Отказоустойчивые локальные сети.

Введение.

Существует множество технических систем, в которых не допускается прерывание связи при выходе из строя телекоммуникационного оборудования или обрывов линий связи даже на доли секунды. К таким системам относятся, в частности, системы управления производством и распределением электрической энергии, управления летательными аппаратами и ряд других. Так, например, стандарт международной электротехнической комиссии МЭК 62439 «Промышленные сети связи. Сети с высокой готовностью к автоматической обработке» предписывает, что время восстановления связи на объектах должно быть менее 10 мс. Выполнение этих жестких требований к непрерывности функционирования промышленных компьютерных сетей в таких системах обеспечивается за счет введения избыточности телекоммуникационного оборудования, в частности, многократного параллельного резервирования. Наличие резервирование требуется также и во многих корпоративных сетях, но с менее жесткими требованиями ко времени восстановления связи.

Существует несколько типов резервирования сетевого оборудования:

- **на уровне оборудования** — дублируются физические устройства, такие как маршрутизаторы, коммутаторы или сервера.
- **на уровне каналов передачи данных** — использование нескольких каналов связи для повышения устойчивости сети.
- **программное резервирование** — используется виртуализация и специальные протоколы для автоматического переключения на резервные ресурсы в случае сбоя.

Однако наличие параллельных трактов передачи пакетов, характерных при введении резервирования, может привести к появлению петель связи и возникновению зацикливания пакетов, так называемый «широковещательный шторм», что в свою очередь приведет к перегрузке сети и нарушении связи.

Для исключения петель связи в сетях с резервированием разработаны специальные протоколы резервирования, задачей которых является мониторинг дублированных каналов связи с целью недопущения зацикливания пакетов и перераспределение трафика в аварийных ситуациях. Протокол резервирования должен гарантировать логическое существование только одного пути доставки сообщения в конкретный момент времени при физическом наличии нескольких. Из существующих физических каналов связи один выбирается основным, остальные находятся в резерве.

Такой принцип был впервые применён в протоколе STP (*Spanning Tree Protoco*l), согласно предписаниям которого отслеживается состояние каналов связи и при обнаружении обрывов трафик переключается с отказавшего канала на резервный. Во время обнаружения обрыва, определения резервного пути, и переключения портов связь теряется. В зависимости от размеров сети и сложности её топологии время восстановления связи может занимать от сотен миллисекунд до десятков секунд.

Эффективность системы резервирования оценивается по **времени восстановления.** Однако стандарта или универсальной методики определения этого времени нет — производители сетевого оборудования могут указывать минимальное время, не принимая в расчёт степень загрузки сети, которая увеличивает время восстановления.

1.1. Способы резервирования в отказоустойчивых промышленных сетях

Существует две основные технологии обеспечения отказоустойчивости компьютерной сети: технология изменения топологии сети и технология «бесшовного» резервирования. Суть первой технологии заключается в изменении топологии сети в случае возникновения какой-либо неисправности в процессе функционирования сети. Изменение топологии занимает определенное время (от миллисекунд до секунд, в зависимости от протокола). Это время является основным параметром, характеризующим отказоустойчивость компьютерной сети и называется «временем восстановления». В течение этого времени связи с частью сети нет и, соответственно, данные теряются. По этой причине обеспечить время восстановления в сетях с перестройкой топологи меньше 1 мс не представляется возможным.

При «бесшовной» топологии отказоустойчивость обеспечивается не за счет перестроения топологии сети, а за счет резервирования оборудования и трактов передачи кадров. Подлежащий передаче от источника кадр дублируется отправителем, затем оба кадра передаются разными путями, а принимающий узел обрабатывает кадр, пришедший первым, и отбрасывает второй. Этот способ функционирования не требует выполнения перестроения топологии и, соответственно, данная технология не требует осуществления определенных действий на стыках фрагментов сети, обеспечивая практически «бесшовность» сети.

Существует несколько способов бесшовного резервирования телекоммуникационного оборудования и каналов связи между ними. Одним из широко распространенных способов является параллельное резервирование. В промышленных компьютерных сетях их функционирование регламентируется протоколом параллельного резервирования **PRP** (*Parallel Redundancy Protocol*). При использовании PRP строятся две независимые сети. Каждый кадр данных дублируется и одновременно передается по обеим сетям. Если до получателя доходят оба кадра, то кадр, который пришел позже, отбрасывается. Это позволяет обеспечить бесшовную передачу данных даже при полном отказе одной из сетей

Структура отказоустойчивой PRP сети изображена на рисунке 1.1. Конечный узел имеет два Ethernet-интерфейса, которые подключаются к двум изолированным друг от друга сетям LAN A и LAN B, функционирующие параллельно и имеющим независимую топологию (т.е. топологии этих двух сетей, задержки передачи кадров в них, производительности могут быть как одинаковыми, так и различаться).

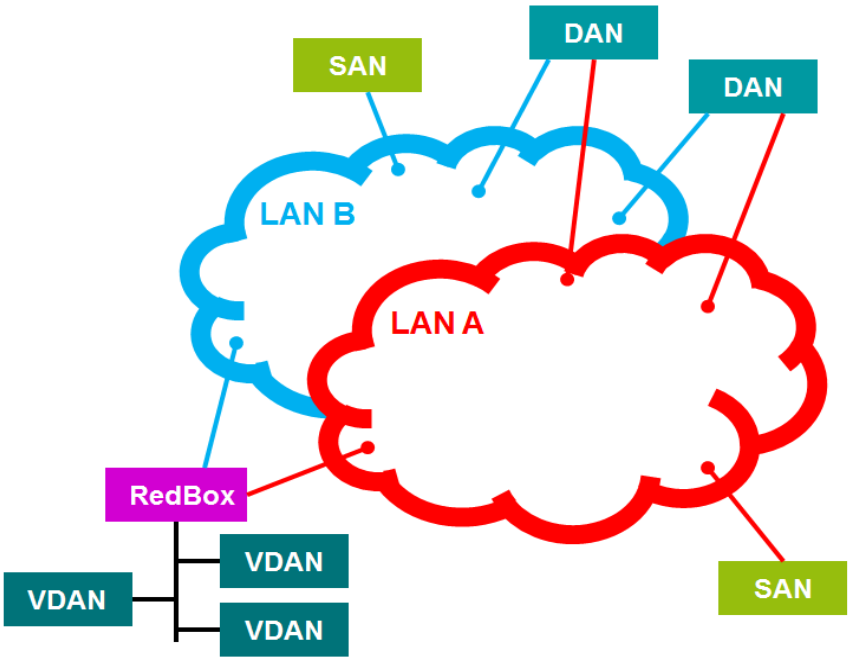


Рисунок 1.1 - Структура отказоустойчивой сети PRP

Сети должны быть изолированными для того, чтобы любая неисправность и остановка передачи данных в одной сети не влияли на вторую, т.е. даже питание сетей осуществляется от разных источников. Никаких прямых соединений между этими сетями быть не должно.

В состав такой сети может входить следующее телекоммуникационное оборудование:

- **DAN** (*Dual Attached Node*) — узел, который подключается к обеим сетям и посылает/принимает дублированные кадры.
- **SAN** (*Single Attached Node*) — узел, который подключается только к одной сети (LAN A или LAN B) и посылает/принимает обычные кадры.
- **RedBox** (*Redundancy Box*) — устройство, имеющее один входной и два выходных Ethernet-интерфейса, и. Он применяется в случае, когда к RPR-сети необходимо резервировано подключить сетевые устройства. не имеющие поддержки протокола PRP. На RedBox'е пакет от стандартного устройства дублируется и передается в сеть PRP, так словно данные передаются от DAN. Более того, устройство, которое находится за RedBox'ом, воспринимается остальными устройств как DAN. RedBox называют зачастую виртуальным DAN или VDAN (Virtual DAN).

RedBox или DAN перед отправкой данных дублируют и маркируют кадры. Маркировка осуществляется за счет добавления в конец стандартного Ethernet-кадра идентификатора RCT (*Redundancy Control Trailer*). Формат измененного кадра показан на рисунке 1.2. Идентификатор RCT состоит из следующих полей:

- номер кадра в последовательности – 16 бит;
- путь - идентификатор сети, по которой будет передаваться пакет – 4

бита;

- размер поля данных – 12 бит (данные + RCT);
- PRP суффикс – 16 бит (0x88FB)

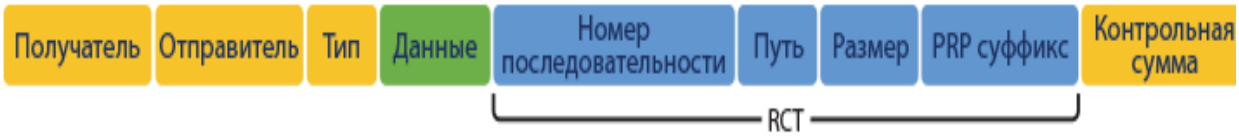


Рисунок 1.2 – Формат Ethernet-кадра с идентификатором RCT

При получении кадра RedBox или DAN анализируют номер последовательности и MAC-адрес отправителя. Если эти параметры совпадают с такими же параметрами предыдущих кадров в течение определенного времени, то кадр будет отброшен. Кадры из разных сетей будут отличаться только контрольной суммой и идентификатором сети. Схема передачи кадров между двумя устройствами DAN в отказоустойчивой сети изображена на рисунке 1.3.

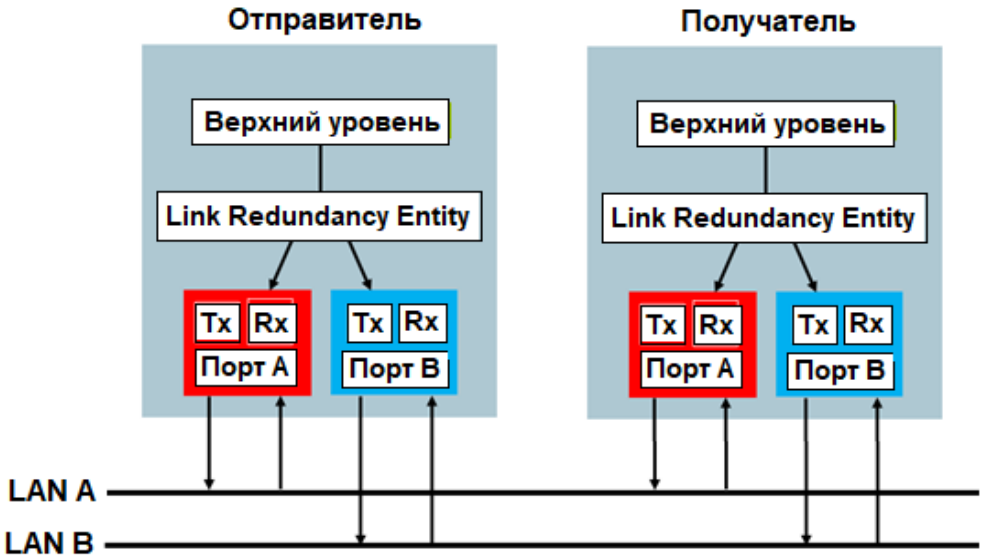


Рисунок 1.3 – Передача данных между двумя устройствами DAN в PRP

Для построения отказоустойчивых сетей с кольцевой топологией разработан протокол резервирования кольцевого соединения **HSR** (*High-availability Seamless Redundancy*). Принцип функционирования сети HSR заключается в том, что все устройства объединяются в кольцо и все сообщения, также, как и в PRP, дублируются. Кадр, поступающий от источника, дублируется и оба кадра отправляются через кольцо: одна копию по часовой стрелке, другая — против. Приемник получает обе копии, но обрабатывает только первую, а вторую удаляет. Если происходит нарушение работы одного из звеньев связи, и один из дублированных кадров не поступил, то принимается другой.

Все HSR-устройства, как и в предыдущей технологии, имеют два Ethernet-интерфейса — порт A и порт B. Структура HSR-сети изображена на рисунке 1.4. В состав сети входит оборудование, выполняющие аналогичные

функции сети PRP, но приспособленное работать по сети с кольцевой топологией.

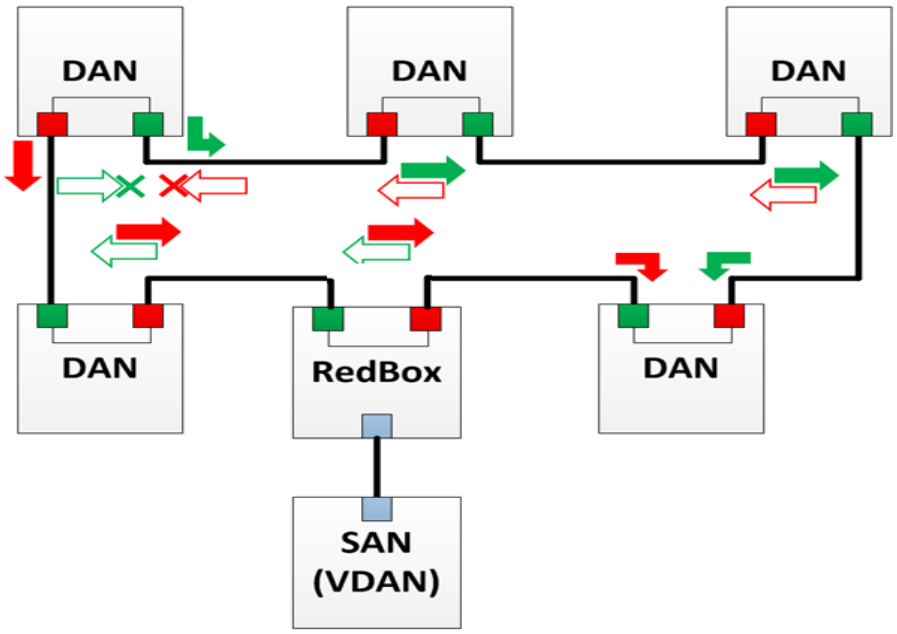


Рисунок 1.4 – Структура отказоустойчивой кольцевой сети с использованием протокола HSR

При необходимости расширения сети вводится дополнительный функциональный узел QuadBox. Это устройство имеет четыре HSR-порта и тем самым позволяет объединять два HSR-кольца. В каждом кольце QuadBox выполняет функцию DAN и может пересылать кадры из одного кольца в другое.

Способ обработки кадров у сетей HSR такой же, как и у сети PRP. Каждое HSR-устройство в кольце анализирует все поступающие на его вход кадры и копирует кадры с адресом получателя, совпадающим со своим адресом, а также широковещательные кадры. При передаче данных внутри HSR-сети к каждому кадру добавляется HSR-тег (метка). HSR-тег содержит следующие поля:

- 16-битный HSR Ethertype;
- 4-битного индикатор направления (path indicator);
- 12-битный размер кадра;
- 16-битный номер последовательности.

Отправитель вставляет одинаковые номера последовательности отправляемым и дублированным кадрам и затем инкрементирует номер последовательности для каждой посылки, отправленной с данного узла. Получатель отслеживает номера последовательности всех кадров от каждого источника, от которого он принимает данные (источники он различает по MAC-адресу). Если кадры приходят с разных линий и имеют одинаковый источник и номер последовательности, то один из них принимается, а второй отбрасывается.

В настоящее время многие современные коммутаторы зарубежных производителей поддерживают протоколы резервирования оборудования и связей

PRP и HSR. Отечественной промышленностью также выпускается ряд коммутаторов, обладающих широким спектром функциональных возможностей и поддерживающих обе технологии резервирования. Например, компания «Ангстрем Телеком» (Россия, Зеленоград) выпускает управляемые промышленные коммутаторы типа Ethernet Корунд-2о-3С5Т, Корунд-4о-3С5Т, Корунд-4о-8Т, Корунд-4о-8Т8Р, Корунд-М-2о-8Е и Корунд-М-4о-8Е другие, которые имеют расширенный L2+ функционал, высокую производительность и устойчивость к окружающей среде при малых габаритах, а также поддерживают протоколы резервирования связей MRP, HSR и PRP. Компания «Ангстрем Телеком» выпускает также и отдельные устройства резервирования RedBox типа Корунд-3С. На рисунке 1.5 показана схема подключения коммутаторов «Корунд» к двум локальным сетям А и В. Коммутаторы «Корунд» содержат встроенный RedBox, который преобразует обычные кадры Ethernet, поступающие к портам доступа, в формат PRP и направляет их по двум интерфейсам, соединенными с сетями А и В. В результате кадры от коммутаторов 1 и 2 через сети А и В доставляются к коммутатору 3, где осуществляется их обратное преобразование в обычные кадря Ethernet.

Из рассмотренных выше технологий резервирования следует. что «Бесшовное» резервирование реализуется на конечных узлах, а не на сетевых компонентах. Это одно из самых главных отличий протоколов PRP и HSR от других протоколов резервирования, таких как RSTP или MRP.

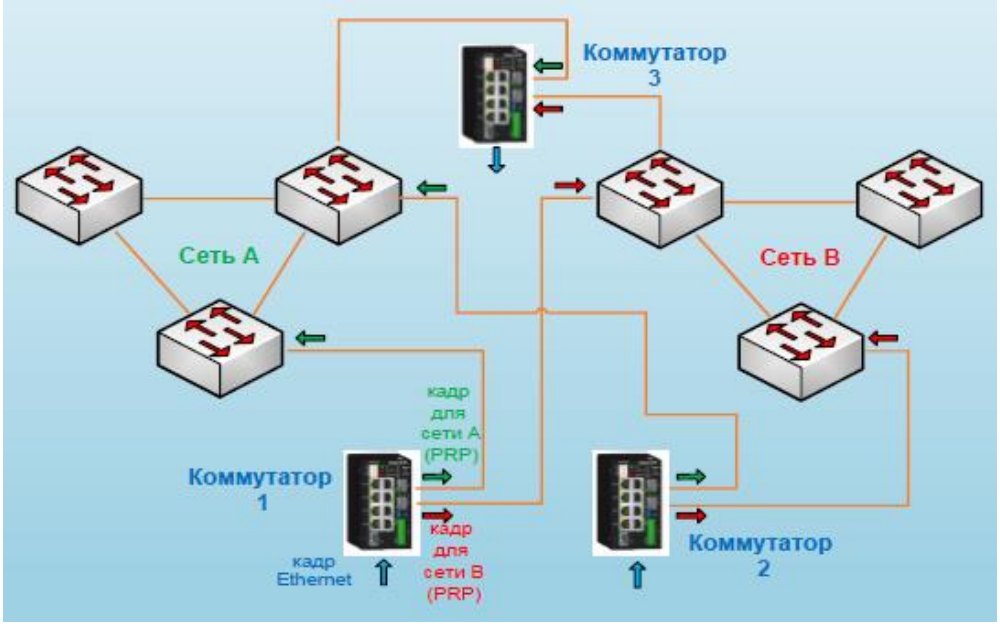


Рисунок 1.5 – Схема подключения коммутаторов «Корунд» к локальным сетям

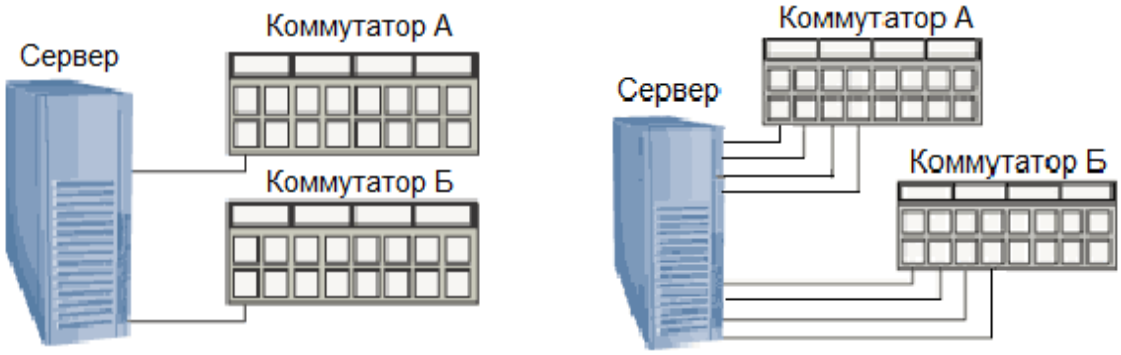
1.2. Способы резервирования в офисных сетях

Резервирование в офисных компьютерных сетях — это процесс создания избыточных элементов в сети, от резервных каналов связи до

дублирующих серверов и маршрутизаторов, служащий **для повышения надёжности и непрерывности работы сети.** Основная цель резервирования — обеспечить бесперебойное функционирование сети даже в случае отказа одного или нескольких компонентов. Отказоустойчивость компьютерной сети зависит от наличия резервирования соединений (линков) и телекоммуникационных устройств, а также от качества конфигурации (настройки параметров) оборудования.

При реализации мероприятий по осуществлению резервирования в сети следует учитывать, что каждый логический канал между активным сетевым оборудованием должен иметь как минимум два физические соединений, а каждый функциональный узел уровня ядра и распределения должен состоять из двух физических устройств (коммутаторов, маршрутизаторов или серверов).

На рисунке 1.6*а* изображена схема соединения сервера с двумя коммутаторами, которая обеспечивает отказоустойчивость в случае отказа коммутатора либо обрыва соединения. Здесь выполнено резервирование соединений и коммутаторов за счет их дублирования. В штатном режиме связь с сервером осуществляется через коммутатор А и одно звено связи. При отказе звена связи или коммутатора А осуществляется переключение на резервный коммутатор Б. При такой конфигурации для исключения появления в сети петли обязательно требуется активировать протокол STP на обоих коммутаторах. Это позволит гарантировать, что только одно соединение будет в данный момент активным. Таким образом, предотвращается ситуация зацикливания пакетов, когда они начинают циркулировать между соединениями по имеющейся петле.



а) *б*)

Рисунок 1.6 – Способы обеспечения отказоустойчивости коммутаторов а) и отказоустойчивости коммутаторов и соединений б)

На рисунке 1.6*б* показана схема соединения сервера с двумя коммутаторам с дополнительным резервированием связей между каждым из коммутаторов и сервером. Эффективность такой схемы резервирования возрастает при агрегировании линий связи между коммутаторами и сервером в один логический канал. При агрегировании линий связи передача пакетов может осуществляться параллельно по всем линкам, а в случае отказа одной или

нескольких линий передача продолжает осуществляться по остальным линиям агрегированного канала.

Для реализации процедуры агрегирования каналов требуется на коммутаторах активировать и сконфигурировать протокол **LACP (*Link Aggregation Control Protocol*).** В случае отказа коммутатора А происходит переключения трафика на резервную группу коммутатора Б.

В компьютерных сетях переключение оборудования на резервное при сбоях в работе сети обеспечивают протоколы, связанные с резервированием**. К ним относятся** STP (*Spanning Tree Protocol*)**,** RSTP (*Rapid Spanning Tree Protocol*)**,** VRRP (*Virtual Router Redundancy Protocol*) **и** HSRP (*Hot Standby Router Protocol*)**.** Эти протоколы обеспечивают автоматическое переключение на резервные маршрутизаторы или каналы в случае выхода из строя основного.

Для обеспечения отказоустойчивости сети при сбоях или выходе сервера из строя производят объединение нескольких серверов в одну функциональную группу (кластер). Если один из серверов выходит из строя, его функции автоматически перераспределяются между другими серверами кластера, что предотвращает простои. Эта стратегия особенно эффективна для критически важных приложений и сервисов, таких как базы данных или веб-сайты с высокой нагрузкой. Кластеры могут быть активными (где все серверы работают одновременно) или пассивными (где резервные серверы включаются только в случае отказа).

С целью повышения надежности функционирования сети при возникновении проблем у Интернет-провайдера применяется подключение к двум провайдерам (рисунок 1.7). В ряде случаев целесообразно организовывать связь с Интенет-провайдером по проводному и беспроводному каналам.

Рисунок 1.7 – Схема повышения отказоустойчивости сети за счет использования двух провайдеров

Упрощенная топология отказоустойчивой корпоративной сети изображена на рисунке 1.8. Резервирование оборудования и линий связи выполнено на уровне распределения. В сети используется подключение к двум Интернет-провайдерам.

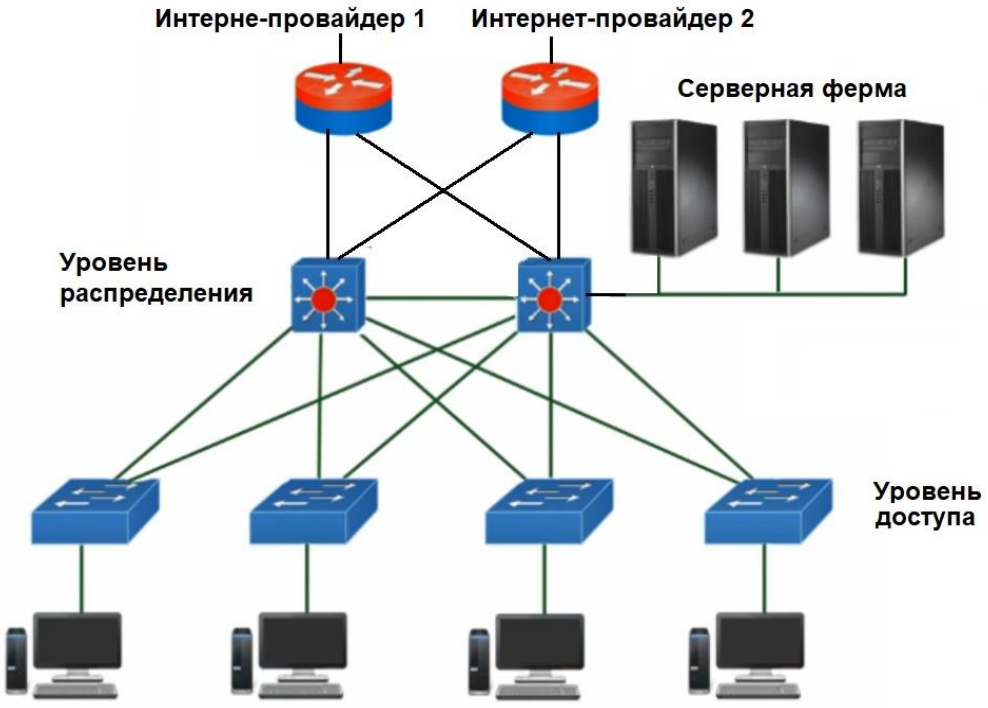


Рисунок 1.8 – Топология отказоустойчивой корпоративной сети

Перспективным направлением в области повышения отказоустойчивости сети является использование облачных технологий хранения информации. Эта технология обладает рядом встроенных механизмов для резервирования, балансировки нагрузки и репликации данных. Облачные провайдеры, такие как AWS, Microsoft Azure или Google Cloud, предлагают сервисы, которые автоматически распределяют данные и вычислительные ресурсы по нескольким зонам доступности, что защищает от сбоев в одном из центров обработки данных.

1.3. Протоколы связующего дерева STP и RSTP

В сетях Ethernet коммутаторы поддерживают только древовидные связи т.е. которые не содержат петель. При построении или модернизации сегментированной сети с большим количеством коммутаторов в результате ошибок монтажа или попыток резервирования соединений возможно образование дополнительных связей между сегментами, когда от одного сегмента к другому пакет может попасть более чем одним путем (рисунок 1.9). Это приведет к циркуляции пакетов в замкнутых петлях и перегрузке сети. Кроме того, каждый посланный пакет, поступающий через разные порты, коммутаторы принимают за два различных пакета и постоянно обновляют свои таблицы. В приведенном примере проблема может быть решена удалением моста 1 и разрывом связи, помеченной знаком “X”.

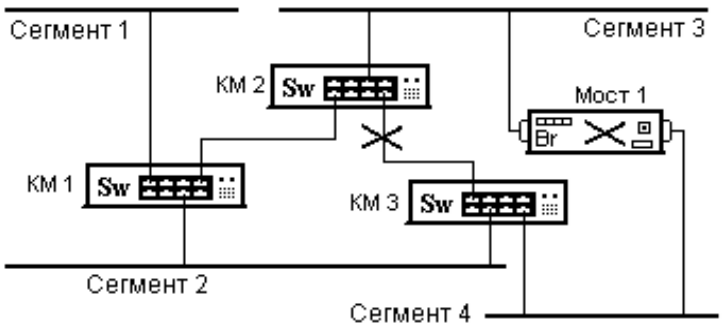


Рисунок 1.9 – Пример иллюстрации наличия петель в сети

Для автоматического решения проблемы зацикливания пакетов был разработан так называемый "алгоритм покрывающего дерева". Алгоритм покрывающего дерева **STA** (*Spanning Tree Algotithm*) обеспечивает построение древовидной топологии связей сети с единственным путем минимальной стоимости от каждого коммутатора и от каждого сегмента (персонального компьютера) до некоторого выделенного корневого коммутатора – **корня дерева**. Реализация алгоритма осуществляется на основе протокола **STP** (*Spanning Tree Protocol*), в результате действия которого коммутатор самостоятельно обнаруживает лишние связи и автоматически блокирует ряд соединений, приведших к образованию петель. В случае возникновения аварийных ситуаций и недоступности основного пути, заблокированные соединения могут быть вновь открыты. Этим обеспечивается высокая надежность сети. STP входит в состав протокола мостов и коммутаторов IEEE 802.1d. На время разработки этого алгоритма для сегментации локальных сетей широко использовались телекоммуникационные мосты (*Bridges*), функционировавшие на втором уровне эталонной модели. Затем мосты были заменены на коммутаторы, которые, собственно, являются многопортовыми мостами, но термин «мост» существует и в настоящее время в описании протокола STP.

При использовании протокола STP при начальной конфигурации каждой линии связи присваивается определенный вес (чем выше приоритет, тем меньше вес). Коммутаторы периодически рассылают специальные сообщения – **протокольные блоки данных моста** BPDU (*Bridge Protocol Data Unit*), которые содержат коды уникальных идентификаторов коммутаторов, присвоенных им при изготовлении. Коммутатор с наименьшим значением такого кода становится корневым ("корнем дерева" – *Root Bridge*), т.е. центральной точкой для всех коммутаторов в сети, а все дерево STP сходится к нему. Затем выявляются наикратчайшие расстояние от корневого коммутатора до любого другого коммутатора в сети. В основу алгоритма STA положена теорема из теории графов, которая утверждает, что *структура любого связного графа, содержащего петли, может быть изменена путем удаления ребер таким образом, что он сохранит прежнюю связность, и при этом не будет иметь петель.* Граф, описывающий дерево наикратчайших связей, и является "**покрывающим деревом**". Алгоритм STA функционирует по умолчанию постоянно на

современных коммутаторах, отслеживая все топологические изменения. Реализация этого алгоритма в компьютерной сети возможна только при условии поддержки его всеми коммутаторами/мостами, входящими в эту сеть.

Коммутаторы, поддерживающие алгоритм STA, автоматически создают активную древовидную конфигурацию связей, то есть связную конфигурацию без петель. Древовидная структура сети строится на основании информации, полученной в результате обмена между всеми коммутаторами служебными пакетами, и адаптивно перестраивается при возникновении изменений в сети. В качестве служебных пакетов, содержащих сведения о текущей топологии и состоянии узлов и связей в протоколе STP, применяются протокольные блоки данных BPDU (**Bridge Protocol Data Unit)**, которые размещаются в информационном поле блоков данных канального уровня – кадров сетей Ethernet или Token Ring. Сообщения BPDU представляют собой пакеты второго уровня эталонной модели OSI, MAC-адрес назначения которых является групповым адресом 01-80-C2-00-00-00. Все коммутаторы, поддерживающие STP, обязаны принимать и обрабатывают полученные BPDU.

Формат пакета BPDU содержит следующие идентификаторы.

1. "Идентификатор протокола" (*Protocol identifier*) – 2 байта. Для протокола STP это поле равно 0x0000.
2. "Версия протокола" (*Version*) длиной 1 байт. Для STP всегда 0x00, 0x02 – протокол RSTP и 0x03 – протокол MSTP.
3. "Тип сообщения" (*Message type*) размером 1 байт. Сообщение может быть конфигурационным (тип 0x00) и извещающим о смене топологии (0x80).
4. "Флаги" (Flags) – 1 байт. Используются только два бита этого поля. Первый из них, TC-бит, сигнализирует о смене топологии (*Topology Change*), а восьмой, TCA-бит, подтверждает изменение конфигурации (*Topology Change Acknowledgment*).
5. "Идентификатор корневого коммутатора " (*Root ID*) состоит из 8 байтов. Первые два байта отображают **приоритет** данного устройства (*Bridge Priority*), который по умолчанию равен 32768 на всех коммутаторах (1 в старшем разряде) и может быть установлен системным администратором. Последующие 6 байтов представляют собой MAC-адрес виртуальной сети Vlan 1, являющегося адресом блока управления корневого моста. В версиях STP, которые поддерживают работу с VLAN на Bridge Priority выделено 4 бита, а остальные 12 бит предназначены для идентификатора виртуальных сетей (Extended System ID – VLAN ID). Формат идентификатора корневого коммутатора показан на рисунке 1.10.

Bridge ID													
Bridge Priority				Extended System ID (VLAN ID)									
32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Рисунок 1.10 ─ Структура идентификатора Bridge Priority

Поле Extended System ID при отсутствии деления сети на VLAN всегда, как правило, принимает нулевое значение.

1. "Стоимость пути до корня" (*Root path cost*). Поле длиной 4 байта отображает суммарную стоимость кратчайшего пути от коммутатора-источника сообщения до корневого коммутатора.
2. "Идентификатор моста/коммутатора" (*Bridge ID*) – 8 байтов. Два старших байта определяют приоритет коммутатора (значение по умолчанию равно 32 768). Он может быть задан вручную, что позволяет администратору сети влиять на процесс выбора корневого коммутатора. Шесть младших байтов занимает MAC-адрес VLAN.
3. "Идентификатор порта" (*Port ID*) указывает порт, через который было передано данное сообщение. Длина его составляет 2 байта, старший из которых может изменяться администратором и является приоритетом порта, а второй представляет собой порядковый номер порта для данного коммутатора (номера портов начинаются с единицы).
4. "Возраст сообщения" (*Message age*). Поле занимает 2 байта и предназначено для указания времени существования сообщения в секундах об изменении топологии в сети. Каждый коммутатор при прохождении через него сообщения модифицирует содержимое этого поля путем добавления значения задержки, соответствующей данному каналу.
5. "Максимальный возраст сообщения" (*Maximum age*) – время, по истечении которого сообщение BPDU игнорируется коммутатором. По умолчанию – это 20 секунд (10**Hello Time*). Если кадр BPDU имеет время жизни, превышающее максимальное, то кадр игнорируется коммутаторами. Таймер при получении BPDU сбрасывается и устанавливается граничное значение 20 секунд по умолчанию. Если текущий Hello BPDU устарел, но новые BPDU не приходят (таймер не обнуляется), то коммутатор восстанавливает резервную линию.
6. "Интервал Hello" (*Hello time*) – промежуток времени между отправкой конфигурационных сообщений корневым коммутатором (1…4 с), по умолчанию этот интервал установлен равным 2 с.
7. "Задержка перехода" (*Forward delay*). Значение этого поля определяет величину интервала времени в секундах, который должен предшествовать переходу коммутатора в новое состояние при изменении топологии системы (т.е. продолжительности состояний «Прослушивание» и «Обучение»). Эта задержка предназначена для исключения возникновения циклических маршрутов во время переходных процессов в сети.

Три последних поля BPDU-блока занимают по 2 байта каждый.

Построение сети без петель по протоколу STP осуществляется следующим образом. Каждый из сетевых коммутаторов путем выдачи на все свои порты BPDU-блоков анонсируют себя в качестве корневого, помещая свой идентификатор в полях "Идентификатор корневого коммутатора" и "Идентификатор коммутатора". Затем для каждого сетевого коммутатора из всех

портов данного коммутатора определяется ***корневой порт (root port)***, т.е. такой порт, путь от которого до любого из портов корневого коммутатора имеет минимальную стоимость. Для этого корневой коммутатор рассылает BPDU-блоки на все свои выходные порты. В поле "Стоимость пути до корня" вначале устанавливается нулевое значение. Следующие коммутаторы прибавляют к этому полю стоимость своих портов и отправляют блоки далее смежным узлам. Это дает возможность каждому коммутатору определить свой корневой порт, через который можно попасть в корневой коммутатор с минимальной стоимостью. Как только коммутатор получает BPDU-блок, содержащий идентификатор корневого коммутатора со значением меньше его собственного, он перестает генерировать свои собственные кадры BPDU, а начинает ретранслировать только кадры нового претендента на статус корневого коммутатора.

В процессе ретрансляции кадров каждый коммутатор увеличивает указанную в пришедшем блоке BPDU стоимость пути до корня дерева на величину стоимости интерфейса (*Segment cost*), через который поступил данный блок. Тем самим в кадре BPDU, по мере прохождения через коммутаторы, аккумулируется стоимость пути до корневого узла. В течение этой процедуры каждый коммутатор для каждого из своих портов запоминает параметры путей минимальной стоимости до корня, содержащиеся во всех принятых этим портом кадрах BPDU.

На следующем этапе функционирования алгоритма для каждого логического сегмента сети из всех портов всех коммутаторов, подсоединенных к данному сегменту, выбирается порт, через который будут передаваться пакеты от этого сегмента в направлении **корня** через **корневой порт** одного из коммутаторов. Для этого сначала из рассмотрения исключаются корневые порты коммутаторов, подключенных к данному сегменту. Затем из всех оставшихся портов выбирается порт с минимальной стоимостью пути до корня. Этот порт называется "**назначенный порт"** (*designated port*), а коммутатор, которому он принадлежит, получил название **назначенный коммутатор** *(designated switch).* Если в данном коммутаторе имеется несколько портов с одинаковой стоимостью, то назначенным становится порт, имеющий минимальный идентификатор.

Все остальные порты, кроме корневых и назначенных, отключаются и переводятся в резервное состояние, то есть такое, при котором они не передают обычные кадры данных. При таком выборе активных портов в сети исключаются петли и оставшиеся связи образуют покрывающее дерево. У корневого коммутатора все порты являются назначенными, а их стоимости до корня полагаются равными нулю. Корневой порт у корневого коммутатора отсутствует.

В процессе нормальной работы корневой коммутатор продолжает генерировать служебные пакеты BPDU, а остальные коммутаторы принимают их своими корневыми портами и ретранслируют через назначенные порты. Один или несколько портов коммутаторов находятся в состоянии **Blocking**. В этом состоянии порт слушает BPDU и не предпринимает никаких действий. Но если вдруг произойдет изменение топологии, то по истечении максимального

времени жизни сообщения (по умолчанию — 20 с) корневой порт любого коммутатора сети не получит служебный пакет BPDU, то он инициализирует новую процедуру построения покрывающего дерева.

Порт, находившийся в состоянии Blocking, сразу переходит в состояние **Listening** (Прослушивание). В этом состоянии он отправляет, слушает только BPDU кадры и обрабатывает полученную информацию. Если он обнаруживает, что у соседей параметры хуже, чем у него, то по истечении 15 секунд, переходит в следующее состояние **Learning** (Обучение). Эта фаза длится также 15 секунд. В состоянии Learning порт делает практически все тоже самое, что и в предыдущем состоянии, за исключением того, что теперь начинается перестроение таблицы коммутации на основании полученных кадров. Если по истечении 15 секунд, он не получит BPDU с параметрами лучше, чем у него, то перейдет в последнее состояние **Forwarding** (Продвижение). В этом состоянии порт обменивается не только служебной информацией, но и пользовательскими данными. То есть переход из состояния **Listening** в **Forwarding** длится 30 секунд.

Если по логике проекта корневым коммутатором должен быть конкретный коммутатор, то сетевой администратор вручную указывает корневой коммутатор путем задания ему нулевого (самого высокого) уровня приоритета, т.е. значение двух старших байт его идентификатора устанавливается равным нулю. Приоритет других коммутаторов должен быть отличным от нуля. В этом случае, не зависимо от значения MAC-адресов коммутаторов сети, корневой будет всегда иметь минимальное значение идентификатора. Кроме этого, администратор должен задать стоимость портов каждого из коммутаторов. Стоимость порта (*Port Cost*) определяется как условное время передачи бита через данный порт. На практике стоимость порта часто вычисляют по формуле:

Стоимость порта = 1000 / (*скорость передачи порта*, Мбит/с).

Например, стоимость порта 10Base-T=100; 100Base-TX =10; Token Ring –250 или 63, канала T1 = 651 и т.д. Значения другого варианта задания стоимостей, регламентированные стандартом IEEE 802.1d, приведены в таблице 1.1.

Таблица 1.1 – Стоимость порта в зависимости от скорости передачи

Скорость передачи, Мбит/с	4	10	16	45	100	155	622	1000	10000
Стоимость порта	250	100	62	39	19	14	6	4	2

В STP существуют следующие состояния портов:

- **Disabled** – отключенный порт. Такой порт не передает и не принимает никаких сообщений: ни пользовательский трафик, ни блоки BPDU и не проводит анализ MAC-адресов.

- **Blocking** – блокирование. Это состояние предназначено для предотвращения петель в сети. Не смотря на заблокированное состояние порт остается активным, т.е. принимает блоки BPDU (но сам их не отправляет), так как ему нужно быть готовым перейти в другое состояние в случае работы алгоритма STA. MAC-адреса прослушиваемых кадров он не анализирует и кадры с данными также не перенаправляет.
- **Listening** – прослушивание. Порт слушает и начинает сам отправлять BPDU (только Designated порт). MAC-адреса порт не изучает и кадры с данными не перенаправляет;
- **Learning** – обучение. Порт слушает и отправляет BPDU (отправляет только Designated порт), а также вносит изменения в таблицу коммутации (по результатам изучения MAC-адресов, по трафику, который приходит на порт). Данные не перенаправляет.
- **Forwarding** – пересылка. Порт слушает/отправляет BPDU (отправляет только Designated порт), участвует в поддержании таблицы MAC-адресов и перенаправляет данные. То есть это обычное состояние рабочего порта.

Состояния Blocking и Forwarding стабильные, а Listening и Learning – промежуточные. При включении коммутатора с STP (или при подключении нового патч-корда) все порты проходят вышеприведенные состояния согласно алгоритму STA. Смена состояний портов происходит с некоторой задержкой, которая задается значением параметра **Forward Delay.** Этот параметр определяет время, как долго Root или Designated порты коммутатора будут оставаться в состоянии Listening и Learning (по умолчанию 15 секунд). Forward Delay – это время, за которое информация, содержащаяся в блоке BPDU гарантированно будет передана от любого коммутатора сети с данной топологией до любого другого.

В протоколе связующего дерева STP длительность состояния блокировки составляет 20 секунд, состояние прослушивания — 15 с, а состояние обучения — 15 с. Таким образом, для STP время прохождения состояний портов до состояния пересылки занимает 50 секунд.

Развитием стандарта 802.1d STP стал стандарт IEEE 802.1w – протокол *Rapid Spanning Tree Protocol* (**RSTP**). Он был разработан для преодоления отдельных ограничений STP, которые мешали внедрению ряда новых функций коммутаторов, в частности, функций 3-го уровня, всё больше и больше применяемых в коммутаторах Ethernet. Процесс вычисления связующего дерева у обоих протоколов одинаков. Однако при работе **RSTP**, **порт может перейти в состояние передачи значительно быстрее**, так как он не зависит от настройки таймеров. Порты больше не должны ждать стабилизации топологии, чтобы перейти в режим продвижения.

RSTP обеспечивает более быструю сходимость (конвергенцию) за счет введения новых ролей и состояний портов, что сокращает время конвергенции с нескольких секунд до 5-10 миллисекунд. RSTP также поддерживает такие

функции, как корректировка стоимости портов, определение типа канала и механизм предложения/согласия для более быстрой конвергенции.

Кроме корневого и назначенного порта, аналогичных протоколу STP, в протоколе RSTP в коммутаторах дополнительно введены **альтернативный** (*Alternate*) корневой порт и р**езервный (Backup)** назначенный порт. Альтернативный и резервный порты не участвуют в пересылке данных, пока не произойдет обрыв связи. Коммутаторы блокирует передачу данных по этим портам во избежание образования петель. В случае обрыва основного канала связи коммутатор начинает передавать данные по резервному пути (по Alternate и Backup портам). Следует заметить, что резервный порт является резервным к общей среде, в такой как концентратор (hub). Поэтому резервный порт в современных сетях практически не используется.

Вместо состояния «Блокировка» и «Прослушивание» в RSTB введено состояние «Отбрасывание» (***Discarding***). В этом состоянии пользовательские данные через порт не передаются. В RSTP коммутаторы обмениваются данными напрямую с соседними коммутаторами, что обеспечивает быструю синхронизацию в топологии. Это позволяет портам быстро переходить из состояния отбрасывания в состояние пересылки без использования таймера задержки.

Быстродействие протокола RSTP повышено за счёт использования механизмов, не привязанных к стандартным таймерам, по сравнению с классическим протоколом STP. Это позволяет портам быстрее переходить в состояние пересылки (Forwarding) в случае сбоя, что сокращает время сходимости сети.

1.4. Протоколы связующего дерева для виртуальных подсетей

Недостатком протоколов STP и RSTP для компьютерной сети, разделенной на виртуальные подсети VLAN, является невозможность управления трафиком VLAN через резервные каналы: если канал заблокирован, он блокируется для всех VLAN. Поэтому, если компьютерная сеть разделена на VLAN, то целесообразно построения покрывающего дерева выполнить для каждой виртуальной сети в отдельности. Обычные протоколы STP и RSTP справиться с этой задачей не могут. Поэтому компания Cisco разработала проприетарный протокол **PVST и PVST+ (*Per*-VLAN *Spanning Tree Plus*),** в котором используется по одному **экземпляру STP на каждую из VLAN**. Каждая виртуальная локальная сеть может использовать собственный корневой коммутатор и топологию пересылки, что обеспечивает более справедливое распределение ресурсов. В PVST для каждой VLAN существует свой процесс STP, что позволяет независимую и гибкую настройку под потребности каждой виртуальной сети, а также использовать балансировку нагрузки за счет того, что конкретная физическая связь может быть заблокирована в одной VLAN, но работать в другой.

Протокол PVST работает только с магистралями внутреннего стандарта Cisco ISL, а PVST+ — с магистралями международного стандарта 802.1Q (следовательно, совместим с оборудованием других производителей). Этот протокол обеспечивает оптимальную управляемость трафиком и резервирование для VLAN в коммутационной среде. Создавая уникальные структуры дерева для каждой VLAN, администраторы могут лучше контролировать поток трафика и оптимизировать производительность сети.

Для повышения эффективности процесса обнаружения и исключения образования петель, возникающих как из-за технических ошибок при конфигурации оборудования, так и по причине действия злоумышленников, в современных коммутаторах введены дополнительные (опциональные) настройки протоколов STP и RSTP. К ним относятся следующие функции:

PortFast; BPDU Guard; Root Guard; Loop Guard; BPDU Filter.

Опция **PortFast** предназначена для минимизации времени ожидания портами доступа схождения протокола spanning-tree. Если порт коммутатора настроен с помощью функции PortFast, то такой порт сразу переходит из состояния блокировки в состояние пересылки, минуя стандартные состояния перехода STP 802.1D (состояния прослушивания и получения данных).

Вместо того, чтобы ожидать схождения протокола STP IEEE 802.1D в каждой сети VLAN, PortFast целесообразно использовать только на портах доступа для обеспечения немедленного подключения этих устройств к сети. При включении на всех портах доступа функции PortFast, это гарантирует невозможность случайного создания петли при добавлении неавторизованного коммутатора в топологию. Если же функция PortFast включена на порту, подключенном к другому коммутатору, то существует риск возникновения петли протокола spanning-tree.

Для предотвращения возникновения такого типа сценариев, в современные коммутаторы введена функция **BPDU Guard**. Когда функция BPDU Guard включена, то при получении блока BPDU она переводит порт в состояние errdisabled (error-disabled — отключение из-за ошибки). Это защищает от потенциальных петель, путем отключения порта.

Настройка Port Fast на интерфейсе доступа выполняется следующим образом:

```
sw(config)#interface fa0/1
sw(config-if)# spanning-tree portfast
```

Если интерфейс настроен как магистральный (транковый), то в команду следует добавлять параметр *trunk:*

```
sw(config)#interface fa0/1
sw(config-if)# spanning-tree portfast trunk
```

Функция **Loop Guard** обеспечивает дополнительную защиту на 2 уровне от возникновения петель, возникающих в результате сбоя телекоммуникационного оборудования.

При активации на интерфейсе опции **BPDU Filter**, интерфейс будет игнорировать входящие BPDU пакеты и не будет отправлять никаких BPDU. Его

следует включать только на интерфейсах уровня доступа. Настройка на интерфейсе BPDU Filtering осуществляется следующей командой

```
sw(config)#interface fa0/1
sw(config-if)# spanning-tree bpdufilter enable
```

К недостаткам PVST относится то, что с увеличением количества виртуальных локальных сетей протокол PVST не мог обеспечить эффективное использование ресурсов коммутаторов, кроме этого, усложнялась процедура управления коммутаторами. Это связано с тем, что количество различных логических топологий на практике существенно меньше, чем количество активных виртуальных локальных подсетей. Для устранения указанных недостатков был разработан протокол **MSTP** (*Multiple Spanning Trees Protocol*, «протокол множественных покрывающих деревьев»), описанный в стандарте IEEE 802.1s. Отличительная особенность протокола MSTP состоит в том, что избыточная физическая топология имеет лишь небольшое количество различных связующих деревьев (логических топологий).

Вместо запуска экземпляра STP для каждой VLAN, протоколом MSTP предусмотрено создание несколько независимых от VLAN экземпляров (инстанций) STP (представляющих логические топологии), в которых передаются собственные блоки BPDU независимо друг от друга. Задача администратора состоит в сопоставлении каждой VLAN с наиболее подходящей логической топологией (экземпляром STP). Количество экземпляров STP сводится к минимуму (экономия ресурсов коммутатора), но пропускная способность сети обеспечивается более рациональным образом, за счет использования всех возможных путей для трафика VLAN. Очевидно, что для реализации этого механизма на каждом коммутаторе должна быть своя таблица сопоставления VLAN с номерами экземпляров STP.

Протоколом MSTP предусмотрено создание до 64 инстанций (экземпляров) STP на одной физической сети ─ M*ultiple Spanning-Tree Instances* (*MSTI*)., что позволяет оптимизировать маршруты для различных VLAN.

MSTP позволяет также создать логическое группирование коммутаторов в управляемые кластеры, называемые регионами ─ **Multiple Spanning Tree (MST) region**. Каждый MST region поддерживает до 64-х инстанций *MSTI*. Технология MSTP значительно уменьшает количество пакетов BPDU в сети путём включения STP-информации для всех MSTI в одну BPDU. Конфигурационное сообщение MSTI передают STP-информацию для каждой инстанции *MSTI*.

Протокол MSTP в системе Packet Tracer **не реализован.**