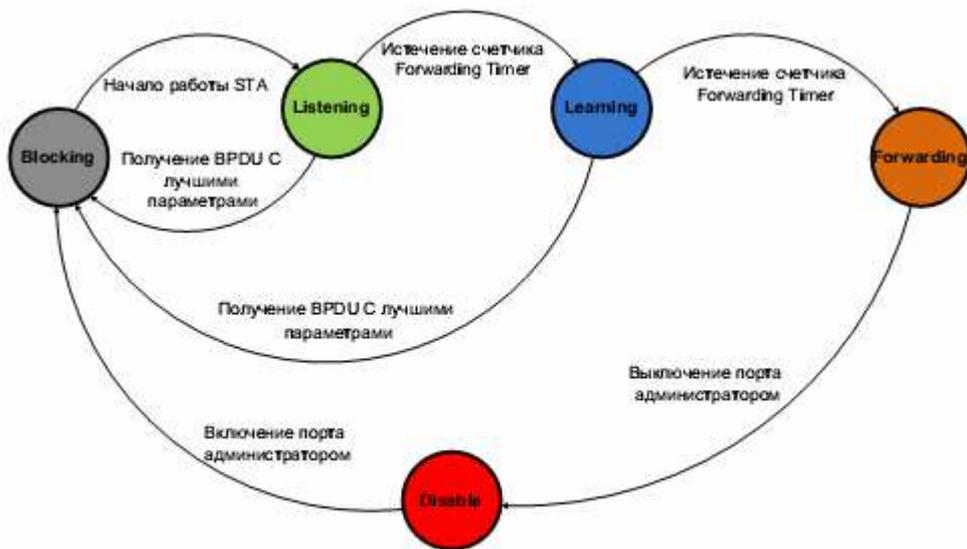


Рисунок 12.

Недостатки протокола STP:

Можно выделить следующие основные недостатки протокола STP:

1. Медленное время восстановления (сходимости) сети после аварии. При использовании настроек по умолчанию, это время может достигать нескольких минут в не-

большой сети для восстановления после простого обрыва соединения. Пока идет процесс восстановления, пользователи оторваны от сети, и большинство приложений закрывают свои сессии по тайм.ауту до того, как работа сети восстановится.

С точки зрения пользователей это большое неудобство, так как они будут вынуж-

дены переустанавливать все сессии, открытые приложениями.

2. Большое количество соединений в сети, использующей STP, находится в заблокированном состоянии и не передает данные. Таким образом, значительная часть пропускной способности сети не используется.

Данные недостатки призваны устранить следующие поколения протокола STP – Rapid

STP и Multiple STP, которые рассмотрены ниже.

6.2. Алгоритм быстрого связующего дерева (Rapid Spanning Tree

Algorithm,

IEEE 802.1w)

Стандарт на протокол связующего дерева IEEE 802.1d Spanning Tree Protocol был разработан в 1983 году, когда время восстановления сети после сбоя в течении 1 минуты считалось достаточно приемлемым. Протокол IEEE 802.1w Rapid Spanning Tree (RSTP) был

предложен многими производителями для уменьшения времени простоя клиентских пор-

тов во время процедуры восстановления сети, использующей STP. Протокол RSTP может

рассматриваться скорее как эволюция протокола STP, нежели как революция. Терминология стандарта IEEE 802.1d в основном осталась прежней. Большинство параметров ос-

тались без изменений. Таким образом, пользователи, хорошо знакомые со стандартом

IEEE 802.1d, смогут быстро понять и сконфигурировать сеть на основе нового стандарта

IEEE 802.1w. Стандарт IEEE 802.1w совместим со стандартом IEEE 802.1d, но при этом

теряются достоинства протокола RSTP.

тремя достоинствами протокола RSTP.

тремя достоинствами протокола RSTP.

тремя достоинствами протокола RSTP.

Основные отличия стандарта IEEE 802.1w от стандарта IEEE 802.1d

Отличие 1. Состояния портов коммутатора

Согласно стандарту на протокол RSTP существует только три состояния порта:

Discarding (Отвергающий), Состояния «Отключен», «Заблокирован» и «Прослуши-

вание» стандарта IEEE 802.1d объединены в стандарте IEEE 802.1w в одно состояние «Отвергающий».

Learning (Обучение).

Forwarding (Продвижение).

Таким образом, временная диаграмма состояния портов для протокола RSTP выглядит следующим образом.



Рисунок 13.

Отличие 2. Роли портов

Так же как и в протоколе STP остались роли корневого и назначенного порта, но протокол

RSTP вводит еще две дополнительные роли – альтернативный порт (alternative port) и

резервный порт (backup port). Протокол STP определяет роль порта на основе пакетов

BPDU. Грубо говоря, всегда существует метод для сравнения любых двух пакетов BPDU

для того, чтобы решить, какой из них более приоритетный. Данное сравнение осуществ-

ляется на основе значений, записанных в поля «Стоимость корневого пути», «Приоритет

коммутатора» и «Идентификатор порта». Корневые и назначенные порты в протоколе

RSTP определяются и назначаются точно также, как и в протоколе STP.

Поэтому имеет

смысл показать, каким образом назначаются альтернативные и резервные порты.

Альтернативный порт. Альтернативный порт является заблокированным портом, кото-

рый получает более приоритетные пакеты BPDU от другого моста (рисунок 14).

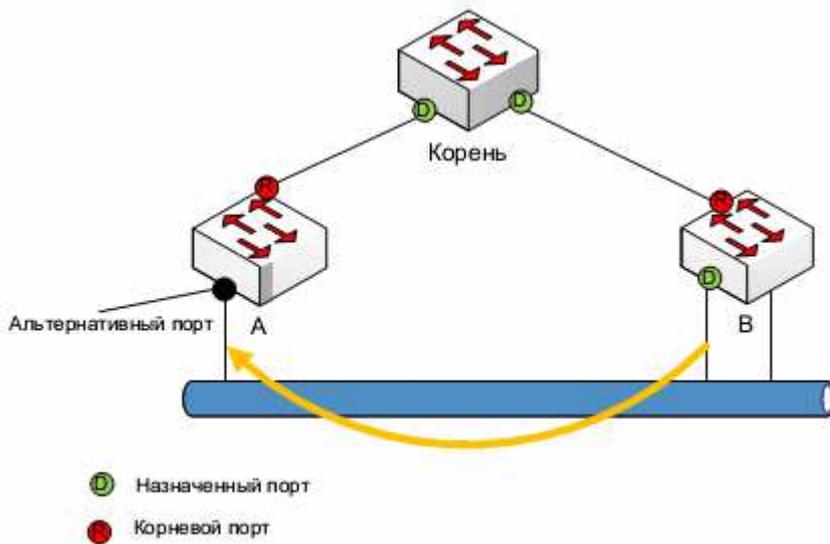


Рисунок 14.

Резервный порт. Резервный порт является заблокированным портом, который получает более приоритетные пакеты BPDU от своего же моста (рисунок 15).

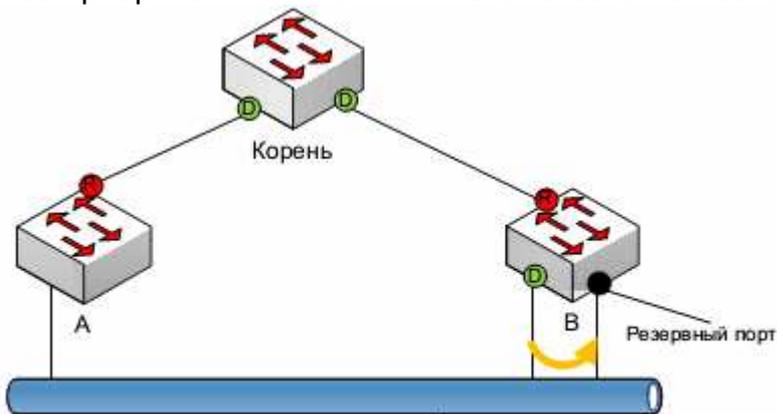


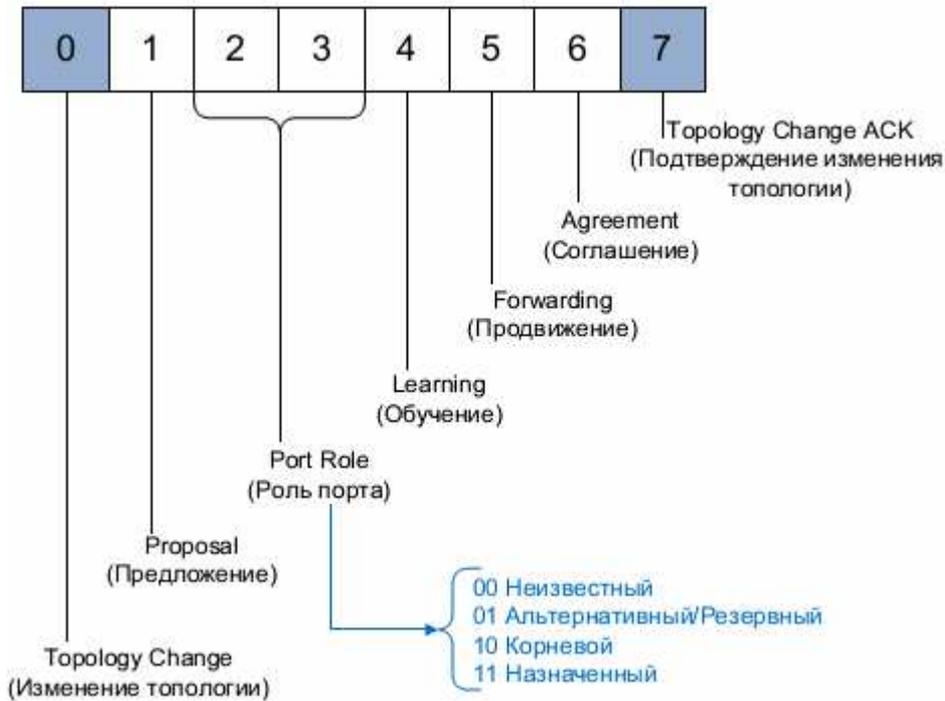
Рисунок 15.

Альтернативный порт обеспечивает альтернативный маршрут к корневому мосту и, следовательно, может заменить текущий назначенный коммутатор. Резервный порт обеспечивает всего лишь дополнительный канал к корневому мосту.

Отличие 3. Новый формат пакета BPDU

Некоторые изменения затронули и формат пакета BPDU. В стандарте IEEE 802.1d определено всего два флага (см. выше). Стандарт IEEE 802.1w использует все 8 бит поля «Флаги» пакета BPDU. Новые флаги обеспечивают выполнение двух основных функций:

- кодирование роли и состояния порта, который сгенерировал пакет BPDU;
- осуществление механизма «предложение/соглашение».



Другое важное изменение, принесенное стандартом IEEE 802.1w в формат пакета DPDU, – это значение поля «Идентификатор версии протокола» содержит значение 2. Мосты, работающие по стандарту IEEE 802.1d, будут отбрасывать пакет данного типа.

Отличие 4. Новый алгоритм работы с пакетами BPDU

Пакеты BPDU рассылаются с интервалом Bridge Hello Time всеми коммутаторами (а не только корневым) и больше не ретранслируются. Согласно стандарту IEEE 802.1d пакеты BPDU генерировал только корневой коммутатор с интервалом Hello Time, который может задаваться администратором сети. При этом остальные не корневые коммутаторы только ретранслировали пакеты BPDU, пришедшие на их корневой порт. Стандарт IEEE 802.1w предусматривает, что все коммутаторы с интервалом Hello Time рассылают пакеты BPDU с собственной информацией через все свои порты, даже если они не получили пакет BPDU от корневого моста. Таким образом, после установления устойчивой конфигурации сети (этап «Функционирование сети», см. выше «Жизненный цикл протокола STP») служебный трафик протокола RSTP несколько больше, чем у протокола STP. Учитывая пропускную способность каналов современных ЛВС, это вообще

нельзя считать недостатком.

Более быстрое устаревание информации. В протоколе RSTP следующие события могут привести к инициализации новой процедуры построения покрывающего дерева:

если по истечении максимального времени жизни корневой порт любого коммутатора сети не получит служебный пакет BPDU от корневого коммутатора (как и в протоколе STP);

если пакеты BPDU не будут получены три раза подряд (три интервала $_{\text{Hello Time}}$) от соседних коммутаторов.

Таким образом, теперь пакеты BPDU используются в качестве механизма определения работоспособности подключенной линии связи или коммутатора (keep-alive mechanism).

Коммутатор, не получивший пакет BPDU от другого коммутатора три раза подряд, считает, что связь с другим коммутатором потеряна. Подобный механизм позволяет быстрее определять неисправность в сети и запускать процедуру реконфигурации.

Обработка «худших» пакетов BPDU. Рассмотрим следующую ситуацию (рисунок 16). В

установившейся топологии сети происходит разрыв связи между корневым коммутатором и коммутатором В. Следовательно, не получив пакет BPDU от корневого коммутатора в течении времени Max Bridge Age, коммутатор В рассылает сообщение о реконфигурации, а затем конфигурационный пакет BPDU. В случае протокола STP всё это приводит к реконфигурации всей сети. При использовании же протокола RSTP события будут развиваться иначе.

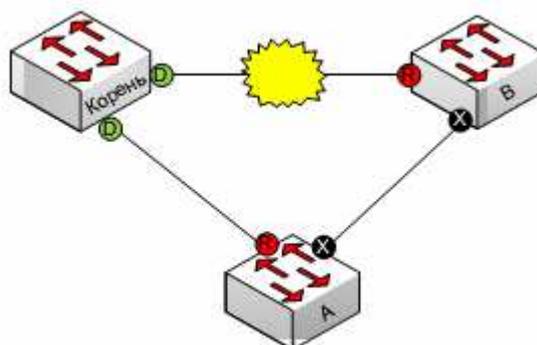


Рисунок 16.

Коммутатор А продолжает получать пакеты BPDU от корневого коммутатора, приоритет которых выше, чем пакеты BPDU коммутатора В. Поэтому коммутатор А сразу пересылает пакет BPDU корневого коммутатора коммутатору В. Следовательно, коммутатор В прекращает генерировать пакеты BPDU и определяет порт X как корневой.

Быстрый переход портов в состояние «Продвижения»

Быстрая конфигурация сети является самым важным нововведением в стандарте IEEE 802.1w. В сетях, в которых применяется классический протокол STP, уменьшение времени сходимости можно достичь только за счет уменьшения значений параметров Bridge Forward Delay и Bridge Max. Age, что зачастую приводит к нестабильности в работе протокола и сети. Для того, чтобы достичь более быстрой сходимости сети, протокол RSTP вводит два новых типа портов – пограничный порт (edge port) и порт точка-точка (point-to-point port, P2P port).

Пограничные порты

Пограничным портом является порт, непосредственно подключенный к сегменту, в котором не могут быть созданы петли. Например, порт непосредственно подключен к рабочей станции. Порт, который определен как пограничный, мгновенно переходит в состояние продвижения, минуя состояние обучения. Пограничный порт теряет свой статус и становится обычным портом связующего дерева в том случае, если получит пакет BPDU.

P2P порт

P2P порт обычно используется для подключения к другим мостам и также способен быстро перейти в состояние продвижения. При работе RSTP все порты, функционирующие в полнодуплексном режиме, рассматриваются как порты P2P, до тех пор, пока не будут переконфигурированы вручную.

Сходимость сети

При использовании протокола STP.

Рассмотрим пример, отражающий ситуацию, приводящую к реконфигурации сети при использовании протокола STP (рисунок 17).

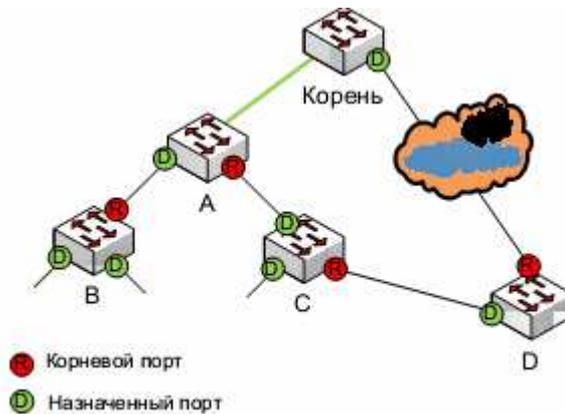


Рисунок 17.

На приведенной топологии добавлен новый канал между корневым коммутатором и коммутатором А. При этом между указанными коммутаторами уже есть связь через коммутаторы С и D (рисунок 18).

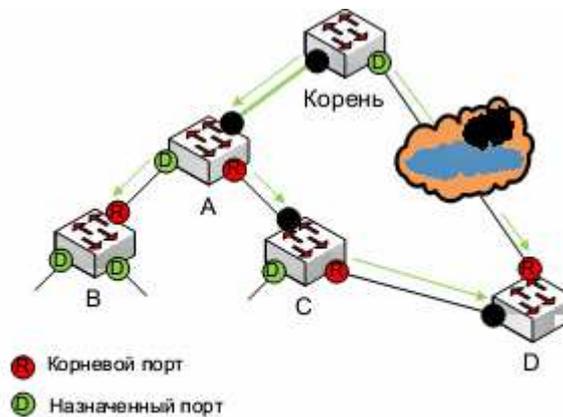


Рисунок 18.

Сейчас коммутатор А в состоянии прослушивать корневой порт напрямую. Таким образом, пакет BPDU от корня приходит не на корневой порт коммутатора А. А это одно из двух событий, приводящих к реконфигурации сети при использовании протокола STP. Таким образом, протокол STP блокирует порты на обоих концах нового канала, то есть порт корневого коммутатора и коммутатора А, и переводит эти порты в состояние «Прослушивание». Таким образом, исключается петля в сети.

Далее пакеты BPDU коммутатор А сразу же ретранслирует на все свои порты, и они попадают на назначенный порт коммутатора С и назначенный порт коммутатора D. Следовательно, коммутаторы С и D тоже понимают, что произошло изменение в конфигурации сети, блокируют все свои порты и рассылают сообщение о реконфигурации

сети. Теперь только через двойное время $\text{bridge Forward Delay}$ произойдет полное перестроение сети.

Теперь посмотрим, как протокол RSTP будет работать в аналогичной ситуации. Обратите внимание на то, что конечная топология сети будет такой же, как и в случае использования протокола STP. Только при использовании протокола RSTP для достижения данной топологии будет применена другая последовательность шагов, что займет гораздо меньше времени.

Также как и в протоколе STP сразу после добавления нового канала связи порты между коммутатором А и корневым коммутатором будут заблокированы (рисунок 19).

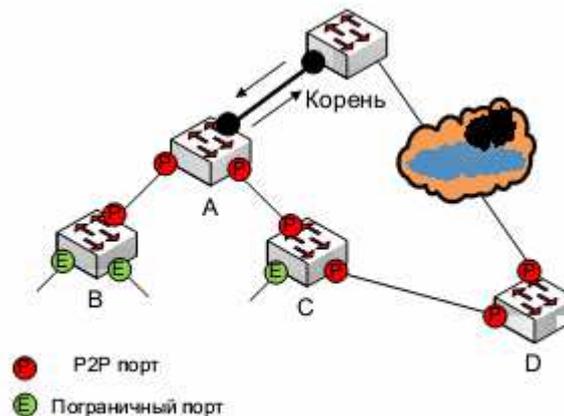


Рисунок 19.

После этого коммутатор А запустит процедуру `sync`, в рамках которой будут выполнены следующие действия:

- коммутатор А блокирует все свои P2P.порты;
- коммутатор А инициализирует процесс согласования с корнем, в рамках которого порты на обоих концах нового канала переводятся обратно в состояние `Forwarding`.

Далее процесс блокировки портов спускается ниже по дереву по мере того, как распространяются служебные пакеты BPDU корневого коммутатора через коммутатор А, и сеть принимает конфигурацию, показанную на рисунке 20.

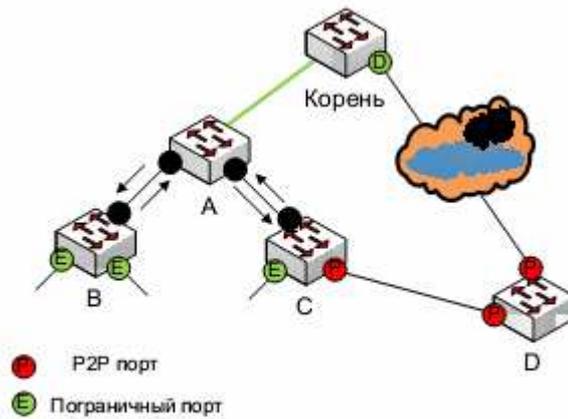


Рисунок 20.

Коммутаторы В и С также выполняют процедуру sync. Коммутатор В помимо заблокированного порта содержит только пограничные порты, поэтому в рамках процедуры sync он дополнительно не блокирует ни одного из своих портов, а только переводит канал между собой и коммутатором А в активное состояние. Коммутатор С блокирует порт, соединяющий его с D. В результате сеть принимает следующий вид (рисунок 21).

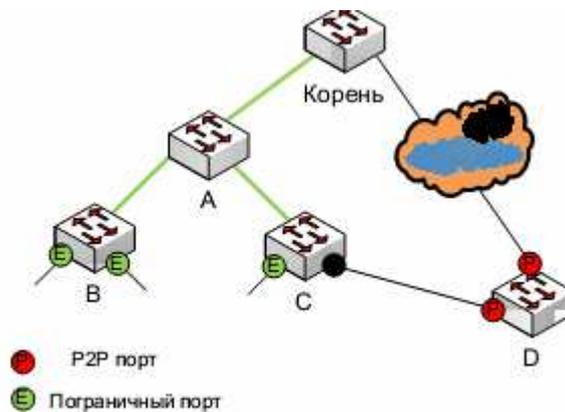


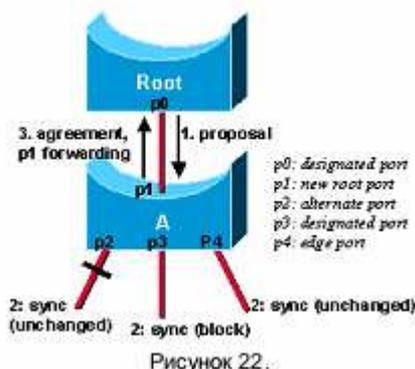
Рисунок 21.

В итоге достигается такая же конфигурация сети, как и при использовании протокола STP (блокируется порт коммутатора D, соединяющий его с коммутатором С). Время, затрачиваемое на перестроение сети, – это время, необходимое для спуска пакетов BPDU вниз по дереву, то есть в процессе реконфигурации не учитываются никакие таймеры.

Механизм «Предложение/Согласие»

После того, как согласно алгоритму STA порт стал назначенным, пройдет еще двойное время $F_{0}rward Delay$, и только тогда порт перейдет в состояние

продвижения. При использовании протокола RSTP переход портов в состояние продвижения осуществляется гораздо быстрее. Рассмотрим следующий пример (рисунок 22).

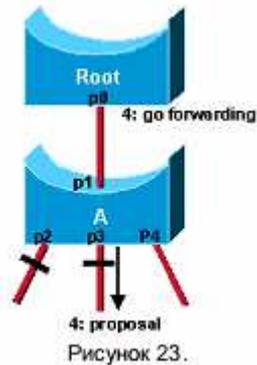


Предположим, что между корневым коммутатором и коммутатором А появился новый канал связи. Соответственно, оба назначенных порта на этом канале будут переведены в состояние блокировки. При этом, согласно протоколу RSTP, когда назначенный порт находится в состоянии блокировки (*Discarding*) или обучения (*Learning*) (и только в этом случае), он выставляет бит «Предложение» (*proposal bit*) в BPDU, который он генерирует. Поэтому порт p_0 корневого коммутатора посылает именно такой пакет BPDU (шаг 1). Так как коммутатор А получает более приоритетную информацию (пакет BPDU, в котором расстояние до корня меньше, чем во всех пакетах BPDU, получаемых до возникновения нового канала связи), то он немедленно назначает порт p_1 новым корневым портом. Далее коммутатор А запускает процедуру *sync* (Шаг 2). В данной процедуре не участвуют следующие типы портов:

- пограничный;
- заблокированный.

Для того, чтобы проиллюстрировать работу процедуры *sync* над различными типами портов, предположим, что на коммутаторе А существует альтернативный порт p_2 , назначенный порт p_3 в состоянии продвижения и пограничный порт p_4 . Заметим, что в проводимой процедуре *sync* порты p_2 и p_4 участвовать не будут. Поэтому коммутатору А для удачного завершения процедуры *sync* необходимо заблокировать только порт

p3. После этого коммутатор A может разблокировать свой новый корневой порт p1 и послать через него пакет BPDU корневому коммутатору с выставленным битом «Согласие» (Agreement bit) (Шаг 3). Сразу после получения портом p0 данного пакета, он сразу переходит в состояние продвижения (Шаг 4, рисунок 23).



После выполнения четвертого шага порт p3 находится в такой же ситуации, как и порт p0 на первом шаге. Поэтому порт p3 начнет посылать своим соседям пакеты BPDU с выставленным битом предложения.

Таким образом, можно выделить следующие основные черты механизма «Предложение/Соглашение»:

- механизм очень быстрый и не использует в своей работе никаких таймеров. «Волна рукопожатий» быстро распространяется по направлению к границе сети и быстро восстанавливает работоспособность сети после происходящих изменений в топологии;
- если заблокированный назначенный порт не получает пакет BPDU с выставленным битом «Соглашение» он выполняет стандартную процедуру перехода в состояние продвижения (discarding-learning-listen), тем самым замедляя процедуру перехода и увеличивая время реконфигурации сети. Это может случиться, если удаленный коммутатор не поддерживает протокол RSTP или порт удаленного коммутатора заблокирован.

UplinkFast

Другой формой немедленного перевода порта в состояние продвижения является механизм UplinkFast. Когда корневой порт выходит из строя, коммутатор переводит свой наилучший альтернативный порт в режим продвижения. Выбор альтернативного порта в качестве корневого изменяет топологию сети, но при этом типичная процедура реконфигурации сети путем рассылки широковещательных (или multicast) сообщений BPDU не запускается. Просто мосты, стоящие выше в дереве иерархии, очищают соответствующие записи в таблице Content Addressable Memory. Данный механизм не требует какой-либо конфигурации со стороны администратора, так как в протоколе RSTP он является встроенным.

Новый механизм изменения топологии

Механизм изменения топологии в протоколе RSTP сильно переработан.

Обнаружение изменения топологии

Согласно стандарту IEEE 802.1w, только не пограничные порты, переведенные в состояние продвижения, могут генерировать события изменения топологии. При обнаружении изменения топологии протокол RSTP в первую очередь производит следующие действия:

- запускает на всех не пограничных портах и корневом порту таймер TC While Timer на время, равное двум величинам Hello Time. Во время работы таймера TC While Timer через порт посылаются сообщения BPDU с выставленным флагом TC;
- очищает все записи в таблице перенаправления, связанные с данными портами.

Распространение объявления об изменении топологии

Когда коммутатор получает сообщение BPDU с выставленным флагом TC, он производит следующие операции:

- очищает все записи в таблице перенаправления, связанные со всеми его портами за исключением того, по которому было получено сообщение BPDU;
- запускает таймер TC While Timer и рассылает сообщения BPDU с выставленным флагом TC через все свои назначенные порты и корневой порт.

Таким образом, сообщения об изменении топологии распространяются

очень быстро по всей сети. Распространение данных сообщений теперь укладывается в один шаг. То есть коммутатор, обнаруживший изменение топологии, распространяет сообщения через всю сеть в отличие от протокола STP, где данную функцию может выполнять только корневой коммутатор (рисунок 24).



Рисунок 24.

За несколько секунд большинство записей в таблице перенаправления сбрасываются у всех коммутаторов сети. С одной стороны это приводит к временному «флуду» в сети (который необходим для повторного заполнения таблицы), а с другой – к удалению устаревших записей, которые мешают быстрому восстановлению связности сети.

Совместимость со стандартом IEEE 802.1d

Протокол RSTP в состоянии взаимодействовать с протокол STP, но при этом теряются все достоинства стандарта IEEE 802.1w, связанные с быстрой сходимостью сетевой топологии.

Недостатки стандарта IEEE 802.1w

Хотя протокол RSTP имеет гораздо более высокие показатели по надежности и отказоустойчивости по сравнению со стандартным протоколом STP, он сохраняет такие недостатки, как долгое время восстановления сети. И поэтому любые версии Spanning Tree не применимы в сетях, которые требуют надежности в 99,999%.

6.3. Множественные группы Spanning Tree (Multiple Spanning Tree, IEEE 802.1s)

MSTP расширяет стандарт IEEE 802.1w (RSTP) для поддержки нескольких копий STP. Это

расширение обеспечивает как быструю сходимость сети, так и возможность баланса нагрузки в сети с настроенными VLAN. Стандарт 802.1s MSTP также вносит дополнения и в стандарт 802.1Q. Протокол MSTP обратно совместим с протоколами 802.1d и 802.1w и позволяет настраивать несколько независимых "деревьев" STP в разных VLAN – администратор может группировать и назначать VLAN на отдельные "связующие деревья". Каждое такое "дерево" может иметь свою независимую от других "деревьев" топологию. Подобная новая архитектура обеспечивает несколько разных вариантов для передачи данных и позволяет организовать баланс нагрузки. Это свойство улучшает отказоустойчивость сети к возможным сбоям, так как сбой соединений в отдельном "дереве" (маршруте передачи данных) не отразится на других "деревьях" и, соответственно, возможных маршрутах. Также благодаря MSTP облегчается задача администрирования и управления крупными сетями: можно использовать резервные маршруты передачи данных путем настройки нескольких VLAN и настройкой независимых "деревьев" на поучившихся сегментах сети.

Проиллюстрируем работу MSTP на простом примере (рисунок 25).

Допустим, сеть состоит из 3 коммутаторов, соединенных между собой. В сети настроены 2 VLAN с VID 10 и 20. На коммутаторе 1 VLAN 10 и 20 настроены на разных портах таким образом, что трафик для обоих VLAN 10 и 20 передается по разным соединениям. На первый взгляд, такая конфигурация достаточно обычна и хорошо подходит для балансировки нагрузки при передаче трафика двух различных VLAN. Однако в сети настроен протокол STP.

Если коммутатор 3 будет выбран корневым коммутатором для STP, то соединение между коммутаторами 1 и 2 будет заблокировано. В этом случае трафик из VLAN 20 не сможет передаваться по сети. Эта проблема возникает потому, что коммутаторы рассматривают VLAN 10 и 20 как независимые сети, в то время как протокол STP рассматривает топологию сети как одну целую сеть.

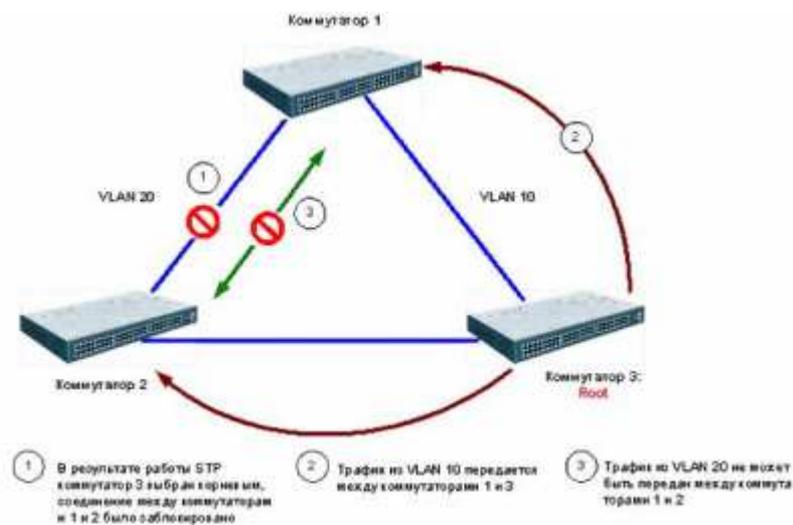


Рисунок 25.

Одним из решений подобной проблемы может стать настройка независимых копий STP на коммутаторах. Но настройка независимых копий STP не применяется, т.к. это может привести к значительному снижению производительности сети. К тому же, для многих сетей нет необходимости выполнять сложные настройки. Гораздо предпочтительнее настроить единую структуру STP на всей сети.

Для нормального взаимодействия нескольких устройств, они должны сопоставлять различные VLAN с несколькими возможными вариантами прохождения трафика .копиями дерева STP. Именно для этого и служит протокол 802.1S MSTP. Возвращаясь к примеру, можно увидеть, что 802.1s действительно решает поставленную задачу: если назначить VLAN 10 на копию MSTP под номером 1, а VLAN 20 сопоставить с копией 2. Т.о. получится две независимых топологии дерева STP. Коммутатор 3 становится корневым для копии MSTP номер 2 и блокирует прохождение трафика между коммутаторами 1 и 2. Но, в отличие от протокола 802.1D STP, это соединение блокируется только для прохождения трафика из VLAN 10. Трафик из VLAN 20 будет передаваться по этому соединению. Аналогичным образом копия MSTP под номером 2 выберет коммутатор 2 в качестве корневого и заблокирует соединение между коммутаторами 1 и 3 для трафика из VLAN 20.

Таким образом, достигается требуемая работа сети: осуществляется баланс нагрузки при передаче трафика нескольких VLAN по разным соединениям и в то же время в сети отсутствуют логические "петли".

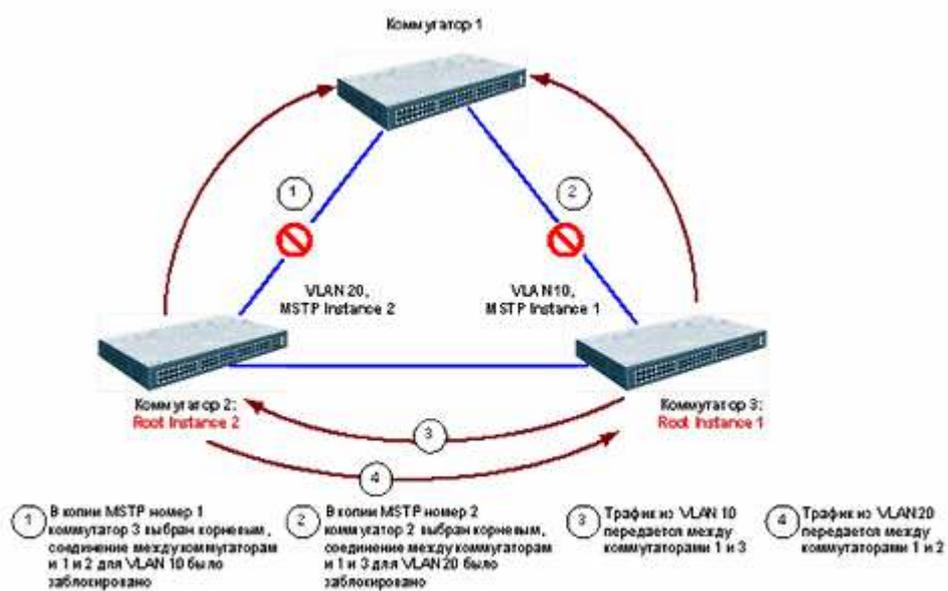


Рисунок 26.

Описание стенда

1. Типовой комплект учебного оборудования

«Локальные компьютерные сети»

1.1. Описание комплекта

Комплект состоит из одного коммутатора третьего уровня Cisco Catalyst 3560, двух управляемых коммутаторов второго уровня Cisco Catalyst 2960, двух неуправляемых коммутаторов Cisco SD205, 4 персональных компьютеров и 24-портовой коммутационной панели категории 5E, которая позволяет коммутировать устройства в зависимости от выполняемых задач и формировать необходимую топологию сети. На компьютерах установлена операционная система Arch Linux. Все компьютеры имеют три сетевых интерфейса (интегрированный в материнскую плату, на шине PCI и внешний USB), чтобы иметь возможность выполнять функции программного маршрутизатора. Внешние сетевые интерфейсы (eth1 и eth2) не поддерживают технологию MDI/MDX, поэтому возможно соединение двух компьютеров напрямую только кросс обжатым патч-кордом. Внешний вид комплекта представлен на рисунке 1.1.

Для регистрации на рабочих станциях используйте следующую учетную запись:
пользователь: *root*
пароль: *qwerty*

Разводка портов коммутационной панели приведена на самой панели.

Все узлы стенда включены в IP-подсеть 192.168.0.0/24. Распределение IP-адресов приведено в таблице 1.1.

Таблица 1.1 – Распределение IP-адресов

Узел	IP-адрес	
Компьютер 1	Интерфейс eth0	192.168.0.1
	Интерфейс eth1	192.168.1.1
	Интерфейс eth2	192.168.2.1
Компьютер 2	Интерфейс eth0	192.168.0.2
	Интерфейс eth1	192.168.1.2
	Интерфейс eth2	192.168.2.2
Компьютер 3	Интерфейс eth0	192.168.0.3
	Интерфейс eth1	192.168.1.3
	Интерфейс eth2	192.168.2.3
Компьютер 4	Интерфейс eth0	192.168.0.4
	Интерфейс eth1	192.168.1.4
	Интерфейс eth2	192.168.2.4
Управляемый коммутатор 3-го уровня	192.168.0.5	
Управляемый коммутатор 2-го уровня (верхний)	192.168.0.6	
Управляемый коммутатор 2-го уровня (нижний)	192.168.0.7	

1.2. Система восстановления операционных систем компьютеров «U-Profi Rescue System»

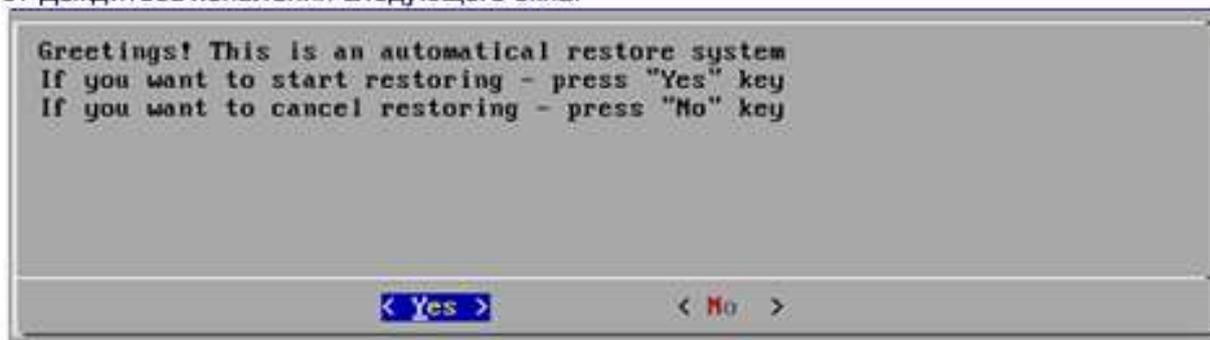
Комплект поставляется вместе с системой восстановления операционных систем (ОС), установленных на всех компьютерах. Данная система может оказаться полезной не только в случае отказа ОС Linux, но также в тех случаях, когда после проведения ряда лабораторных работ необходимо быстро привести ОС к первоначальному виду. Далее приведена инструкция по работе с системой восстановления:

1. **ВНИМАНИЕ!** Восстановление уничтожит все данные на жёстком диске. Сделайте резервные копии важных данных.
2. Перезагрузите компьютер и вставьте DVD-диск «Система восстановления ОС Linux» в привод. Соблюдайте соответствие номеров диска и компьютера.
3. Загрузите меню настройки BIOS. Выберите в меню загрузку с DVD-привода. Сохраните выполненные настройки.
4. Перезагрузите компьютер.
5. Дождитесь появления следующего окна:



Для выхода из системы клавишей Tab выделите кнопку «Выход» и нажмите Enter. Это приведёт к перезагрузке компьютера. Иначе, для продолжения работы нажмите любую клавишу.

6. Дождитесь появления следующего окна:



Чтобы начать восстановление, нажмите «Yes», иначе нажмите «No» или просто перезагрузите компьютер. В связи с техническими ограничениями на данный момент при нажатии клавиш навигации перерисовка экрана с диалогом пользователя не происходит. Однако сами нажатия учитываются.

7. Когда восстановление начнётся, вы увидите следующее окно:



В системе не реализован прогресс-индикатор (индикатор объема выполненной работы). Признаком того, что система не зависла, является индикатор обращения к жесткому диску ноутбука. Процесс восстановления в среднем занимает около 20 минут.

8. Когда восстановление будет закончено, появится следующее окно:



9. Нажмите «OK». После окончания процесса восстановления произойдет автоматическая перезагрузка. В момент полного завершения работы восстановительной системы (погас экран) отключите флэш-диск, чтобы загрузиться с внутреннего диска.

10. После загрузки ОС Arch Linux необходимо каждому из ноутбуков назначить индивидуальный IP-адрес в соответствии с таблицей 1.2 и индивидуальное сетевое имя. Для этого необходимо изменить содержимое конфигурационных файлов в соответствии с инструкциями в разделе 4.2.

Практическая часть

Лабораторная работа № 4.

Виртуальные локальные сети vLAN.

Цель работы:

Изучение технологий виртуальных сетей. Получение навыков настройки VLAN на основе тэгов IEEE 802.1q в сети, построенной на коммутаторах Cisco.

Порядок выполнения работы:

1. Изучите раздел 1.5 «Виртуальные сети. Сети на базе маркированных кадров» теоретического пособия.
2. Постройте топологию сети, представленную на рисунке 6.

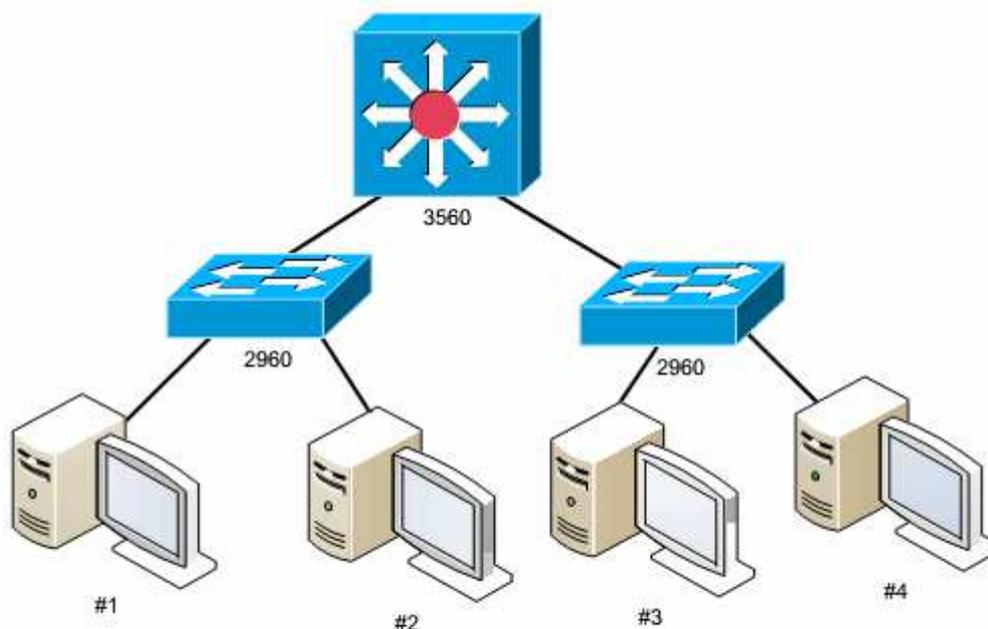


Рисунок 6. Коммутируемая топология для настройки виртуальных сетей VLAN.

3. Сконфигурируйте VLAN на основе тэгов таким образом, чтобы рабочие станции 1 и 4 принадлежали виртуальной сети №1, станция 2 принадлежала виртуальной сети №2, а станция №3 – виртуальной сети №3 и при этом являлась общедоступным ресурсом. То есть машины в виртуальных сетях №1 и №2 не должны взаимодействовать между собой, но должны взаимодействовать с машиной №3.
4. Проверьте правильность конфигурации сети. Результаты мониторинга покажите и поясните преподавателю.

5. Сбросьте настройки коммутатора в фабричные и перезагрузите его.

Лабораторная работа № 5.

Построение магистральных линий связи.

Цель работы:

Изучение технологии агрегирования каналов связи (Link Aggregation, Trunking).

Порядок выполнения работы:

1. Постройте топологию сети, показанную на рисунке 7.
2. Одновременно запустите процессы передачи файла с машины 1 на машину 3 и с машины 2 на машину 4. Определите время передачи (*подсказка: удобнее пользоваться утилитой ftp, которая автоматически определяет время передачи*).
3. Изучите раздел 1.4 «Объединение портов в магистральные линии связи» теоретического пособия.

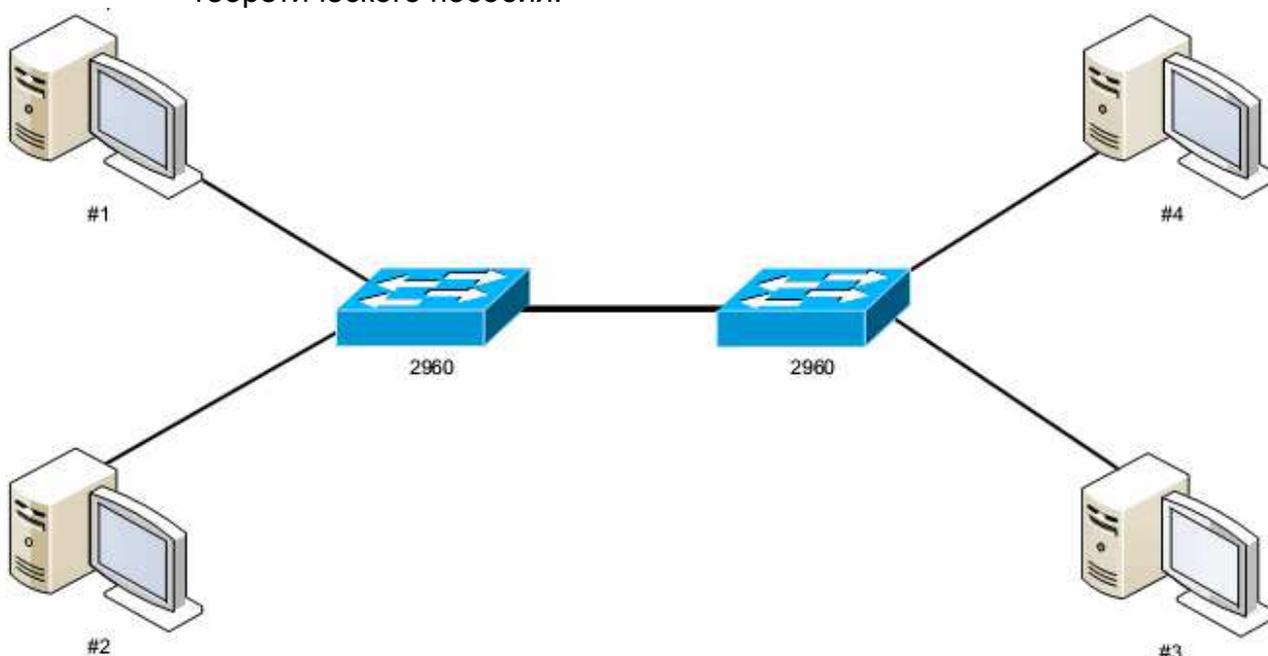


Рисунок 7. Топология коммутируемой сети.

4. Создайте магистраль между коммутаторами, включающую 2 порта.
5. Одновременно запустите процессы передачи файла с машины 1 на машину 3 и с машины 2 на машину 4. Определите время передачи и сравните его со временем, полученным в пункте 10.
6. Удалите созданную магистраль и соедините коммутаторы через гигабитные порты.
7. Одновременно запустите процессы передачи файла с машины 1 на машину 3 и с машины 2 на машину 4. Определите время передачи и сравните его с предыдущими полученными значениями. Сделайте вывод.

Лабораторная работа № 6.

Протокол IGMP.

Цель работы:

Изучение протокола IGMP и способов его настройки на коммутаторах и маршрутизаторах.

Порядок выполнения работы:

1. Постройте топологию сети, показанную на рисунке 8.

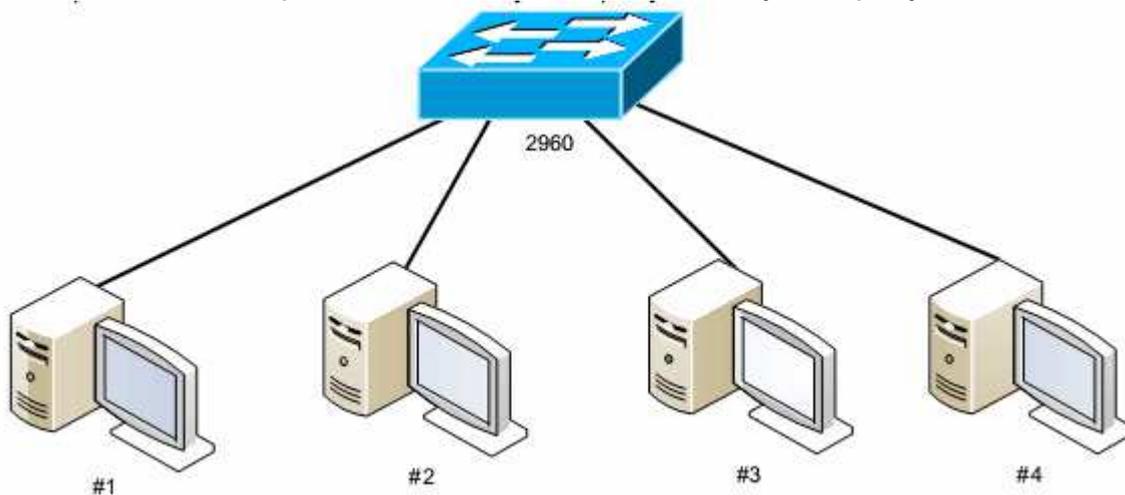


Рисунок 8.

2. Изучите раздел 3.2 «Протокол IGMP» теоретического пособия.
3. Создайте на коммутаторе дополнительную виртуальную сеть «multicast», которая включает машины 1 и 3.

4. Для vLAN «multicast» включите поддержку протокола IGMP.

5. Запустите на машине 1 сервер вещания медиапотока.

6. Сделайте попытку принять данный поток со всех остальных машин.

7. Ответьте, какие машины в итоге получают медиапоток, а какие – нет.

8. С помощью утилиты tcpdump отследите пути распространения мультикаст.

трафика в сети.

9. Постройте топологию сети, показанную на рисунке 9.

10. Создайте две IP-подсети на коммутаторе 3560 и настройте протокол IGMP для каждой подсети.

11. На обоих коммутаторах 2960 настройте протокол IGMP, используя функцию IGMP Snooping.

12. Запустите на машинах 1 и 3 сервера вещания. Настройте машину 2 на получения

медиапотока от сервера 3, а машину 4 – от сервера 1.

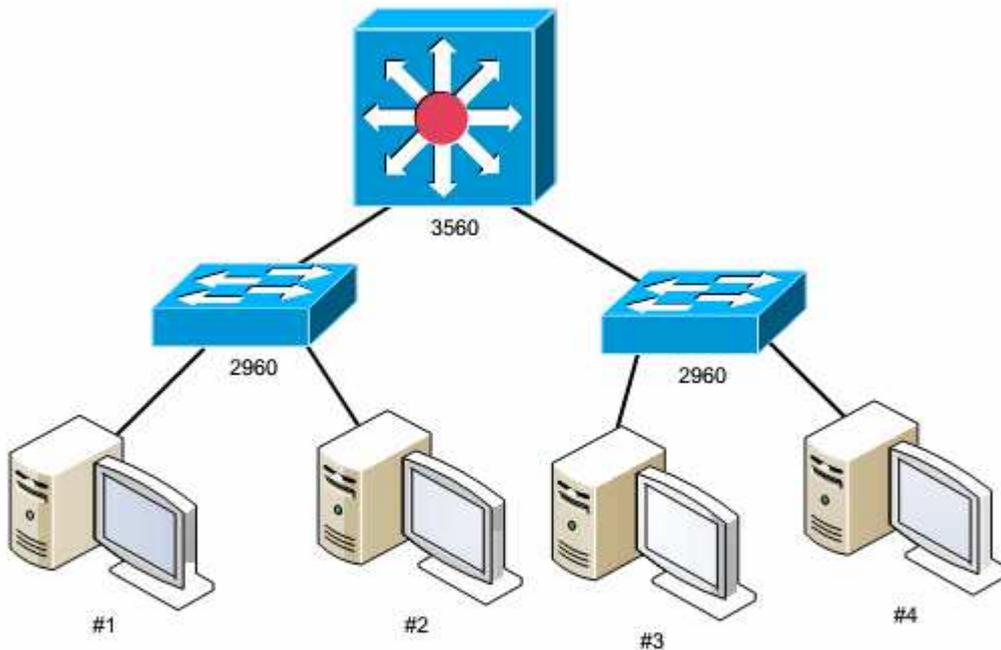


Рисунок 9.

13. Убедитесь в том, что сеть функционирует правильно.
14. Используя средства мониторинга коммутатора 3560 выясните, какие VMP-группы присутствуют в сети и кто в них входит.
15. Сбросьте настройки коммутаторов в фабричные и перезагрузите их.

Лабораторная работа № 7.

Алгоритмы связующего дерева *spanning Tree*.

Цель работы:

Изучение алгоритма Spanning Tree и Rapid Spanning Tree. Получение навыков построения сети с поддержкой резервных линий связи.

Порядок выполнения работы:

1. Изучите раздел 1.6 «Протоколы связующего дерева» теоретического пособия.
2. Соберите сеть с топологией, представленной на рисунке 10.

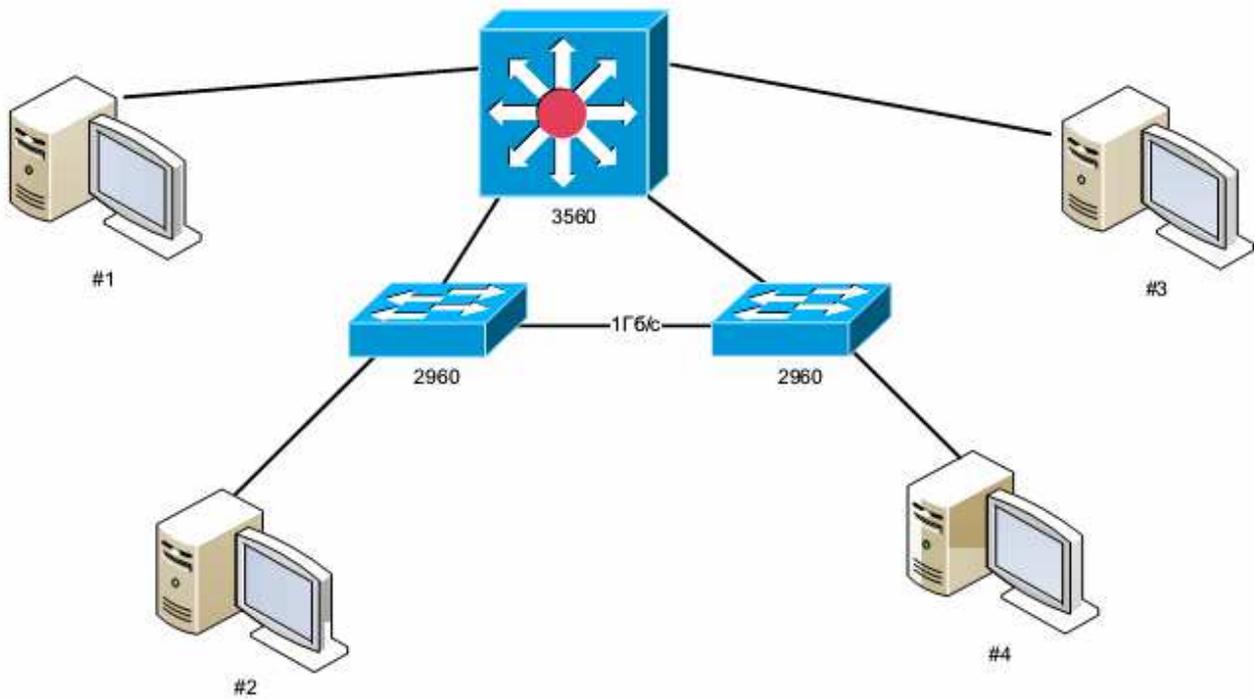


Рисунок 10. Топология коммутируемой сети.

3. Активизируйте на всех коммутаторах протокол Spanning Tree.
4. Настройте приоритеты коммутаторов и портов таким образом, чтобы получить сеть, показанную на рисунке 11 (пунктиром показана резервная связь)
5. С помощью системы мониторинга убедитесь, что через порт коммутатора 2960 #1, отмеченного черным квадратом, трафик не идет. Проверьте остальные задействованные в работе порты. Результаты мониторинга сохраните.
6. Имитируйте сбой в сети, нарушив соединение, выделенное на рисунке 8 жирной линией. В соответствии с алгоритмом Spanning Tree сеть должна быть переконфигурирована. Убедитесь в этом с помощью системы мониторинга

коммутаторов. Определите время сходимости сети. Результаты сохраните.

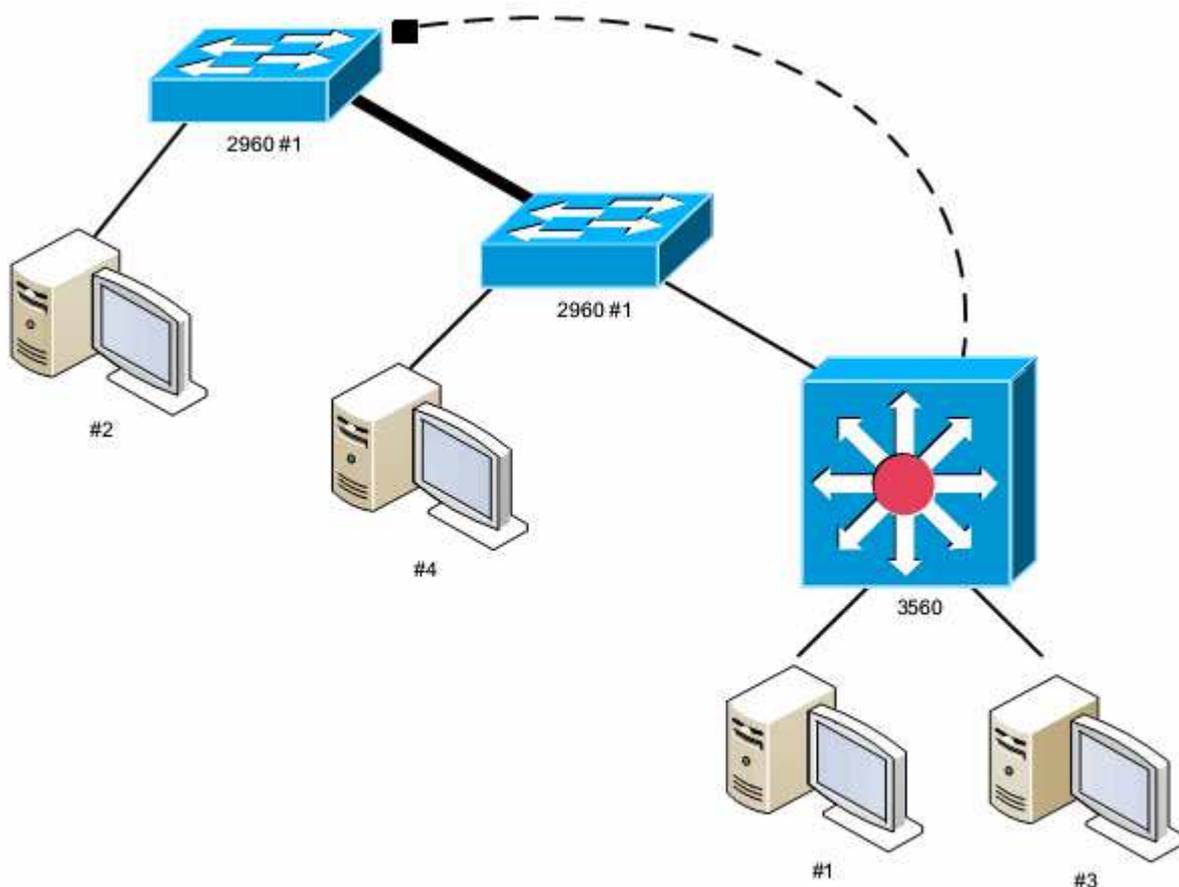


Рисунок 11. Топология сети после применения алгоритма STP.

7. Активизируйте на всех коммутаторах протокол Rapid Spanning Tree.
8. Повторите действия пунктов 4.6.
9. Сравните времена сходимости, полученные при использовании двух протоколов: STP и RSTP. Сделайте выводы.
10. Сбросьте настройки коммутатора в фабричные и перезагрузите его.

Требования к отчету

Отчет должен содержать цель работы, краткие теоретические сведения, задания, методики и результаты выполнения, ответы на контрольные вопросы, выводы.

Отчет предоставляется к сроку проведения следующей работы и зачитывается преподавателем в результате контрольной беседы.

Контрольные вопросы

1. Эталонная модель OSI. Эталонная модель TCP/IP.
2. Физический уровень. Физические носители.
3. Канальный уровень. Службы канального уровня. Сетевые адаптеры.
4. Обнаружение и исправление ошибок.
5. Контроль четности. Двухмерный контроль четности.
6. Контрольная сумма. Циклический избыточный код.
7. Протоколы коллективного доступа.
8. Протоколы последовательного доступа.
9. Адресация в локальных сетях. Протокол ARP.
10. Протокол Ethernet. Структура Ethernet-кадра.
11. Немодулированная передача и Манчестерское кодирование.
12. Протокол CSMA/CD. Экспоненциальный откат.
13. Концентраторы, коммутаторы, мосты.
14. Алгоритм связующего дерева.

Литература

1. В.Г.Олифер, Н.А.Олифер. Компьютерные сети. Принципы, технологии, протоколы. Учебник для вузов. 4-е издание./- СПб: Издательство «Питер», 2011.
2. Олифер В.Г., Олифер Н.А. Основы компьютерных сетей. Учебное пособие – СПб.: Издательство «Питер», 2009.
3. Таненбаум Э., Уэзеролл Д. Компьютерные сети. 5-е изд. – СПб.: Издательство «Питер», 2012.
4. Н.А. Олифер, В.Г. Олифер Средства анализа и оптимизации локальных сетей. © Центр Информационных Технологий, 1998.
5. Microsoft Corporation. Администрирование сети Microsoft Windows NT. Учебный курс. Пер. с англ. Под ред. А.В.Кузьмина – М.: Издательство – торговый дом «Русская редакция», 1999.
6. В.Г. Олифер, Н. А. Олифер. Новые технологии и оборудование IP-сетей. – СПб: БХВ, 2000.
7. М. Кульгин. Технологии корпоративных сетей. Энциклопедия. – СПб: Изд-во “Питер”, 1999г.
8. М. Гук. Аппаратные средства локальных сетей. Энциклопедия. – СПб: Изд-во “Питер”, 2000.
9. Мониторинг в корпоративных сетях. Апрышкина Г. КомпьютерПресс 7'2001.
10. Управление сетями. Правила для системного администратора. Апрышкина Г. КомпьютерПресс 5'2001
11. Распределенная система управления сетью. Чашков Ю. Еженедельник "Computerworld", #18, 2001 год.