**Практическая работа № 3**

**Тема:** протоколы устранения петель (STP) и агрегирование каналов (ETHERCHANNEL)

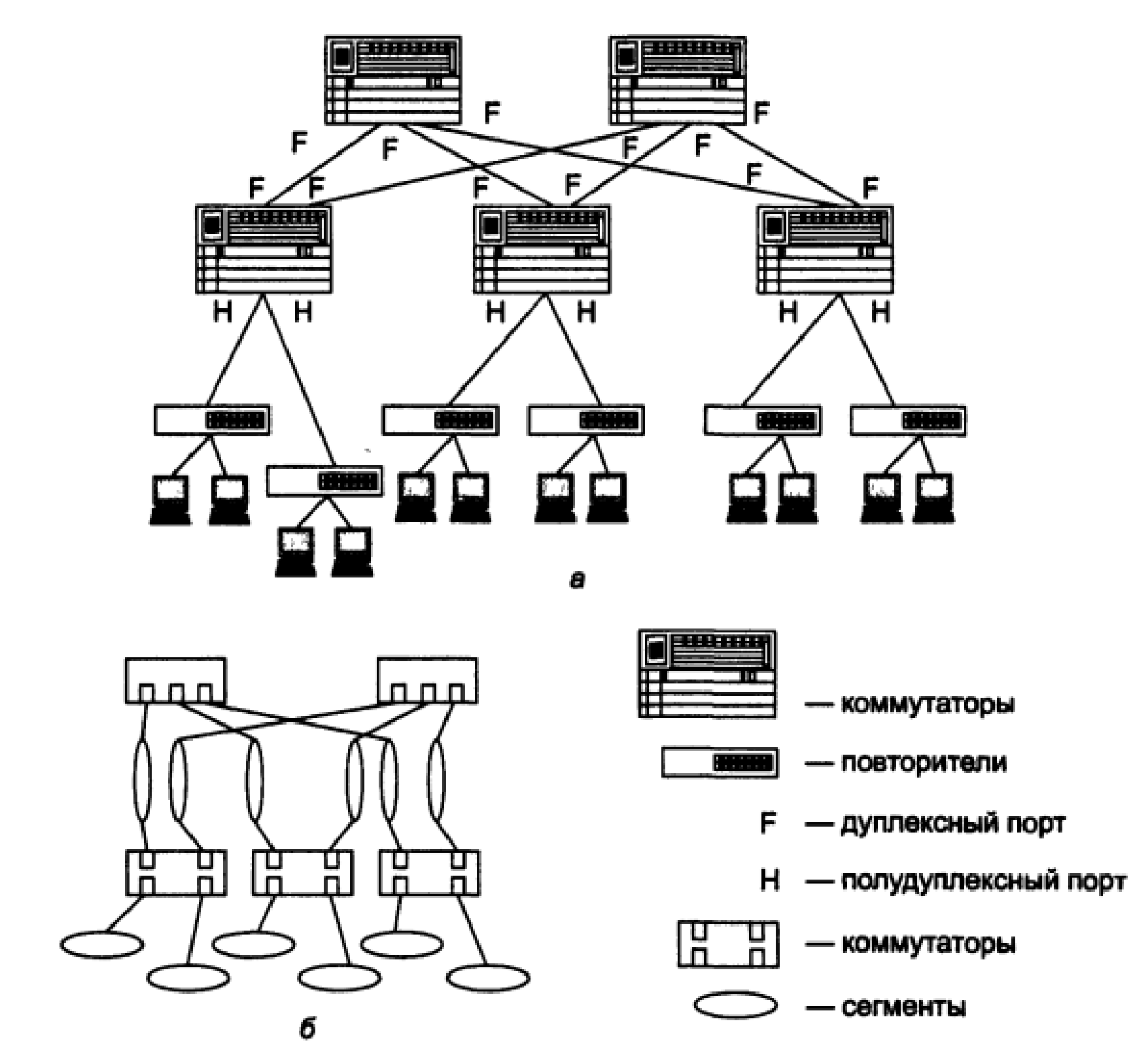
**Цель работы:** Изучить метод устранения петель с помощью протокола Spanning Tree Protocol (STP), а также изучить метод организации отказоустойчивых каналов - агрегирование каналов с помощью протокола Ether Channel.

**Используемые средства и оборудование:** IBM/PC совместимый компьютер с пакетом Cisco Packet Tracer; лабораторный стенд Cisco.

1. **КРАТКАЯ ТЕОРИЯ**

Протокол STP формализует сеть (рис. 3.1, а) в виде графа (рис. 3.1, б), вершинами которого являются коммутаторы и сегменты сети.

Сегмент — это связная часть сети, не содержащая коммутаторов (и маршрутизаторов). Сегмент может быть разделяемым (во время создания алгоритма STA это был единственный тип сегмента) и включать устройства физического уровня — повторители/концентраторы, существование которых коммутатор, будучи устройством канального уровня, «не замечает». Сегмент также может представлять собой двухточечный канал, в коммутируемых локальных сетях это единственный тип сегмента.

 о

**Рис. 3.1. Формализованное представление сети в соответствии с алгоритмом STA**

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

В качестве расстояния в STA используется метрика — традиционная для протоколов маршрутизации величина, обратно пропорциональная пропускной способности сегмента. В STA метрика определяется так же, как условное время передачи бита сегментом. В версии 802.1D-1998 эта величина является 16разрядной, а в версии 802.1 D-2004 — 32-разрядной.

В версии 1998 года выбраны следующие значения метрики: 10 Мбит/с — 100,100 Мбит/с -19,1 Гбит/с — 4,10 Гбит/с — 2. В текущей версии 802.1 D-2004 используются такие значения метрик, которые расширяют диапазон скоростей сегментов до 10 Тбит/с (то есть с большим запасом относительно сегодняшнего уровня максимальной для Ethernet скорости в 10 Гбит/с),’'давая такому сегменту значение 2; соответственно сегмент 100 Гбит/с получает значение 200, 10 Гбит/с — 2000, 1 Гбит/с — 20 000, 100 Мбит/с — 200 000, а 10 Мбит/с — 2 000 000.

Идентификатор коммутатора — это 8-байтовое число, шесть младших байтов которого составляют MAC-адрес его блока управления, отрабатывающего алгоритм STA (напомним, что портам коммутаторов и мостов для выполнения своей основной функции МАС-адреса не требуются), а два старших байта называются приоритетом коммутатора (значение по умолчанию равно 32 768) и конфигурируются вручную, что, как мы увидим далее, позволяет администратору сети влиять на процесс выбора корневого коммутатора.

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

Идентификатором порта служит 2-байтовое число. Младший байт содержит порядковый номер данного порта в коммутаторе, а значение старшего байта является приоритетом (значение по умолчанию равно 128) и задается администратором.

Назначенным коммутатором сегмента объявляется коммутатор, у которого расстояние до корневого коммутатора является минимальным.

Назначенный порт — это порт назначенного коммутатора сегмента, подключенный к данному сегменту.

Протокольными единицами данных моста (Bridge Protocol Data Unit, BPDU) называются специальные пакеты, которыми периодически обмениваются коммутаторы для автоматического определения конфигурации дерева. Пакеты BPDU переносят данные об идентификаторах коммутаторов и портов, а также о расстоянии до корневого коммутатора. Существует два типа сообщений, которые переносят пакеты BPDU: конфигурационные сообщения, называемые также сообщениями Hello, и сообщения с уведомлениями об изменении конфигурации. Для доставки BPDU используется групповой адрес 01:80:С2:00:00:00, позволяющий организовать эффективный обмен данными.

Интервал Hello — это интервал между генерацией сообщений Hello; он настраивается администратором и обычно составляет от 1 до 4 секунд; по умолчанию — 2 секунды.

EtherChannel-каналы третьего уровня.

 Канал EtherChannel представляет собой суммирование множества физических каналов в одно логическое соединение.

 Одно логическое соединение называется портом-каналом

(port channel).

 На некоторых коммутаторах можно настроить такой канал для функционирования в качестве интерфейса третьего уровня.

 При назначении IP-адреса каналу он становится логическим интерфейсом третьего уровня.

 Если какой-либо канал выхолит из строя, интерфейс канала остается доступным через другие каналы.

 Функции EtherChannel-канала третьего уровня подобны функциям группы EtherChannel второго уровня по распределению трафика и установке каналов.

Конфигурация функции.

Канал EiherChannel предоставляет возможность связывать множество физических соединений в целях обеспечения большей пропускной способности для каналов, по которым транспортируется трафик нескольких узлов. Поскольку канал EthcrChanncI функционирует «почти» на физическом уровне, в один канал могут быть связаны несколько интерфейсов третьего уровня. После того как канал был сформирован, виртуальный интерфейс, который называется копалом, начинает функционировать как канал третьего уровня для всех его членов. Настройка группы F.therChanncI включает в себя следующие этапы.

1. Доступ к процессору коммутации третьего уровня.

Если процессор коммутации представляет собой плату, которая функционирует как подсистема в COS-устройстве, то для конфигурирования необходимо получить доступ к этому устройству. Для этого используется команда session. При использовании коммутатора, работающего с системой Supervisor 10S, вводить указанную команду не нужно, поскольку в таком случае он непосредственно подключен к процессору коммутации третьего уровня.

Система COS session mod

(режим привилегированного пользователя)

Параметр mod указывает номер модуля процессора коммутации. Чтобы определить местоположение процессора в коммутаторе, если оно неизвестно, используется команда show modules. 2. Создание логического канала.

Система IOS interface port-channel number (режим глобальной конфигурации)

Для создания логического интерфейса канала эта команда вводится в режиме глобальной конфигурации. Интерфейс функционирует, как интерфейс третьего уровня для всех членов капала. Параметр number определяет номер группы канала, который будет назначен каждому члену канала. 3. Настройка протоколов в канале.

Система IOS ip address address netmas/c (режим конфигурирования интерфейса)

Для настройки сетевой адресации на интерфейсе третьего уровня используются соответствующие команды. В приведенном выше примере демонстрируется конфигурирование IP-адреса.

4. Назначение физических интерфейсов третьего уровня группе каналов.

а) Выбор интерфейса.

Система IOS interface type mod/poct

(режим глобальной конфигурации)

Нужно выбрать интерфейс третьего уровня для назначения группе каналов. Поскольку создается канал третьего уровня, должен использоваться интерфейс также третьего уровня. Для коммутаторов, допускающих функционирование интерфейса в качестве интерфейса второго или третьего уровня, вводится команда по switchport, обеспечивающая функционирование интерфейса на третьем уровне.

б) Удаление адресации всех протоколов.

Система IOS no ip address

(режим конфигурирования интерфейса)

Если интерфейсу присвоен адрес какого-либо протокола, например, IP, необходимо удалить адрес с помощью ключевого слова по и команды, устанавливающей адресацию. Например, для удаления IP-адреса с интерфейса используется команда no ip address.

в) Назначение интерфейса группе каналов.

Система IOS channel-group number node {auto | desirable | on}

(режим конфигурирования интерфейса)

Для физического интерфейса третьего уровня, который планируется использовать как часть канала, применяется команда channel-group. Параметр number указывает, с каким интерфейсом канала порта связан данный физический интерфейс. Режимы (mode) определяют взаимодействие одной стороны канала с другой стороной.

г) Проверка работоспособности интерфейса.

Система IOS no shutdown

(режим конфигурирования интерфейса)

Стандартным состоянием многих интерфейсов третьего уровня является shutdown (отключен). Чтобы интерфейс функционировал, следует включить его с помощью команды по shutdown.

д) Повторение этапов а-г для всех интерфейсов с

одинаковой скоростью, которые будут членами данного канала.

Проверка канала.

После настройки канала можно проверить его работоспособность с помощью описанных ниже команд.

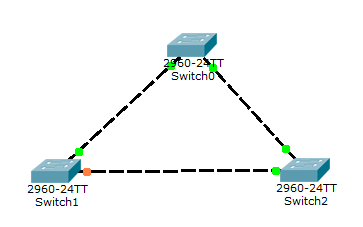
Система IOS show etherchannel number port-channel show interfaces type number etherchannel

(обе команды вводятся в режиме привилегированного пользователя)

При использовании команды show etherchannel параметр number определяет канал порта или номер группы для канала, состояние которого необходимо проверить. Команда show interfaces позволяет указать отдельные члены этого канала и просмотреть EtherChannel-парметры для данных интерфейсов.

2. **STP – УСТРАНЕНИЕ ПЕТЕЛЬ**

Открываем Cisco Packet Tracer и добавляем 3 коммутатора 2960. Соединяем их. Происходит инициализация портов, и алгоритм STP уже работает (рис. 1).



### Рисунок 1 – Схема соединения трех коммутаторов

Это можно увидеть, если переключиться в режим симуляции и посмотреть проходящие пакеты. Заглянем внутрь пакета. Можно увидеть, что протокол STP передает BPDU кадры. По умолчанию они передаются каждые 2 секунды.

Для того, чтобы определить какой коммутатор - корневой, зайдем в CLI switch 1 и перейдем в привилегированный режим. С помощью команды show spanning-tree можно увидеть, что данный коммутатор является корневым.

Все его порты находятся в режиме передачи и являются назначенными.

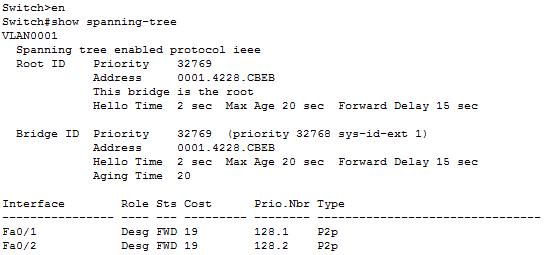


Рисунок 2 – Команда show spanning-tree для switch 0

Аналогично смотрим другие коммутаторы. Как видим, порт Fa0/1, который находится ближе к корневому коммутатору, является корневым, а другой порт является назначенным.

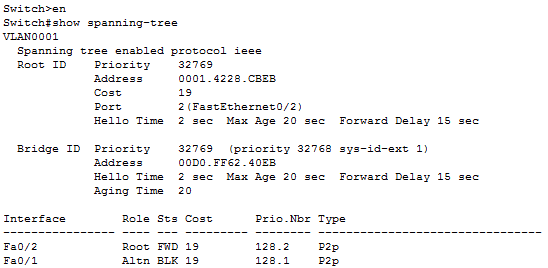


Рисунок 3 – Команда show spanning-tree для switch 2

Аналогично проверяем 3 коммутатор. Порт Fa0/2 является корневым и находится в состоянии передачи, а другой порт является заблокированным, так как на данный сегмент есть назначенный порт у коммутатора Switch 0. Этот порт является резервным и активизируется в случае падения одного из «линков».

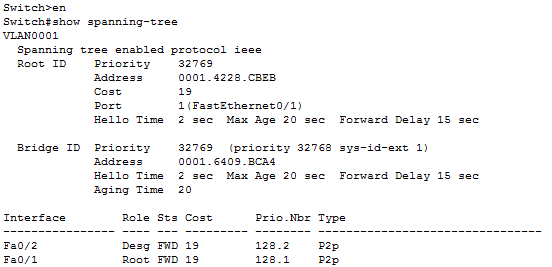
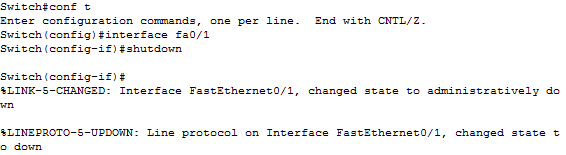


Рисунок 4 – Команда show spanning-tree для switch 1

Проверим, что протокол STP работает и попробуем потушить один из

«линков». Для этого нужно положить Fa0/1 на коммутаторе Switch 0. Заходим в режим конфигурирования интерфейса Fa0/1 и выключаем порт.



Если зайти на соседний коммутатор и набрать show spanning-tree, видно, что порт перешел в состояние прослушивания, затем в режим обучения и в режим передачи. Связь восстановилась при падении одного из активных «линков».

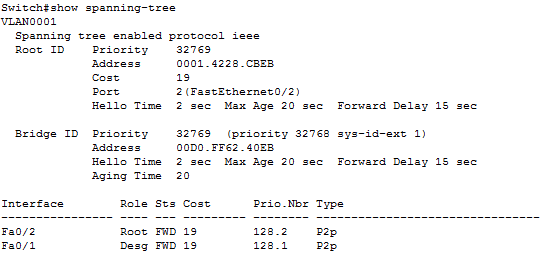
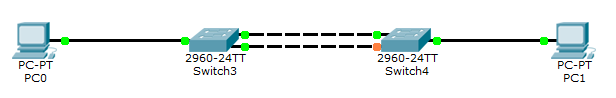


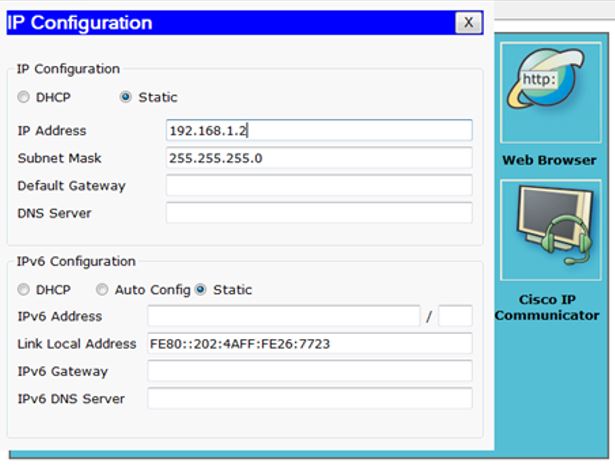
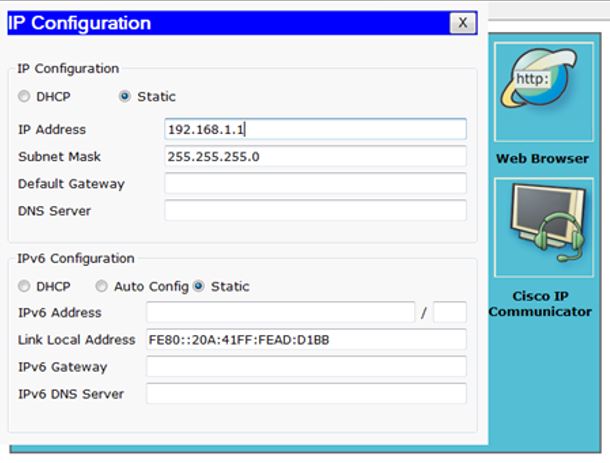
Рисунок 5 – Команда show spanning-tree для switch 1 после падения одного из линков

Рассмотрим другой пример. Соберем схему из 2 коммутаторов 2960 и 2 компьютеров. Соединим. Образовалась коммутационная петля и начинает работу алгоритм STP



### Рисунок 6 – Схема соединения с образованием коммутационной петли

### Настроим IP-адресацию на компьютерах (рис. 7).



### Рисунок 7 – Окна настройки IP-адресации

Проверим связь командой ping. Связь работает.

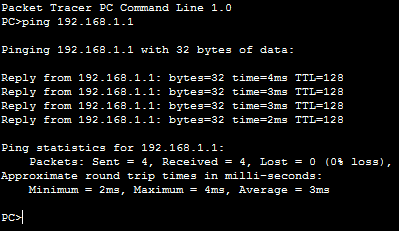
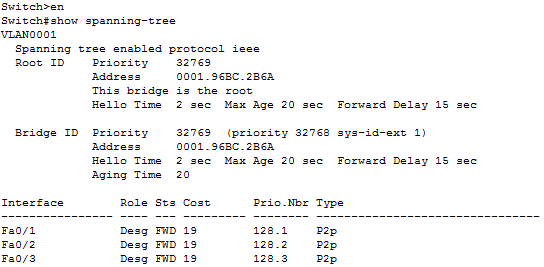
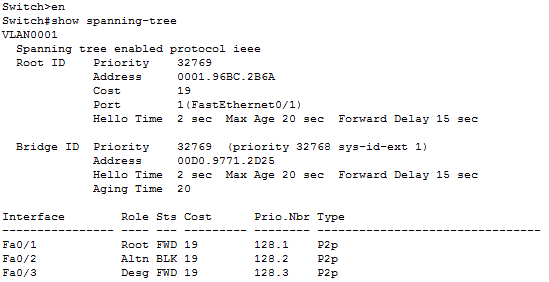


Рисунок 8 – Результат работы команды ping

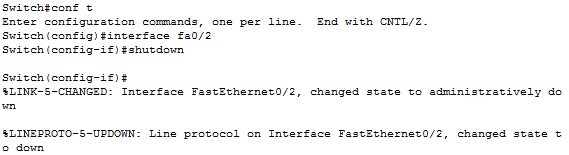
Протокол STP сделал свою работу и один из портов находится в режиме заблокированного. Рассмотрим с помощью команды show spanning-tree Switch 3. Коммутатор является корневым и все его порты в режиме передачи.

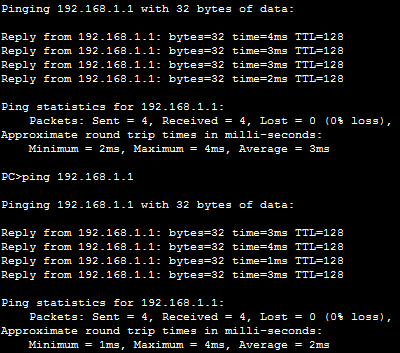


Аналогично рассмотрим Switch 4. Видно, что порт Fa0/2 заблокирован.



Посмотрим, как отразиться на пользователе время работы STP, то есть время сходимости. Для этого «потушим» порт Fa0/2 на Switch 3. Запустим ping. Видим, что связь нарушена





### Рисунок 9 – Результат работы команды ping

Происходит инициализации портов. Порт, который был заблокирован, переходит в состояние прослушивания, затем режим обучения и в режим передачи. Все это время связь между пользователями нарушена. Связь восстановилась в течение 15-20 секунд. Хотелось бы сократить время переключения. Для этого используется протокол RSTP. Настроим его. Для этого переходим к конфигурированию Switch 3, заходим в режим глобального конфигурирования, и вводим команду spanning-tree mode rapid - pvst.



Проделываем аналогичную операцию с Switch 4. Если воспользоваться командой show spanning-tree, можно увидеть, что включен режим RSTP (рис. 10).

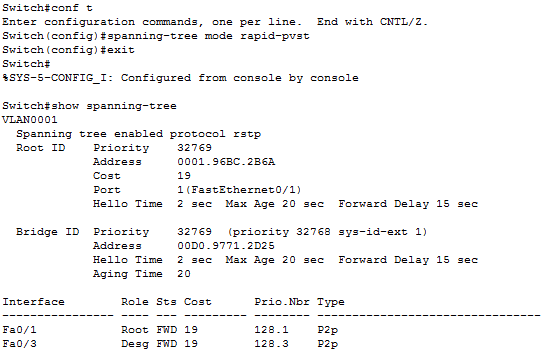
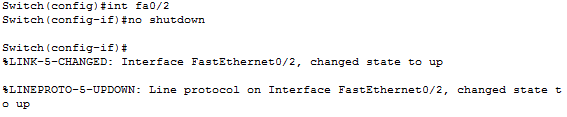


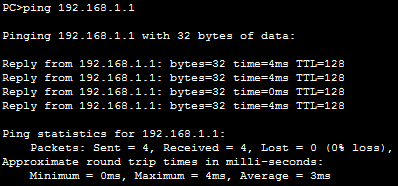
Рисунок 10 – Режим RSTP

Восстанавливаем работу коммутатора, на котором был «потушен» порт



### Рисунок 11 – Включение порта на коммутаторе Switch3

Переключение произошло моментально. Проверим связь командой ping. Рing успешен (рис. 12)



### Рисунок 12 – Результат работы команды ping

Выключаем порт, чтобы посмотреть насколько быстро произойдет переключение на резервный канал. Проверяем связь командой ping и выключаем порт. Как видим, переключение произошло мгновенно (рис. 3.10).

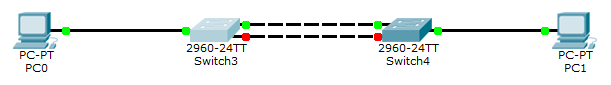
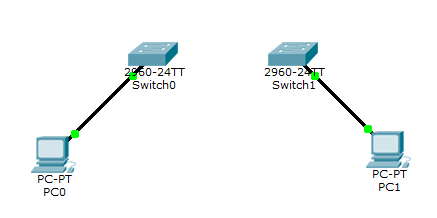


Рисунок 13 – Проверка переключения на резервный канал

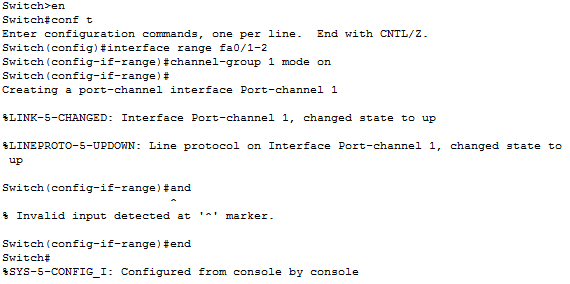
3. **АГРЕГАЦИЯ КАНАЛОВ – ETHERCHANNEL.**

Открываем Cisco Packet Tracer, добавляем 2 switch 2960 и 2 компьютера. Соединяем их. Пусть это будет порты FastEthernet 0/3 (рис. 13).

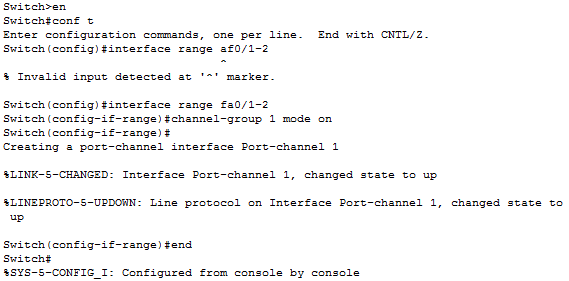


### Рисунок 14 – Схема исследуемой сети

Перед объединением 2 коммутаторов настроим порты FastEthernet 0/1 и FastEthernet 0/2, так как их будем объединять в агрегированный канал. Переходим в CLI Switch 0, заходим в режим глобального конфигурирования и редактируем оба интерфейса сразу, так как они будут содержать одинаковые настройки. Для этого используется команда interface range fa0/1-2. Определяем данные интерфейсы в channel-group 1 mode on. Cоздался интерфейс Port-channel 1. Это логический интерфейс, который объединяет два физических интерфейса. Сохраняем.



Аналогично настраиваем Switch 1.



Соединяем 2 коммутатора посредством FastEthernet 0/1 и FastEthernet 0/2. Происходит инициализации портов (рис. 14).

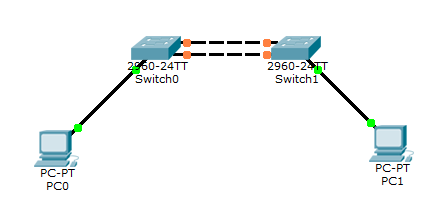
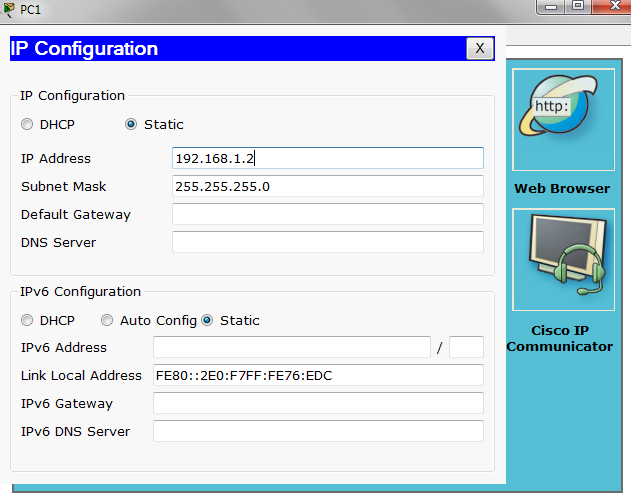
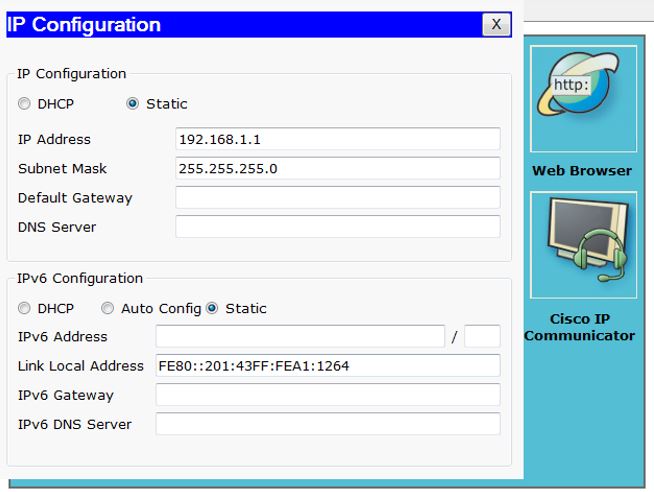


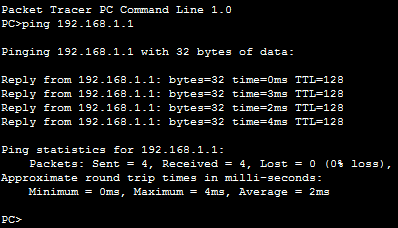
Рисунок 15 – Инициализация портов

Настраиваем IP-адресацию на компьютерах (рис. 15).



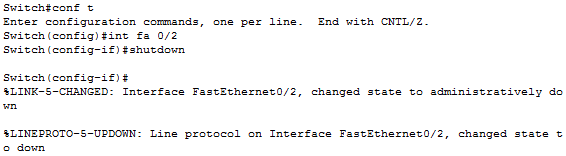
### Рисунок 16 – Настройка IP-адресации на компьютерах

Линки поднялись и оба активны. Проверяем связь командой ping. Связь работает (рис. 16).

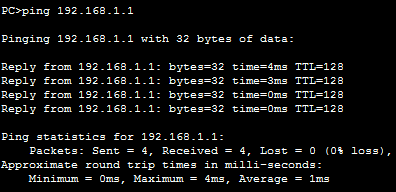


### Рисунок 17 – Проверка связи между коммутаторами

Таким образом, получили агрегированный канал между 2 коммутаторами. Канал уже не 100 мегабит, а 200 мегабит, поскольку оба «линка» являются активными. Для проверки отказоустойчивости «потушим» FastEthernet 0/2 на switch1.

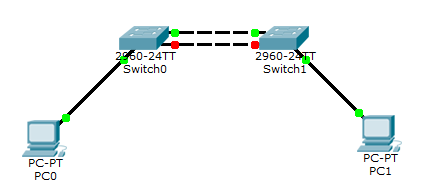


Проверим связь командой ping. Ping успешен (рис. 17).



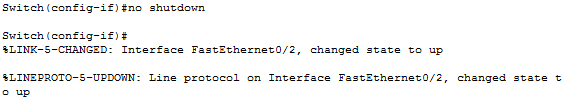
### Рисунок 18 – Проверка отказоустойчивости

Если посмотреть на схему, можно увидеть, что 1 канал до сих пор активен (рис. 18).



### Рисунок 19 – Результаты моделирования сети

Восстанавливаем работу FastEthernet 0/2 на switch1.



Связь восстановилась (рис. 19).

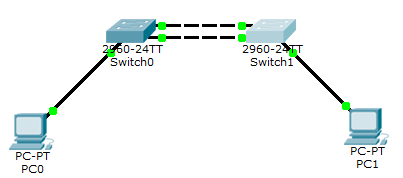
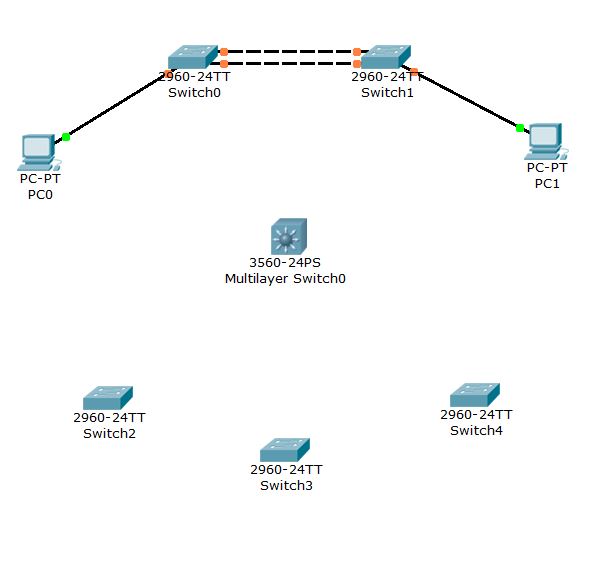


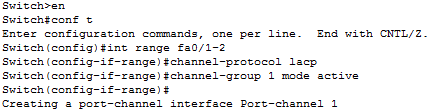
Рисунок 20 ­– Результаты моделирования сети

Добавляем switch 3560 и 3 switch 2960 (рис. 20).

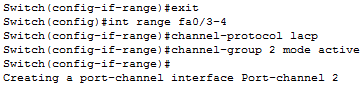


### Рисунок 21 – Схема исследуемой сети

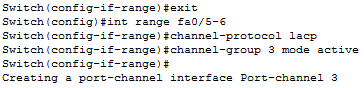
Подключаем каждый из коммутаторов 2 портами к центральному коммутатору, используя динамическое агрегирование. Переходим в CLI Switch 3560, заходим в режим глобального конфигурирования и редактируем интерфейсы, используя команду interface range fa0/1-2. Это будет первый агрегированный канал. Выбираем channel-protocol lacp и присваиваем channel-group 1 mode active. Создался интерфейс Port-channel 1. Выходим.



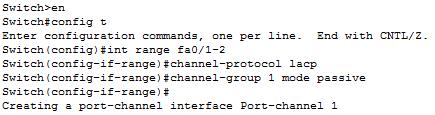
Аналогично настраиваем Port-channel 2, используя порты fast ethernet 0/ 3-4.



Аналогично настраиваем Port-channel 3, используя порты fast ethernet 0/ 5-6. Сохраняем настройки.



Переходим к настройке коммутаторов уровня доступа. Переходим в CLI коммутатора switch 2, заходим в режим глобального конфигурирования и редактируем интерфейсы, используя команду interface range fa0/1-2. Выбираем channelprotocol lacp и присваиваем channel-group 1 mode passive. Создался интерфейс Port-channel 1. Сохраняем.





Аналогичные действия производим на остальных двух коммутаторах.

Соединяем, используя тип кабеля: Copper StraightThrough (рис. 21).

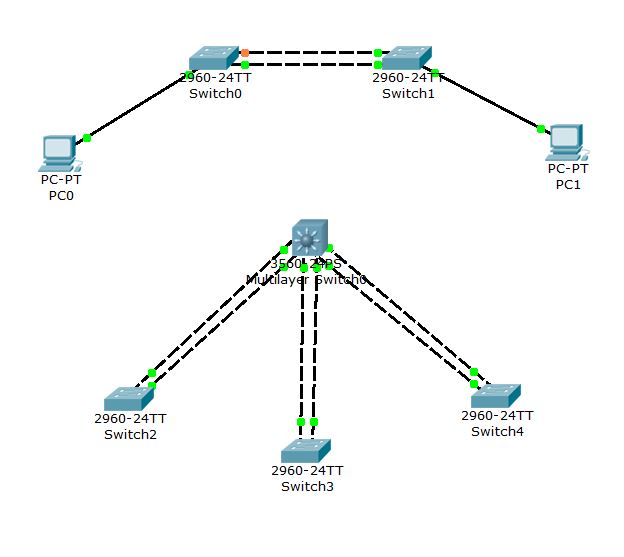


Рисунок 22 – Результаты моделирования сети

Посмотреть статус порта для 1 примера можно с помощью команды show etherchannel summary. Здесь не используется никакой протокол, настроена статическая агрегация.

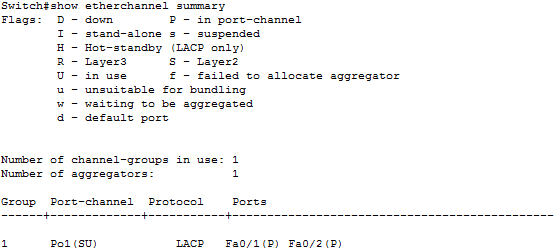


Рисунок 23 – Команда show etherchannel summary

Посмотреть статус порта для 2 примера можно с помощью команды show etherchannel summary. Здесь используется протокол lacp.

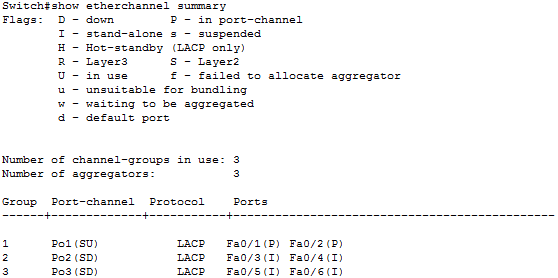


Рисунок 24 – Команда show etherchannel summary

**Контрольные вопросы**

1. Охарактеризуйте протокол STP.
2. Каков принцип действия протокола STP?
3. Охарактеризуйте проблемы, возникающие в случае отказа от применения протокола STP в локальной сети с избыточными каналами связи.
4. Назовите режимы работы портов, задействованных в STP.
5. Охарактеризуйте протокол RSTP.
6. Охарактеризуйте технологию агрегирование каналов.
7. Какие существуют методы агрегирования?
8. Охарактеризуйте протокол LACP.
9. Каковы достоинства технологии EtherChannel?
10. Каковы ограничения технологии EtherChannel?