ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «ОРЕНБУРГСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ»

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Б1.Б.33 Сетевые технологии

Направление подготовки (специальность) 10.03.01 - «Информационная безопасность»

Профиль образовательной программы «Безопасность автоматизированных систем»

Форма обучения *очная*

Содержание:

КОНСПЕКТ ЛЕКЦИЙ

1.1Лекция №1**(2 часа).**

Тема: «Основы коммутации»

1.1.1 Вопросы лекции:

1. Эволюция локальных сетей.
2. Функционирование коммутаторов локальных сетей.

1.1.2 Краткое содержание вопросов:

1. Эволюция локальных сетей.

Эволюция локальных сетей неразрывно связана с историей развития технологии Ethernet, которая по сей день остается самой распространенной технологией локальных сетей.

Первоначально технология локальных сетей рассматривалась как времясберегающая и экономичная технология, обеспечивающая совместное использование данных, дискового пространства и дорогостоящих периферийных устройств. Снижение стоимости персональных компьютеров и периферии привело к их широкому распространению в бизнесе, и количество сетевых пользователей резко возросло. Одновременно изменились архитектура приложений ("клиент-сервер") и их требования к вычислительным ресурсам, а также архитектура вычислений (распределенные вычисления). Стал популярным downsizing (разукрупнение) — перенос информационных систем и приложений с мэйнфреймов на сетевые платформы. Все это привело к смещению акцентов в использовании сетей: они стали обязательным инструментом в бизнесе, обеспечив наиболее эффективную обработку информации.

В первых сетях Ethernet (10Base-2 и 10Base-5) использовалась шинная топология, когда каждый компьютер соединялся с другими устройствами с помощью единого коаксиального кабеля, используемого в качестве среды передачи данных. Сетевая среда была разделяемой и устройства, прежде чем начать передавать пакеты данных, должны были убедиться, что она свободна. Несмотря на то, что такие сети были простыми в установке, они обладали существенными недостатками, заключающимися в ограничениях поразмеру, функциональности и расширяемости, недостаточной надежности, а также неспособностью справляться с экспоненциальным увеличением сетевого трафика. Для повышения эффективности работы локальных сетей требовались новые решения.

Следующим шагом стала разработка стандарта 10Base-T с топологией типа "звезда", в которой каждый узел подключался отдельным кабелем к центральному устройству — концентратору (hub). Концентратор работал на физическом уровне модели OSI и повторял сигналы, поступавшие с одного из его портов на все остальные активные порты, предварительно восстанавливая их. Использование концентраторов позволило повысить надежность сети, т.к. обрыв какого-нибудь кабеля не влек за собой сбой в работе всей сети. Однако, несмотря на то, что использование концентраторов в сети упростило задачи ее управления и сопровождения, среда передачи оставалась разделяемой (все устройства находились в одном домене коллизий). Помимо этого, общее количество концентраторов и соединяемых ими сегментов сети было ограничено из-за временных задержек и других причин.

С появлением стандарта IEEE 802.3af-2003 PoE, описывающего технологию передачи питания по Ethernet (Power over Ethernet, PoE), разработчики начали выпускать коммутаторы с поддержкой данной технологии, что позволило использовать их в качестве питающих устройств для IP-телефонов, Интернет-камер, беспроводных точек доступа и другого оборудования.

С ростом популярности технологий беспроводного доступа в корпоративных сетях производители оборудования выпустили на рынок унифицированные коммутаторы с поддержкой технологии PoE для питания подключаемых к их портам точек беспроводного доступа и централизованного управления как проводной, так и беспроводной сетью.

Повышение потребностей заказчиков и тенденции рынка стимулируют разработчиков коммутаторов более или менее регулярно расширять аппаратные и функциональные возможности производимых устройств, позволяющие предоставлять в локальных сетях новые услуги, повышать их надежность, управляемость и защищенность.

2. Функционирование коммутаторов локальной сети.

Коммутаторы локальных сетей обрабатывают кадры на основе алгоритма прозрачного моста (transparentbridge), который определен стандартом IEEE 802.1D. Процесс работы алгоритма прозрачного моста начинается с построения таблицы коммутации (ForwardingDataBase, FDB).

Изначально таблица коммутации пуста. При включении питания, одновременно с передачей данных, коммутатор начинает изучать расположение подключенных к нему сетевых устройств путем анализа MAC-адресов источников получаемых кадров. Записи в таблице коммутации создаются динамически. Это означает, что, как только коммутатором будет прочитан новый MAC-адрес, то он сразу будет занесен в таблицу коммутации. Дополнительно к MAC-адресу и ассоциированному с ним порту в таблицу коммутации для каждой записи заносится время старения (agingtime). Время старения позволяет коммутатору автоматически реагировать на перемещение, добавление или удаление сетевых устройств. Каждый раз, когда идет обращение по какому-либо MAC-адресу, соответствующая запись получает новое время старения. Записи, к которым не обращались долгое время, из таблицы удаляются. Это позволяет хранить в таблице коммутации только актуальные MAC-адреса, что уменьшает время поиска соответствующей записи в ней и гарантирует, что она не будет использовать слишком много системной памяти.

Помимо динамического создания записей в таблице коммутации в процессе самообучения коммутатора, существует возможность создания статических записей таблицы коммутации вручную. Статическим записям, в отличие отдинамических, не присваивается время старения, поэтому время их жизни не ограничено.

Статическую таблицу коммутации удобно использовать с целью повышения сетевой безопасности, когда необходимо гарантировать, что только устройства с определенными MAC-адресами могут подключаться к сети. В этом случае необходимо отключить автоизучение MAC-адресов на портах коммутатора.

Как только в таблице коммутации появляется хотя бы одна запись, коммутатор начинает использовать ее для пересылки кадров.

Когда коммутатор получает кадр, отправленный компьютером A компьютеру B, он извлекает из него MAC-адрес приемника и ищет этот MAC-адрес в своей таблице

коммутации. Как только в таблице коммутации будет найдена запись, ассоциирующая MAC-адрес приемника (компьютера B) с одним из портов коммутатора, за исключением порта-источника, кадр будет передан через соответствующий выходной порт (в приведенном примере — порт 2). Этот процесс называется продвижением (forwarding) кадра.

Если бы оказалось, что выходной порт и порт-источник совпадают, то передаваемый кадр был бы отброшен коммутатором. Этот процесс называется фильтрацией (filtering).

В том случае, если MAC-адрес приемника в поступившем кадре неизвестен (в таблице коммутации отсутствует соответствующаязапись), коммутатор создает множество копий этого кадра и передает их через все свои порты, за исключением того, на который он поступил. Этот процесс называется лавинной передачей (flooding). Несмотря на то, что процесс лавинной передачи занимает полосу пропускания, он позволяет коммутатору избежать потери кадров, когда MAC-адрес приемника неизвестен, и осуществлять процесс самообучения.

Лекция № 2 Основы коммутации. Типы интерфейсов коммутаторов.

1. Архитектура коммутаторов.

7.Архитектура коммутаторов.

Одним из основных компонентов всего коммутационного оборудования является коммутирующая матрица (switchfabric). Коммутирующаяматрица представляет собой чипсет, соединяющий множество входов с множеством выходов на основе фундаментальных технологий и принципов коммутации. Коммутирующая матрица выполняет три функции:

- переключает трафик с одного порта матрицы на другой, обеспечивая их равнозначность;
- предоставляет качество обслуживания (QualityofService, QoS);
- обеспечивает отказоустойчивость.

7.1 Архитектура с разделяемой шиной;

Архитектура с разделяемой шиной (SharedBus), как следует из ее названия, использует в качестве разделяемой среды шину, которая обеспечивает связь подключенных к ней устройств ввода-вывода (портов).

7.2 Архитектура с разделяемой памятью;

Улучшения архитектуры с разделяемой шиной привели к появлению высокопроизводительной архитектуры с разделяемой памятью (SharedMemory).

7.3 Архитектура на основе коммутационной матрицы:

Параллельно с появлением архитектуры с разделяемой памятью (в середине 1990-х годов) была разработана архитектура на основе коммутационной матрицы (Crossbararchitecture). Эта архитектура используется для построения коммутаторов различных типов.

В коммутаторах на основе коммутационной матрицы с буферизацией буферы расположены на трех основных стадиях: на входе и выходе и непосредственно на коммутационной матриц.

Эта архитектура характеризуется наличием безбуферных коммутирующих элементов и арбитра, который управляет передачей трафика между входами и выходами матрицы.

Лекция № 3 Начальная настройка коммутатора.

1. **Классификация коммутаторов по возможности управления.**
2. **Средства управления коммутаторами.**

Краткое содержание вопросов:

1.Классификация коммутаторов по возможности управления

Коммутаторы локальной сети можно классифицировать *по* возможности управления. Существует три категории коммутаторов:

- неуправляемые коммутаторы;
- управляемые коммутаторы;
- настраиваемые коммутаторы.

Неуправляемые коммутаторы не поддерживают возможности управления и обновления программного обеспечения.

Управляемые коммутаторы являются сложными устройствами, позволяющими выполнять расширенный набор функций 2-го и 3-го уровня модели *OSI*. Управление коммутаторами может осуществляться посредством Web-интерфейса, командной строки (*CLI*), протокола *SNMP*, Telnet и т.д.

Настраиваемые коммутаторы занимают промежуточную позицию между ними. Они предоставляют пользователям возможность настраивать определенные параметры сети с помощью интуитивно понятных утилит управления, Web-интерфейса, упрощенного интерфейса командной строки, протокола *SNMP*.

2. Средства управления коммутаторами

Большинство современных коммутаторов поддерживают различные функции управления и мониторинга. К ним относятся дружественный пользователю Web-*интерфейс* управления, *интерфейс* командной строки (*CommandLineInterface*, *CLI*), Telnet, *SNMP*-управление. В коммутаторах D-Link серии *Smart* также реализована *поддержка* начальной настройки и обновления программного обеспечения через утилиту D-LinkSmartConsole *Utility*.

Web-*интерфейс* управления позволяет осуществлять настройку и *мониторинг* параметров коммутатора, используя любой *компьютер*, оснащенный стандартнымWeb-браузером. *Браузер* представляет собой универсальное средство доступа и может непосредственно подключаться к коммутатору *по* протоколу *HTTP*.

Главная страница Web-интерфейса обеспечивает *доступ* к различным настройкам коммутатора и отображает всю необходимую информацию об устройстве. *Администратор* может быстро посмотреть статус устройства, статистику *по* производительности и т.д., а также произвести необходимые настройки.

Доступ к интерфейсу командной строки коммутатора осуществляется путем подключения к его консольному порту терминала или персонального компьютера с установленной программой эмуляции терминала. Это *метод доступа* наиболее удобен при первоначальном подключении к коммутатору, когда *значение* IP-адреса неизвестно или не установлено, в случае необходимости восстановления пароля и при выполнении

расширенных настроек коммутатора. Также *доступ* к интерфейсу командной строки может быть получен *по* сети с помощью протокола Telnet.

Пользователь может использовать для настройки коммутатора любой удобный ему *интерфейс* управления, т.к. набор доступных через разные интерфейсы управления функций одинаков для каждой конкретной модели.

Еще один способ управления коммутатором — использование протокола *SNMP* (Simple *NetworkManagement Protocol*). Протокол*SNMP* является протоколом 7-го уровня модели *OSI* и разработан специально для управления и мониторинга сетевыми устройствами и приложениями связи. Это выполняется путем обмена управляющей информацией между агентами, располагающимися на сетевых устройствах, и менеджерами, расположенными на станциях управления. Коммутаторами D-Link поддерживается протокол *SNMP*версий 1, 2c и 3.

Также стоит отметить возможность обновления программного обеспечения коммутаторов (за исключением неуправляемых). Это обеспечивает более долгий срок эксплуатации устройств, т.к. позволяет добавлять новые функции либо устранять имеющиеся ошибки *по* мере выхода новых версий *ПО*, что существенно облегчает и удешевляет использование устройств. Компания D-Link распространяет новые версии *ПО* бесплатно. Сюда же можно включить возможность сохранения настроек коммутатора на случай сбоев с последующим восстановлением или тиражированием, что избавляет администратора от выполнения рутинной работы.

Лекция № 4 Начальная настройка коммутатора. (Интерактивная форма-2 ч)

1. Подключение к коммутатору.

Начальная конфигурация коммутатора

3.Подключение к коммутатору:

Перед тем, как начать настройку коммутатора, необходимо установить физическое соединение между ним и рабочей станцией. Существуют два типа кабельного соединения, используемых для управления коммутатором. Первый тип — через консольный *порт*(если он имеется у устройства), второй — через *порт Ethernet* (*по* протоколу Telnet или через Web-*интерфейс*). Консольный *порт*используется для первоначальной конфигурации коммутатора и обычно не требует настройки. Для того чтобы получить *доступ* к коммутатору через *порт Ethernet*, в браузере необходимо ввести IP-адрес *по* умолчанию его интерфейса управления (обычно он указан в руководстве пользователя).

При подключении к медному (*разъем RJ-45*) порту *Ethernet* коммутатора *Ethernet*-совместимых серверов, маршрутизаторов или рабочих станций используется четырехпарный *кабель UTP* категории 5, 5e или 6 для *Gigabit Ethernet*. Поскольку коммутаторы D-Link поддерживают функцию автоматического определения полярности (*MDI*/MDIX), можно использовать любой тип кабеля (*прямой* или кроссовый).

Для подключения к медному (*разъем RJ-45*) порту *Ethernet* другого коммутатора также можно использовать любой четырехпарный*кабель UTP* категории 5, 5e, 6, при условии, что порты коммутатора поддерживают автоматическое *определение* полярности. В противном случае надо использовать кроссовый *кабель*.

Правильность подключения поможет определить светодиодная индикация порта. Если соответствующий *индикатор* горит, то *связь*между коммутатором и подключенным

устройством установлена. Если *индикатор* не горит, возможно, что не включено питание одного из устройств, или возникли проблемы с сетевым адаптером подключенного устройства, или имеются неполадки с кабелем. Если *индикатор* загорается и гаснет, возможно, есть проблемы с автоматическим определением скорости и режимом работы (дуплекс/полудуплекс) (за подробным описанием сигналов индикаторов необходимо обратиться к руководству пользователя коммутатора конкретной модели).

Управляемые коммутаторы D-Link оснащены консольным портом. В зависимости от модели коммутатора консольный порт может обладать разъемом DB-9 или *RJ-45*. С помощью консольного кабеля, входящего в комплект поставки, коммутатор подключается к последовательному порту компьютера. Подключение по консоли иногда называют "Out-of-Band-подключением. Это означает, что консоль использует отличную от обычного сетевого подключения схему (не использует полосу пропускания портов Ethernet).

После подключения к консольному порту коммутатора на персональном компьютере необходимо запустить программу эмуляции терминала VT100 (например, программу HyperTerminal в Windows). В программе следует установить следующие параметры подключения, которые, как правило, указаны в документации к устройству:

Скорость (бит/с): **9600 или 115200**

Биты данных: 8

Четность: нет

Стоповые биты: 1

Управление потоком: нет

При соединении коммутатора с консолью появится следующее окно (только для коммутаторов, имеющих поддержку интерфейса командной строки CLI).

Если окно не появилось, необходимо нажать *Ctrl+r*, чтобы его обновить.

Все управляемые коммутаторы обладают защитой от доступа неавторизованных пользователей, поэтому после загрузки устройства появится приглашение ввести имя пользователя и пароль. По умолчанию имя пользователя и пароль не определены, поэтому необходимо дважды нажать клавишу Enter. После этого в командной строке появится следующее приглашение, например DES-3528#. Теперь можно вводить команды.

4.Начальна конфигурация коммутатора.

Вызов помощи по командам.

Существует большое количество команд CLI. Команды бывают сложные, многоуровневые, требующие ввода большого количества параметров, и простые, состоящие из одного параметра. Наберите в командной строке "?" и нажмите клавишу *"Enter"*, для того чтобы вывести на экран список всех команд данного уровня.

4.2 Базовая конфигурация коммутатора.

Самым первым шагом при создании конфигурации коммутатора является обеспечение его защиты от доступа неавторизованных пользователей. Самая простая форма безопасности — создание учетных записей для пользователей с соответствующими правами. Создавая учетную запись для пользователя, можно задать один из следующих

уровней привилегий: *Admin, Operator* или *User*. Учетная запись *Admin*имеет наивысший *уровень привилегий.*

Лекция № 5 Обзор функциональных возможностей коммутаторов. (Интерактивная форма-2 ч)

1. Подключение к коммутатору.

Подключение к консоли интерфейса командной строки коммутатора

1. Подключение к коммутатору:

Перед тем, как начать настройку коммутатора, необходимо установить физическое соединение между ним и рабочей станцией. Существуют два типа кабельного соединения, используемых для управления коммутатором. Первый тип — через консольный *порт*(если он имеется у устройства), второй — через *порт Ethernet* (*по* протоколу Telnet или через Web-*интерфейс*). Консольный *порт* используется для первоначальной конфигурации коммутатора и обычно не требует настройки. Для того чтобы получить *доступ* к коммутатору через *порт Ethernet*, в браузере необходимо ввести IP-адрес *по* умолчанию его интерфейса управления (обычно он указан в руководстве пользователя).

При подключении к медному (*разъем RJ-45*) порту *Ethernet* коммутатора *Ethernet*-совместимых серверов, маршрутизаторов или рабочих станций используется четырехпарный *кабель UTP* категории 5, 5e или 6 для *Gigabit Ethernet*. Поскольку коммутаторы D-Link поддерживают функцию автоматического определения полярности (*MDI*/MDIX), можно использовать любой тип кабеля (*прямой* или кроссовый).

Для подключения к медному (*разъем RJ-45*) порту *Ethernet* другого коммутатора также можно использовать любой четырехпарный*кабель UTP* категории 5, 5e, 6, при условии, что порты коммутатора поддерживают автоматическое *определение* полярности. В противном случае надо использовать кроссовый *кабель*.

Правильность подключения поможет определить светодиодная индикация порта. Если соответствующий *индикатор* горит, то *связь*между коммутатором и подключенным устройством установлена. Если *индикатор* не горит, возможно, что не включено питание одного из устройств, или возникли проблемы с сетевым адаптером подключенного устройства, или имеются неполадки с кабелем. Если *индикатор* загорается и гаснет, возможно, есть проблемы с автоматическим определением скорости и режимом работы (дуплекс/полудуплекс) (за подробным описанием сигналов индикаторов необходимо обратиться к руководству пользователя коммутатора конкретной модели).

2. Подключение к консоли интерфейса командной строки коммутатора.

Управляемые коммутаторы D-Link оснащены консольным портом. В зависимости от модели коммутатора консольный порт может обладать разъемом DB-9 или *RJ-45*. С помощью консольного кабеля, входящего в комплект поставки, коммутатор подключается к последовательному порту компьютера. Подключение по консоли иногда называют "Out-of-Band-подключением. Это означает, что консоль использует отличную от обычного сетевого подключения схему (не использует полосу пропускания портов Ethernet).

После подключения к консольному порту коммутатора на персональном компьютере необходимо запустить программу эмуляции терминала VT100 (например,

программу HyperTerminal в Windows). В программе следует установить следующие параметры подключения, которые, как правило, указаны в документации к устройству:

Скорость (бит/с): **9600 или 115200**

Биты данных: 8

Четность: нет

Стоповые биты: 1

Управление потоком: нет

При соединении коммутатора с консолью появится следующее окно (только для коммутаторов, имеющих поддержку интерфейса командной строки CLI).

Если окно не появилось, необходимо нажать *Ctrl+r*, чтобы его обновить.

Все управляемые коммутаторы обладают защитой от доступа неавторизованных пользователей, поэтому после загрузки устройства появится приглашение ввести имя пользователя и пароль. По умолчанию имя пользователя и пароль не определены, поэтому необходимо дважды нажать клавишу Enter. После этого в командной строке появится следующее приглашение, например DES-3528#. Теперь можно вводить команды.

Лекция № 6 Обзор функциональных возможностей коммутаторов. (Интерактивная форма-2 ч)

1. **Начальная конфигурация коммутатора.**
2. **Вызов помощи по командам.**
3. **Базовая конфигурация коммутатора.**

4. Начальная конфигурация коммутатора.
5. Существует большое количество команд CLI. Команды бывают сложные, многоуровневые, требующие ввода большого количества параметров, и простые, состоящие из одного параметра. Наберите в командной строке "?" и нажмите клавишу *"Enter"*, для того чтобы вывести на экран список всех команд данного уровня.
6. Используйте знак вопроса "?" также в том случае, если вы не знаете параметров команды. Например, если надо узнать возможные варианты синтаксиса команды show, введите в командной строке:
7. DES-3528#show + пробел
8. Далее можно ввести "?" или нажать кнопку *Enter*. На экране появятся все возможные завершения команды. Также можно воспользоваться кнопкой *TAB*, которая будет последовательно выводить на экран все возможные завершения команды.

```
..
?
cable_diag ports
cfm linktrace
cfm lock md
cfm loopback
clear
clear address_binding dhcp_snoop binding_entry ports
clear address_binding nd_snoop binding_entry ports
clear arptable
clear attack_log
clear cfm pkt_cnt
clear counters
clear dhcp binding
clear dhcp conflict_ip
clear ethernet_oam ports
clear fdb
clear igmp_snooping data_driven_group
clear igmp_snooping statistics counter
clear jwac auth_state
clear log
clear mac_based_access_control auth_state
CTRL+C ESC q Quit SPACE n Next Page ENTER Next Entry a All
```

Рис. 1. Результат выполнения команды "?"

Внимание: при работе в CLI можно вводить сокращенный вариант команды. Например, если ввести команду "shsw", то коммутатор интерпретирует эту команду как "showswitch".

```
DES-3528:admin#show
Command: show
Next possible completions:
802.1p              802.1x              access_profile      account
accounting          acct_client         address_binding
arp_spoofing_prevention                 arpentry            asymmetric_vlan
attack_log          auth_client         auth_diagnostics
auth_session_statistics                 auth_statistics     authen
authen_enable       authen_login        authen_policy       authentication
authorization       autoconfig          bandwidth_control   bpdu_protection
cfm                 command             command_history     config
cpu                 cpu_filter          current_config      device_status
dhcp                dhcp_local_relay    dhcp_relay          dhcp_server
dhcpv6_relay        dnsr                dot1v_protocol_group
dscp                duld                erps                error
```

Рис. 2. Результат вызова помощи о возможных параметрах команды show

4. Вызов помощи по командам.

Существует большое количество команд CLI. Команды бывают сложные, многоуровневые, требующие ввода большого количества параметров, и простые, состоящие из одного параметра. Наберите в командной строке "?" и нажмите клавишу *"Enter"*, для того чтобы вывести на экран список всех команд данного уровня.

5.Базовая конфигурация коммутатора.

Шаг 1. Обеспечение защиты коммутатора от доступа неавторизованных пользователей.

Самым первым шагом при создании конфигурации коммутатора является обеспечение его защиты от доступа неавторизованных пользователей. Самая простая форма безопасности — создание учетных записей для пользователей с

соответствующими правами. Создавая учетную запись для пользователя, можно задать один из следующих уровней привилегий: *Admin, Operator* или *User*. Учетная запись *Admin* имеет наивысший *уровень привилегий*.

21. Создать *учетную запись пользователя* можно с помощью следующих команд CLI:
22. create account [admin | operator | user] <username 15>
23. Далее появится приглашение для ввода пароля и подтверждения ввода:
24. Enter a case-sensitive new password:
25. Enter the new password again for confirmation:
26. Максимальная длина имени пользователя и пароля — от 0 до 15 символов. На коммутаторе можно создать до 10 учетных записей пользователей. После успешного создания учетной записи на экране появится слово Success.
27. **Внимание**: все команды чувствительны к регистру. Перед вводом команды удостоверьтесь, что отключен *CapsLock* или другие нежелательные функции, которые изменят регистр текста.
28. Ниже приведен пример создания учетной записи с уровнем привилегий "admin" и именем пользователя (*Username*) "dlink" на коммутаторе *DES-3528:*
29. DES-3528#create account admin dlink
30. Command: create account admin dlink
31. Enter a case-sensitive new password:****
32. Enter the new password again for confirmation:****
33. Success.
34. Изменить пароль для пользователя с существующей учетной записью, можно с помощью команды
35. config account <username> {encrypt [plain_text | sha_1]
36. <password>}
37. Ниже приведен пример создания на коммутаторе *DES-3528* нового пароля для учетной записи dlink:
38. DES-3528#config account dlink
39. Command: config account dlink
40. Enter a old password:****
41. Enter a case-sensitive new password:****
42. Enter the new password again for confirmation:****
43. Success
44. Проверить созданную учетную запись можно с помощью команды
45. showaccount
46. Ниже приведен пример выполнения этой команды на коммутаторе *DES-3528*.
47. DES-3528#show account
48. Command: show account
49. Current Accounts:
50. Username Access Level
51. dlink Admin
52. TotalEntries: 1
53. Удалить учетную запись можно, выполнив команду
54. deleteaccount<username>
55. Ниже приведен пример удаления учетной записи dlink на коммутаторе *DES-3528*.

56. DES-3528#delete account dlink
57. Command: delete account dlink
58. Are you sure to delete the last administrator account?(y/n)
59. Success.
60. **Шаг 2.** Настройка IP-адреса.
61. Для того чтобы коммутатором можно было удаленно управлять через Web-интерфейс или Telnet, ему необходимо назначить IP-адрес из адресного пространства сети, в которой планируется его использовать. IP-адрес может быть задан автоматически, с помощью протоколов DHCP или *BOOTP*, или статически, с помощью следующих команд CLI:
62. configipifSystemdhcp
63. configipifSystemipaddressxxx.xxx.xxx.xxx/yyy.yyy.yyy.yyy
64. где xxx.xxx.xxx.xxx — IP-адрес, yyy.yyy.yyy.yyy — *маска подсети*, System — имя *управляющего интерфейса* коммутатора.
65. Ниже приведен пример использования команды присвоения IP-адреса *управляющему интерфейсу* на коммутаторе *DES-3528*:
66. DES-3528#config ipif System ipaddress 192.168.100.240/255.255.255.0
67. Command: configipif System ipaddress 192.168.100.240/24
68. Success.
69. **Шаг 3.** Настройка параметров портов коммутатора.
70. По умолчанию порты всех коммутаторов *D-Link* поддерживают автоматическое определение скорости и режима работы (дуплекса). Но может возникнуть ситуация, в которой автоопределение будет действовать некорректно и потребуется ручная установка скорости и режима. В этом случае ручную настройку параметров необходимо выполнить на обоих концах канала связи.
71. Для установки параметров портов, таких как скорость передачи, дуплексный/полудуплексный режим работы, активизация/отключение управления потоком, изучение MAC-адресов, автоматическое определение полярности и т.д., на коммутаторах *D-Link*можно воспользоваться командой configports.
72. Ниже приведен пример настройки параметров портов на коммутаторе *DES-3528*. Для портов 1, 2, 3 производятся следующие настройки: скорость передачи устанавливается равной 10 Мбит/с, активизируются дуплексный режим работы, изучение MAC-адресов, управление потоком, порты переводятся в состояние "включен".
73. DES-3528#config ports 1-3 speed 10_full learning enable state
74. enableflow_control enable
75. Command: config ports 1-3 speed 10_full learning enable state
76. enableflow_controlenable
77. Success
78. Команда showports<список портов> выведет на экран информацию о настройках портов коммутатора. Ниже показан результат выполнения команды showports на коммутаторе *DES-3528*.
79. DES-3528#show ports 1-3
80. Command: show ports 1-3
81. Port State/ Settings Connection Address

82. MDIX Speed/Duplex/ Speed/Duplex/ Learning
83. FlowCtrlFlowCtrl
84. 1 Enabled Auto 10M/Full/Enabled Link Down Enabled
85. 2 Enabled Auto 10M/Full/Enabled Link Down Enabled
86. 3 Enabled Auto 10M/Full/Enabled Link Down Enabled
87. **Шаг 4.** Сохранение текущей конфигурации коммутатора в энергонезависимую память *NVRAM*. Дляэтогонеобходимовыполнитькоманду save.
88. DES-3528#save
89. Command: save
90. Saving all settings to NV-RAM……Done
91. **Внимание**: активная конфигурация хранится в оперативной памяти *SDRAM*. При отключении питания конфигурация, хранимая в этой памяти, будет потеряна. Для того чтобы сохранить конфигурацию в энергонезависимой памяти *NVRAM*, необходимо выполнить команду "save".
92. **Шаг 5**. Перезагрузка коммутатора с помощью команды *reboot*.
93. DES-3528#reboot
94. Command: reboot
95. Are you sure you want to proceed with the system reboot? (y/n)
96. Please wait, the switch is rebooting...
97. Сброс настроек коммутатора к заводским установкам выполняется с помощью команды reset {[config | system]} {force_agree}
98. Если в команде не будет указано никаких ключевых слов, то все параметры, за исключением IP-адреса, учетных записей пользователей и *Log-файла*, будут возвращены к заводским параметрам по умолчанию. Коммутатор не сохранит настройки в энергонезависимой памяти *NVRAM* и не перегрузится.
99. Если указано ключевое слово config, на коммутаторе восстановятся все заводские настройки по умолчанию, включая IP-адрес интерфейса *управления, учетные записи* пользователей и журнал регистрации. Коммутатор не сохранит настройки в энергонезависимой памяти *NVRAM* и не перезагрузится.
100. Если указано ключевое слово system, на коммутаторе восстановятся все заводские настройки по умолчанию в полном объеме. Коммутатор сохранит эти настройки в энергонезависимой памяти *NVRAM* и перезагрузится.
101. Параметр force_agree позволяет произвести безусловное выполнение команды reset. Не нужно вводить "Y/N". На коммутаторе восстановятся все заводские настройки по умолчанию, исключая IP-адрес, учетные записи пользователей и журнал регистрации.
102. DES-3528#reset
103. Command: reset
104. Success
105. **Шаг 6.** Просмотр конфигурации коммутатора.
106. Получить информацию о коммутаторе (посмотреть его общую конфигурацию) можно с помощью команды showswitch.
107. Команды "Show" являются удобным средством проверки состояния и параметров коммутатора, предоставляя информацию, требуемую для мониторинга

и поиска неисправностей в работе коммутаторов. Ниже приведен список наиболее общих команд"Show".

showconfig	**эта команда используется для отображения конфигурации, сохраненной в *NVRAM* или созданной в текущий момент**
showfdb	эта команда используется для отображения текущей *таблицы коммутации*
showswitch	эта команда используется для отображения общей информации о коммутаторе
showdevice_status	эта команда используется для отображения состояния внутреннего и внешнего питания коммутатора
showerrorports	эта команда используется для отображения статистики об ошибках для заданного диапазона портов
showpacketports	эта команда используется для отображения статистики о переданных и полученных портом пакетах
show *firmware*information	эта команда используется для отображения информации о программном обеспечении коммутатора (прошивке)
showipif	эта команда используется для отображения информации о настройках IP-интерфейса на коммутаторе
showlog	эта команда используется для просмотра *Log-файла* коммутатора

Лекция № 7 Виртуальные локальные сети (VLAN). (Интерактивная форма-2 ч)

1. **Типы VLAN.**
2. **VLAN на основе портов.**
3. **VLAN на основе стандарта IEEE 802.1Q.**

1. Типы VLAN.

В коммутаторах могут быть реализованы следующие типы *VLAN*:

- на основе портов;
- на основе стандарта *IEEE 802.1Q*;
- на основе стандарта IEEE 802.1ad (Q-in-Q *VLAN*);
- на основе портов и протоколов IEEE 802.1v;
- на основе MAC-адресов;
- асимметричные.

Также для *сегментирования* сети на канальном уровне модели *OSI* в коммутаторах могут использоваться другие функции, например *функцияTrafficSegmentation*.

2. VLAN на основе портов.

При использовании *VLAN* на основе портов (Port-based *VLAN*) каждый *порт* назначается в определенную *VLAN*, независимо от того, какой *пользователь* или *компьютер* подключен к этому порту. Это означает, что все пользователи, подключенные к этому порту, будут членами одной *VLAN*. *Конфигурация* портов статическая и может быть изменена только вручную.

Основные характеристики *VLAN* на основе портов:

1. применяются в пределах одного коммутатора. Если необходимо организовать несколько рабочих групп в пределах небольшой сети на основе одного коммутатора, например, необходимо разнести технический отдел и отдел продаж, то решение *VLAN* на базе портов оптимально подходит для данной задачи;

2. простота настройки. Создание виртуальных сетей на основе группирования портов не требует от администратора большого объема ручной работы — достаточно всем портам, помещаемым в одну *VLAN*, присвоить одинаковый идентификатор *VLAN*(*VLAN* ID);

3. возможность изменения логической топологии сети без физического перемещения станций. Достаточно всего лишь изменить настройки порта с одной *VLAN* (например, *VLAN* технического отдела) на другую (*VLAN* отдела продаж), и рабочая станция сразу же получает возможность совместно использовать ресурсы с членами новой *VLAN*. Таким образом, *VLAN* обеспечивают гибкость при перемещениях, изменениях и наращивании сети;

4. каждый порт может входить только в одну *VLAN*. Для объединения виртуальных подсетей как внутри одного коммутатора, так и между двумя коммутаторами, нужно использовать сетевой уровень OSI-модели. Один из портов каждой *VLAN* подключается к интерфейсу маршрутизатора, который создает таблицу маршрутизации для пересылки кадров из одной подсети (*VLAN*) в другую (IP-адреса подсетей должны быть разными).

Недостатком такого решения является то, что один *порт* каждой *VLAN* необходимо подключать к маршрутизатору. Это приводит к*дополнительным расходам* на покупку кабелей и маршрутизаторов, а также порты коммутатора используются очень расточительно. Решить данную проблему можно двумя способами: использовать коммутаторы, которые на основе фирменного решения позволяют включать *порт* в несколько *VLAN*, или использовать коммутаторы уровня 3.

.

3. VLAN на основе стандарта IEEE 802.1Q.

Построение *VLAN* на основе портов основано только на добавлении дополнительной информации к адресным таблицам коммутатора и не использует возможности *встраивания* информации о принадлежности к виртуальной сети в передаваемый *кадр*.*Виртуальные локальные сети*, построенные на основе стандарта *IEEE 802.1Q*, используют дополнительные поля кадра для хранения информации о принадлежности к *VLAN* при его перемещении *по* сети. С точки зрения удобства и гибкости настроек, *VLAN* стандарта*IEEE 802.1Q* является лучшим решением *по* сравнению с *VLAN* на основе портов. Его основные преимущества:

1. гибкость и удобство в настройке и изменении — можно создавать необходимые комбинации *VLAN* как в пределах одного коммутатора, так и во всей сети, построенной на коммутаторах с поддержкой стандарта *IEEE 802.1Q*. Способность добавления тегов позволяет информации о *VLAN* распространяться через множество 802.1Q-совместимых коммутаторов по одному физическому соединению (*магистральному каналу, TrunkLink*);

2. позволяет активизировать алгоритм связующего дерева (*SpanningTree*) на всех портах и работать в обычном режиме. Протокол*SpanningTree* оказывается весьма полезным для применения в крупных сетях, построенных на нескольких коммутаторах, и позволяет коммутаторам автоматически определять древовидную конфигурацию связей в сети при произвольном соединении портов между собой. Для нормальной работы

коммутатора требуется отсутствие *замкнутых маршрутов* в сети. Эти маршруты могут создаваться администратором специально для образования резервных связей или же возникать случайным образом, что вполне возможно, если сеть имеет многочисленные связи, а кабельная система плохо структурирована или документирована. С помощью протокола *SpanningTree* коммутаторы после построения схемы сети блокируют избыточные маршруты. Таким образом, автоматически предотвращается возникновение петель в сети;

3. способность *VLAN IEEE 802.1Q* добавлять и извлекать теги из заголовков кадров позволяет использовать в сети коммутаторы и сетевые устройства, которые не поддерживают стандарт *IEEE 802.1Q*;

4. устройства разных производителей, поддерживающие стандарт, могут работать вместе, независимо от какого-либо фирменного решения;

5. чтобы связать подсети на сетевом уровне, необходим маршрутизатор или коммутатор L3. Однако для более простых случаев, например, для организации доступа к серверу из различных *VLAN*, маршрутизатор не потребуется. Нужно включить порт коммутатора, к которому подключен сервер, во все подсети, а сетевой адаптер сервера должен поддерживать стандарт *IEEE 802.1Q*.

Некоторые определения IEEE 802.1Q

- **Tagging ("Маркировка кадра")** — процесс добавления информации о принадлежности к 802.1Q *VLAN* в заголовок кадра.
- **Untagging ("Извлечение тега из кадра")** — процесс извлечения информации о принадлежности к 802.1Q *VLAN* из заголовка кадра.
- **VLAN ID (VID)** — идентификатор *VLAN*.
- **Port VLAN ID (PVID)** — идентификатор порта *VLAN*.
- **Ingressport ("Входной порт")** — порт коммутатора, на который поступают кадры, и при этом принимается решение о принадлежности к *VLAN*.
- **Egressport ("Выходной порт")** — порт коммутатора, с которого кадры передаются на другие сетевые устройства, коммутаторы или рабочие станции, и, соответственно, на нем должно приниматься решение о маркировке.

Любой порт коммутатора может быть настроен как *tagged* (маркированный) или как *untagged* (немаркированный). Функция*untagging* позволяет работать с теми сетевыми устройствами виртуальной сети, которые не понимают тегов в заголовке кадра Ethernet. Функция *tagging* позволяет настраивать *VLAN* между несколькими коммутаторами, поддерживающими стандарт *IEEE 802.1Q*.

Лекция № 8 Виртуальные локальные сети (VLAN). (Интерактивная форма-2 ч)

1. **Продвижение кадров VLAN IEEE 802.1Q.**
2. **Пример настройки VLAN IEEE 802.1Q.**

4. Продвижение кадров VLANIEEE 802.1Q.

Решение о продвижении кадра внутри *виртуальной локальной сети* принимается на основе трех следующих видов правил.

- Правила входящего трафика (*ingressrules*) — классификация получаемых кадров относительно принадлежности к *VLAN*.

• Правила продвижения между портами (*forwardingrules*) — принятие решения о продвижении или отбрасывании кадра.

• Правила исходящего трафика (*egressrules*) — принятие решения о сохранении или удалении в заголовке кадра тега 802.1Q перед его передачей.

Правила входящего трафика выполняют классификацию каждого получаемого кадра относительно принадлежности к определенной *VLAN*, а также могут служить для принятия решения о приеме кадра для дальнейшей обработки или его отбрасывании на основе формата принятого кадра.

Классификация кадра по принадлежности *VLAN* осуществляется следующим образом:

1. если кадр не содержит информацию о *VLAN* (немаркированный кадр), то в его заголовок коммутатор добавляет тег с идентификатором *VID*, равным идентификатору PVID порта, через который этот кадр был принят;

2. если кадр содержит информацию о *VLAN* (маркированный кадр), то его принадлежность к конкретной *VLAN* определяется по идентификатору *VID* в заголовке кадра. Значение тега в нем не изменяется.

Активизировав функцию проверки формата кадра на входе, администратор сети может указать, кадры каких форматов будут приниматься коммутатором для дальнейшей обработки. Управляемые коммутаторы D-Link позволяют настраивать прием портами либо только маркированных кадров (*tagged_only*), либо обоих типов кадров — маркированных и немаркированных (*admitall*).

Правила продвижения между портами осуществляют принятие решения об отбрасывании или передаче кадра на порт назначения на основе его информации о принадлежности конкретной *VLAN* и MAC-адреса узла-приемника.

Если входящий кадр маркированный, то коммутатор определяет, является ли входной порт членом той же *VLAN*, путем сравнения идентификатора *VID* в заголовке кадра и набора идентификаторов *VID*, ассоциированных с портом, включая его PVID. Если нет, то кадр отбрасывается. Этот процесс называется *ingressfiltering* (*входной фильтрацией*) и используется для сохранения пропускной способности внутри коммутатора путем отбрасывания кадров, не принадлежащих той же *VLAN*, что и входной порт, на стадии их приема.

Если кадр немаркированный, входная фильтрация не выполняется.

Далее определяется, является ли порт назначения членом той же *VLAN*. Если нет, то кадр отбрасывается. Если же выходной порт входит в данную *VLAN*, то коммутатор передает кадр в подключенный к нему сегмент сети.

Правила исходящего трафика определяют формат исходящего кадра — маркированный или немаркированный. Если выходной порт является немаркированным (*untagged*), то он будет извлекать тег 802.1Q из заголовков всех выходящих через него маркированных кадров. Если выходной порт настроен как маркированный (*tagged*), то он будет сохранять тег 802.1Q в заголовках всех выходящих через него маркированных кадров.

5. Пример настройки VLANIEEE 802.1Q.

На рис. 3 показана схема сети, состоящая из двух групп *VLAN*. В качестве примера передачи данных между устройствами одной*VLAN*,построенной на нескольких

коммутаторах, рассмотрим пересылку кадра с порта 5 коммутатора 1 на порт 6 коммутатора 3.

- Порт 5 коммутатора 1 является немаркированным портом *VLAN* v2 (PVID=2). Поэтому, когда любой немаркированный кадр поступает на порт 5, коммутатор снабжает его тегом 802.1Q со значением *VID*, равным 2.
- Далее коммутатор 1 проверяет в своей *таблице коммутации*, через какой порт необходимо передать кадр и принадлежит ли этот порт *VLAN* v2. Кадр может быть передан через порт 1, т.к. он является маркированным членом *VLAN* v2. После передачи кадра через порт 1 тег 802.1Q в нем будет сохранен.
- После этого маркированный кадр поступит на порт 1 коммутатора 2. Прежде чем передать кадр дальше, порт 1 проверит, является ли он сам членом *VLAN* v2. Поскольку порт 1 коммутатора 2 является маркированным членом *VLAN* v2, он примет кадр и передаст его на порт 2, согласно *таблице коммутации*. После передачи кадра через порт 2 коммутатора 2 тег 802.1Q в нем будет сохранен, т.к. порт 2 является маркированным портом *VLAN* v2.
- Порт 1 коммутатора 3 примет поступивший кадр. После проверки на принадлежность к *VLAN* порт 1 передаст кадр на порт 6, найденный обычным образом в *таблице коммутации* коммутатора 3. Порт 6 является немаркированным портом *VLAN* v2, поэтому при выходе кадра через этот порт тег 802.1Q из него будет удален.

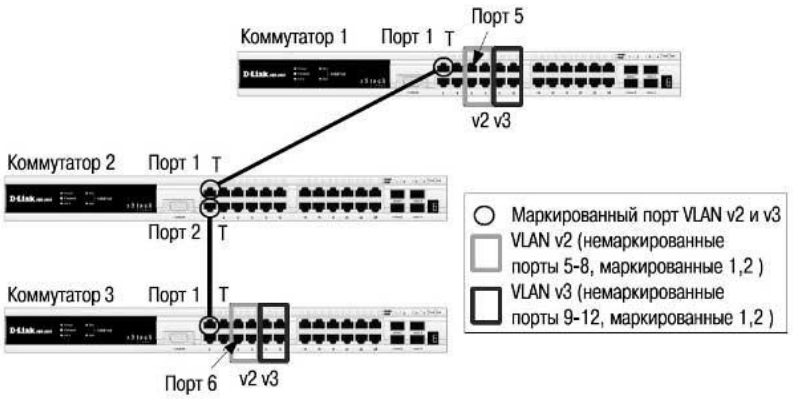

Рис. 3. Схема сети VLAN

Ниже приведен пример настройки коммутаторов, позволяющий реализовать заданную схему сети *VLAN*.

Настройка коммутатора 1

- Удалить соответствующие порты из *VLAN* по умолчанию (default *VLAN*) и создать новые *VLAN*.
  - configvlandefaultdelete 1-12
  - createvlan v2 tag 2
  - createvlan v3 tag 3

- В созданные *VLAN* добавить порты, для которых необходимо указать, какие из них являются маркированными и немаркированными.
  - configvlan v2 add untagged 5-8
  - configvlan v2 add tagged 1-2

- configvlan v3 add untagged 9-12
- configvlan v3 add tagged 1-2

Настройкакоммутатора 2
configvlan default delete 1-2
createvlan v2 tag 2
createvlan v3 tag 3
configvlan v2 add tagged 1-2
configvlan v3 add tagged 1-2
Настройкакоммутаторов 3
configvlan default delete 1-12
createvlan v2 tag 2
createvlan v3 tag 3
configvlan v2 add untagged 5-8
configvlan v2 add tagged 1
configvlan v3 add untagged 9-12
configvlan v3 add tagged 1

Лекция № 9 Функции повышения надежности и производительности.

1. Протоколы Spanning Tree.

1. Протоколы SpanningTree.

Протокол связующего дерева SpanningTreeProtocol (STP) является протоколом 2 уровня модели *OSI*, который позволяет строить*древовидные*, свободные от петель, конфигурации связей между коммутаторами локальной сети. Помимо этого, *алгоритм* обеспечивает возможность автоматического резервирования альтернативных каналов связи между коммутаторами на случай выхода активных каналов из строя.

Лекция № 10 Функции повышения надежности и производительности.

1. Spanning Tree Protocol (STP).

Rapid Spanning Tree Protocol Настройка RSTP

Понятиепетель.

Если для обеспечения избыточности между коммутаторами создается несколько соединений, то могут возникать коммутационные петли. Петля предполагает существование нескольких маршрутов по промежуточным сетям, а сеть с несколькими маршрутами между источником и приемником отличается повышенной отказоустойчивостью. Хотя наличие избыточных каналов связи очень полезно, петли, тем не менее, создают проблемы, самые актуальные из которых:

- широковещательные штормы;
- множественные копии кадров;
- множественные петли.

Построение активной топологии связующего дерева.

Для построения устойчивой активной топологии с помощью протокола *STP* необходимо с каждым коммутатором сети ассоциировать уникальный *идентификатор моста (Bridge ID)*, а с каждым портом коммутатора ассоциировать *стоимость пути (PathCost)* и *идентификатор порта (Port ID)*.

Bridge Protocol Data Unit (BPDU).

Вычисление связующего дерева происходит при включении коммутатора и при изменении топологии. Эти вычисления требуют периодического обмена информацией между коммутаторами связующего дерева, что достигается при помощи специальных кадров, называемых блоками данных протокола моста — BPDU (*Bridge* ProtocolDataUnit).

Состояние портов.

Описание стадий пройденных портом. Схемы.

Таймеры STR.

Для того чтобы все коммутаторы сети имели возможность получить точную информацию о конфигурации связующего дерева, в протоколе*STP* используются следующие таймеры**.**

Изменение топологии.

Коммутатор отправляет BPDU с уведомлением об изменении топологии (*Topology* Change *Notification* BPDU, TCN BPDU) в случае возникновения одного из следующих событий:

- некорневой мост получает сообщение TCN BPDU на свой назначенный порт;
- после истечения времени, определенного таймером Forward *Delay*, порт переходит в состояние Forwarding, но коммутатор уже имеет назначенный порт для данного сегмента;
- порт, находившийся в состоянии Forwarding или Listening, переходит в состояние Blocking (в случае проблем с каналом связи);
- когда коммутатор становится корневым мостом.

RapidSpanningTreeProtocol.

Протокол Rapid *SpanningTree Protocol* (*RSTP*) является развитием протокола *STP* и в настоящее время определен в стандарте *IEEE 802.1D*-2004 (ранее был определен в стандарте *IEEE* 802.1w-2001). Он был разработан для преодоления отдельных ограничений протокола *STP*, связанных с его производительностью. Протокол *RSTP* значительно ускоряет время сходимости коммутируемой сети за счет мгновенного перехода корневых и назначенных портов в состояние продвижения.

Роли портов.

Выбор активной топологии завершается присвоением протоколом *RSTP* определенной роли каждому порту. Эти роли следующие:

- корневой порт (RootPort);

- назначенный порт (*DesignatedPort*);
- альтернативный порт (AlternatePort);
- резервный порт (BackupPort).

Формат BPDU.

Формат кадра BPDU протокола *RSTP* аналогичен формату BPDU протокола *STP* за исключением следующего:

- поля версии протокола и типа BPDU *RSTP* содержат значение 2;
- в поле Flag BPDU протокола *STP* используются только два бита, которые определяют флаги изменения топологии TC и подтверждения TC (*TCA*). В поле Flag протокола *RSTP* используются все 8 бит. Бит 1 — флаг изменения топологии (*TopologyChange*), бит 2 — флаг предложения (*Proposal*), биты 3 и 4 предназначены для кодирования роли порта (*PortRole*), бит 5 — флаг изучения (*Learning*), бит 6 — флаг продвижения (*Forwarding*), бит 7 — флаг соглашения (*Agreement*), бит 8 — флаг подтверждения TC (*TopologyChangeAcknowledgment*).
- кадр BPDU протокола *RSTP* имеет дополнительное поле *Version 1 Length* длиной 1 байт. Это поле содержит значение 0000 0000 и показывает, что BPDU не содержит никакой информации протокола *STP* версии 1.

Быстрый переход в состояние продвижения.

Процесс построения связующего дерева у протоколов *STP* и *RSTP* одинаков. Однако при работе *RSTP* порт может перейти в состояние продвижения значительно быстрее, т.к. он больше не зависит от настроек таймеров. Протокол *RSTP* предоставляет механизм предложений и соглашений, который обеспечивает быстрый переход корневых и назначенных портов в состояние Forwarding, а альтернативных и резервных портов - в состояние *Discarding*.

Стоимость пути RSTP.

Протокол *RSTP* определяет следующие рекомендованные значения стоимости пути по умолчанию для портов коммутаторов. Эти значения вычисляются в соответствии со скоростью канала связи, к которому подключен порт.

Совместимость с STP.

Протокол *RSTP* может взаимодействовать с оборудованием, поддерживающим *STP*, и, если необходимо, автоматически преобразовывать кадры BPDU в формат 802.1D. Однако преимущество быстрой сходимости *RSTP* (когда все коммутаторы быстро переходят в состояние пересылки или блокировки и обладают тождественной информацией) теряется.

Лекция № 11 Адресация сетевого уровня и маршрутизация. (Интерактивная форма-2 ч)

1. **Сетевой уровень.**
2. **Обзор адресации сетевого уровня.**

1.Сетевой уровень.

Сетевой уровень (Networklayer) служит для образования единой транспортной системы, объединяющей несколько сетей, причем эти сети могут использовать совершенно различные принципы передачи сообщений между конечными узлами и обладать произвольной структурой связей. Функции сетевого уровня достаточно разнообразны. Начнем их рассмотрение на примере объединения локальных сетей.

Протоколы канального уровня локальных сетей обеспечивают доставку данных между любыми узлами только в сети с соответствующей типовой топологией, например топологией иерархической звезды. Это очень жесткое ограничение, которое не позволяет строить сети с развитой структурой, например, сети, объединяющие несколько сетей предприятия в единую сеть, или высоконадежные сети, в которых существуют избыточные связи между узлами. Можно было бы усложнять протоколы канального уровня для поддержания петлевидных избыточных связей, но принцип разделения обязанностей между уровнями приводит к другому решению. Чтобы с одной стороны сохранить простоту процедур передачи данных для типовых топологий, а с другой допустить использование произвольных топологий, вводится дополнительный сетевой уровень.

На сетевом уровне сам термин *сеть* наделяют специфическим значением. В данном случае под сетью понимается совокупность компьютеров, соединенных между собой в соответствии с одной из стандартных типовых топологий и использующих для передачи данных один из протоколов канального уровня, определенный для этой топологии.

Внутри сети доставка данных обеспечивается соответствующим канальным уровнем, а вот доставкой данных между сетями занимается сетевой уровень, который и поддерживает возможность правильного выбора маршрута передачи сообщения даже в том случае, когда структура связей между составляющими сетями имеет характер, отличный от принятого в протоколах канального уровня.

Сети соединяются между собой специальными устройствами, называемыми маршрутизаторами. *Маршрутизатор* — это устройство, которое собирает информацию о топологии межсетевых соединений и на ее основании пересылает пакеты сетевого уровня в сеть назначения. Чтобы передать сообщение от отправителя, находящегося в одной сети, получателю, находящемуся в другой сети, нужно совершить некоторое количество *транзитных передач между сетями, или хопов* (от *hop* — прыжок), каждый раз выбирая подходящий маршрут. Таким образом, маршрут представляет собой последовательность маршрутизаторов, через которые проходит пакет.

Проблема выбора наилучшего пути называется *маршрутизацией,* и ее решение является одной из главных задач сетевого уровня. Эта проблема осложняется тем, что самый короткий путь не всегда самый лучший. Часто критерием при выборе маршрута является время передачи данных по этому маршруту; оно зависит от пропускной способности каналов связи и интенсивности трафика, которая может изменяться с течением времени. Некоторые алгоритмы маршрутизации пытаются приспособиться к изменению нагрузки, в то время как другие принимают решения на основе средних показателей за длительное время. Выбор маршрута может осуществляться и по другим критериям, например надежности передачи.

В общем случае функции сетевого уровня шире, чем функции передачи сообщений по связям с нестандартной структурой, которые мы сейчас рассмотрели на примере

объединения нескольких локальных сетей. Сетевой уровень решает также задачи согласования разных технологий, упрощения адресации в крупных сетях и создания надежных и гибких барьеров на пути нежелательного трафика между сетями.

Сообщения сетевого уровня принято называть *пакетами (packets)*. При организации доставки пакетов на сетевом уровне используется понятие «номер сети». В этом случае адрес получателя состоит из старшей части — номера сети и младшей — номера узла в этой сети. Все узлы одной сети должны иметь одну и ту же старшую часть адреса, поэтому термину «сеть» на сетевом уровне можно дать и другое, более формальное определение: сеть — это совокупность узлов, сетевой адрес которых содержит один и тот же номер сети.

На сетевом уровне определяются два вида протоколов. Первый вид — *сетевые протоколы (routedprotocols)* — реализуют продвижение пакетов через сеть. Именно эти протоколы обычно имеют в виду, когда говорят о протоколах сетевого уровня. Однако часто к сетевому уровню относят и другой вид протоколов, называемых протоколами обмена маршрутной информацией или просто *протоколами маршрутизации (routingprotocols). С* помощью этих протоколов маршрутизаторы собирают информацию о топологии межсетевых соединений. Протоколы сетевого уровня реализуются программными модулями операционной системы, а также программными и аппаратными средствами маршрутизаторов.

На сетевом уровне работают протоколы еще одного типа, которые отвечают за отображение адреса узла, используемого на сетевом уровне, в локальный адрес сети. Такие протоколы часто называют *протоколами разрешения адресов* — AddressResolutionProtocol, ARP. Иногда их относят не к сетевому уровню, а к канальному, хотя тонкости классификации не изменяют их сути.

Примерами протоколов сетевого уровня являются протокол межсетевого взаимодействия IP стека TCP/IP и протокол межсетевого обмена пакетами IPX стека Novell.

2.Обзор адресации сетевого уровня.

Как и у других протоколов сетевого уровня, схема адресации IP является интегральной по отношению к процессу маршрутизации дейтаграмм IP через объединенную сеть. Длина адреса IP составляет 32 бита, разделенных на две или три части. Первая часть обозначает адрес сети, вторая (если она имеется) - адрес подсети, и третья - адрес главной вычислительной машины. Адреса подсети присутствуют только в том случае, если администратор сети принял решение о разделении сети на подсети. Длина полей адреса сети, подсети и главной вычислительной машины являются переменными величинами.

Лекция № 12 Адресация сетевого уровня и маршрутизация. (Интерактивная форма-2 ч)

1. **Формат пакета IPv4.**
2. **Представление и структура адреса IPv4.**

3. Формирование подсетей. 3.Формат пакета IPv4.Заголовок IP начинается с номера версии (versionnumber), который указывает номер используемой версии IP.

4. Поле длины заголовка (IHL) обозначает длину заголовка дейтаграммы в 32-битовых словах.

5. Поле типа услуги (type-of-service) указывает, каким образом должна быть обработана текущая дейтаграмма в соответствии с указаниями конкретного протокола высшего уровня. С помощью этого поля дейтаграммам могут быть назначены различные уровни значимости.

6. Поле общая длина (totallength) определяет длину всего пакета IP в байтах, включая данные и заголовок.

Поле идентификации (identification) содержит целое число, обозначающее текущую дейтаграмму. Это поле используется для соединения фрагментов дейтаграммы.

Поле флагов (flags) (содержащее бит DF, бит MF и сдвиг фрагмента) определяет, может ли быть фрагментирована данная дейтаграмма и является ли текущий фрагмент последним.

Поле срок жизни (time-to-live) поддерживает счетчик, значение которого постепенно уменьшается до нуля; в этот момент дейтаграмма отвергается. Это препятствует зацикливанию пакетов.

Поле протокола (protocol) указывает, какой протокол высшего уровня примет входящие пакеты после завершения обработки IP.

Поле контрольной суммы заголовка (headerchecksum) помогает обеспечивать целостность заголовка ID.

Поля адресов источника и пункта назначения (sourceanddestinationaddress) обозначают отправляющий и принимающий узлы.

Поле опции (options) позволяет IP обеспечивать факультативные возможности, такие, как защита данных.

Поле данных (data) содержит информацию высших уровней.

4.Представление и структура адреса IPv4.

IP-адрес состоит из двух частей: номера сети и номера узла. В случае изолированной сети её адрес может быть выбран администратором из специально зарезервированных для таких сетей блоков адресов (10.0.0.0/8, 172.16.0.0/12 или 192.168.0.0/16). Если же сеть должна работать как составная часть Интернета, то адрес сети выдаётся провайдером либо региональным интернет-регистратором (RegionalInternetRegistry, RIR). Согласно данным на сайте IANA, существует пять RIR: ARIN, обслуживающий Северную Америку, а также Багамы, Пуэрто-Рико и Ямайку; APNIC, обслуживающий страны Южной, Восточной и Юго-Восточной Азии, а также Австралии и Океании; AfriNIC, обслуживающий страны Африки; LACNIC, обслуживающий страны Южной Америки и бассейна Карибского моря; и RIPE NCC, обслуживающий Европу, Центральную Азию, Ближний Восток. Региональные регистраторы получают номера автономных систем и большие блоки адресов у IANA, а затем выдают номера автономных систем и блоки адресов меньшего размера локальным интернет-регистраторам (LocalInternetRegistries, LIR), обычно являющимся крупными провайдерами.

Номер узла в протоколе IP назначается независимо от локального адреса узла. Маршрутизатор по определению входит сразу в несколько сетей. Поэтому каждый порт маршрутизатора имеет собственный IP-адрес. Конечный узел также может входить в несколько IP-сетей. В этом случае компьютер должен иметь несколько IP-адресов, по числу сетевых связей. Таким образом, IP-адрес характеризует не отдельный компьютер или маршрутизатор, а одно сетевое соединение.

Лекция № 13 Адресация сетевого уровня и маршрутизация.

1. Протокол IPv6.

Типы адресов IPv6

Протокол IPv6.

IPv6 (англ. InternetProtocolversion 6) — новая версия протокола IP, призванная решить проблемы, с которыми столкнулась предыдущая версия (IPv4) при её использовании в Интернете, за счёт использования длины адреса 128 бит вместо 32. Протокол был разработан IETF.

В настоящее время протокол IPv6 уже используется в нескольких тысячах сетей по всему миру (более 14000 сетей на осень 2013), но пока ещё не получил столь широкого распространения в Интернете, как IPv4. На конец 2012 года доля IPv6 в сетевом трафике составляла около 1 %[1]. К концу 2013 года ожидался рост до 3 %. В России коммерческое использование операторами связи невелико (не более 1 % трафика). DNS-серверы многих российских регистраторов доменов и провайдеров хостинга используют IPv6.

7.Типы адресов IPv6.

Существуют различные типы адресов IPv6: одноадресные (Unicast), групповые (Anycast) и многоадресные (Multicast). Адреса типа Unicast хорошо всем известны. Пакет, посланный на такой адрес, достигает в точности интерфейса, который этому адресу соответствует. Адреса типа Anycast синтаксически неотличимы от адресов Unicast, но они адресуют группу интерфейсов. Пакет, направленный такому адресу, попадёт в ближайший (согласно метрике маршрутизатора) интерфейс. Адреса Anycast могут использоваться только маршрутизаторами. Адреса типа Multicast идентифицируют группу интерфейсов. Пакет, посланный на такой адрес, достигнет всех интерфейсов, привязанных к группе многоадресного вещания. Широковещательные адреса IPv4 (обычно xxx.xxx.xxx.255) выражаются адресами многоадресного вещания IPv6. Адреса разделяются двоеточиями (напр. fe80:0:0:0:200:f8ff: fe21:67cf). Большое количество нулевых групп может быть пропущено с помощью двойного двоеточия (fe80::200:f8ff: fe21:67cf). Такой пропуск должен быть единственным в адресе.

Лекция № 14 Качество обслуживания (QoS). (Интерактивная форма-2 ч)

1. Модели QoS

1. Модели QoS.

При передаче по одной сети трафика потоковых мультимедийных приложений (Voiceover IP (VoIP), IPTV, видеоконференции, он-лайн игры и др.) и трафика данных с различными требованиями к пропускной способности, необходимы механизмы, обеспечивающие возможность дифференцирования и обработки различных типов сетевого трафика в зависимости от предъявляемых ими требований. Негарантированная доставка данных (*besteffortservice*), традиционно используемая в сетях, построенных на основе коммутаторов, не предполагала проведения какой-либо классификации трафика и не обеспечивала надежную доставку трафика приложений, гарантированную пропускную способность канала и определенный уровень потери пакетов. Для решения этой проблемы было введено понятие **качества обслуживания**(*QualityofService, QoS*).

Функции качества обслуживания в современных сетях заключаются в обеспечении гарантированного и дифференцированного уровня обслуживания сетевого трафика, запрашиваемого теми или иными приложениями на основе различных механизмов

распределения ресурсов, ограничения интенсивности трафика, обработки очередей и приоритизации.

Можно выделить три модели реализации QoS в сети:

· **Негарантированная доставка данных (BestEffortService)**– обеспечивает связь между узлами, но не гарантирует надежную доставку данных, время доставки, пропускную способность и определенный приоритет.

· **Интегрированные услуги (IntegratedServices, IntServ)**– эта модель описана в RFC 1633 и предполагает предварительное резервирование сетевых ресурсов с целью обеспечения предсказуемого поведения сети для приложений, требующих для нормального функционирования гарантированной выделенной полосы пропускания на всем пути следования трафика. В качестве примера можно привести приложения IP-телефонии, которым для обеспечения приемлемого качества передачи голоса, требуется канал с минимальной пропускной способностью 64 Кбит/с (для кодека G.711). Модель IntServ использует сигнальный протокол RSVP (ResourceReservationProtocol, протокол резервирования ресурсов) для резервирования ресурсов для каждого потока данных, который должен поддерживаться каждым узлом на пути следования трафика. Эту модель также часто называют *жестким QoS* (*hardQoS*) в связи с предъявлением строгих требований к ресурсам сети.

· **Дифференцированное обслуживание (DifferentiatedService, DiffServ)**– эта модель описана в RFC 2474, RFC 2475 и предполагает разделение трафика на классы на основе требований к качеству обслуживания. В архитектуре DiffServ каждый передаваемый пакет снабжается информацией, на основании которой принимается решение о его продвижении на каждом промежуточном узле сети, в соответствии с политикой обслуживания трафика данного класса (Per-HopBehavior, PHB). Модель дифференцированного обслуживания занимает промежуточное положение между негарантированной доставкой данных и моделью IntServ и сама по себе не предполагает обеспечение гарантий предоставляемых услуг, поэтому дифференцированное обслуживание часто называют *мягким QoS* (*softQoS*).

Лекция № 15 Функции обеспечения безопасности и ограничения доступа к сети.

1. Списки управления доступом (ACL)

1. Списки управления доступом (ACL).

Списки управления доступом (AccessControlList, ACL) являются мощным средством фильтрации потоков данных без потери производительности, т.к. проверка содержимого пакетов выполняется на аппаратном уровне. Фильтруя *потоки данных*, *администратор*может ограничить типы приложений, разрешенных для использования в сети, контролировать *доступ* пользователей к сети и определять устройства, к которым они могут подключаться. Также *ACL* могут использоваться для определения политики *QoS* путем классификации трафика и переопределения его приоритета.

ACL представляют собой последовательность условий проверки параметров пакетов данных. Когда сообщения поступают на *входной порт,коммутатор* проверяет параметры пакетов данных на совпадение с критериями фильтрации, определенными в *ACL*, и выполняет над пакетами одно из действий: Permit ("Разрешить") или Deny ("Запретить"). Критерии фильтрации могут быть определены на основе следующей информации, содержащейся в пакете:

- порт коммутатора;
- MAC/ IP-адрес;
- тип Ethernet/ тип протокола;
- *VLAN*;
- 802.1p/ DSCP;
- порт TCP/ UDP (тип приложения);
- первые 80 байт пакета, включая поле данных.

Профили доступа и правила ACL.

Списки управления доступом состоят из *профилей доступа* (AccessProfile) и *правил* (Rule). Профили доступа определяют типы критериев фильтрации, которые должны проверяться в пакете данных (MAC-адрес, IP-адрес, номер порта, *VLAN* и т.д.), а в правилах непосредственно указываются значения их параметров. Каждый профиль может состоять из множества правил.

Лекция № 16 Функции обеспечения безопасности и ограничения доступа к сети.

1. Функции контроля над подключением узлов к портам

Функции контроля над подключением узлов к портам коммутатора.

В том случае, если какой-либо *порт* на коммутаторе активен, к нему может подключиться любой *пользователь* и получить несанкционированный *доступ* к сети. Этот *пользователь* может начать генерировать вредоносный трафик, который попадет в *сеть* и создаст проблемы внутри нее. Для защиты от подобных ситуаций, а также для контроля подключения узлов к портам коммутаторы D-Link предоставляют *функции безопасности*, которые позволяют указывать MAC- и/или IP-адреса устройств, которым разрешено подключаться к данному порту, и блокировать *доступ* к сети узлам с неизвестными коммутатору адресами.

Функция PortSecurity позволяет настроить какой-либо порт коммутатора так, чтобы доступ к сети через него мог осуществляться только определенными устройствами. Устройства, которым разрешено подключаться к порту, определяются по MAC-адресам. MAC-адреса могут быть изучены динамически или вручную настроены администратором сети. Помимо этого, функция PortSecurity позволяет ограничивать количество изучаемых портом MAC-адресов, тем самым ограничивая количество подключаемых к нему узлов.

Существует три режима работы функции PortSecurity:

• *Permanent* (Постоянный) – занесенные в таблицу коммутации MAC-адреса никогда не устаревают, даже если истекло время, установленное таймером FDB AgingTime, или коммутатор был перезагружен.

• *DeleteonTimeout* (Удалить при истечении времени) – занесенные в таблицу коммутации MAC-адреса устареют после истечения времени, установленного таймером FDB AgingTime, и будут удалены.

Если состояние канала связи на подключенном порте изменяется, MAC-адреса, изученные на нем, удаляются из таблицы коммутации, что аналогично выполнению действий при истечении времени, установленного таймером FDB AgingTime.

• *DeleteonReset* (Удалить при сбросе настроек) – занесенные в таблицу коммутации MAC-адреса будут удалены после перезагрузки коммутатора (этот режим используется по умолчанию).

При подключении неавторизованного пользователя к порту коммутатора, он будет заблокирован, а коммутатор отправит сообщение SNMP Trap или создаст запись в Log- файле, если администратор настроил выполнение этих действий. Порт коммутатора будет отбрасывать трафик, поступающий с неизвестного MAC-адреса.

Функция IP-MAC-PortBinding (IMPB), реализованная в коммутаторах D-Link, позволяет контролировать доступ компьютеров в сеть на основе их IP- и MAC-адресов, а также порта подключения. Администратор сети может создать записи ("белый лист"), связывающие MAC- и IP-адреса компьютеров с портами подключения коммутатора. На основе этих записей, в случае совпадения всех составляющих, клиенты будут получать доступ к сети со своих компьютеров. В том случае, если при подключении клиента связка MAC-IP-порт будет отличаться от параметров заранее сконфигурированной записи, коммутатор заблокирует MAC-адрес соответствующего узла с занесением его в "черный лист".

Функция IP-MAC-PortBinding специально разработана для управления подключением узлов в сетях ETTH (Ethernet-To-The-Home) и офисных сетях. Помимо этого функция IMPB позволяет бороться с атаками типа ARP Spoofing, во время которых злонамеренные пользователи перехватывают трафик или прерывают соединение, манипулируя пакетами ARP.

Функция IP-MAC-PortBinding включает три режима работы: ARP mode (по умолчанию), ACL mode и DHCP Snoopingmode.

ARP mode является режимом, используемым по умолчанию при настройке функции IP-MAC-PortBinding на портах. При работе в режиме ARP коммутатор анализирует ARP- пакеты и сопоставляет параметры IP-MAC ARP-пакета с предустановленной администратором связкой IP-MAC. Если хотя бы один параметр не совпадает, то MAC-адрес узла будет занесен в таблицу коммутации с отметкой «Drop»

(Отбрасывать). Если все параметры совпадают, MAC-адрес узла будет занесен в таблицу коммутации с отметкой

«Allow» (Разрешен).

При функционировании в *ACL mode*, коммутатор на основе предустановленного администратором «белого листа» IMPB создает правила ACL. Любой пакет, связка IP-MAC, которого отсутствует в «белом листе», будет блокироваться ACL. Если режим ACL отключен, правила для записей IMPB будут удалены из таблицы ACL

Режим *DHCP Snooping* используется коммутатором для динамического создания записей IP-MAC на основе анализа DHCP-пакетов и привязки их к портам с включенной функцией IMPB (администратору не требуется создавать записи вручную). Таким образом, коммутатор автоматически создает «белый лист» IMPB в таблице коммутации или аппаратной таблице ACL (если режим ACL включен). При этом для обеспечения корректной работы, сервер DHCP должен быть подключен к доверенному порту с выключенной функцией IMPB. Администратор может ограничить максимальное количество создаваемых в процессе автоизучения записей IP-MAC на порт, т.е. ограничить для каждого порта с активизированной функцией IMPB количество узлов, которые могут получить IP-адрес с DHCP-сервера. При работе в режиме DHCP Snooping коммутатор не будет создавать записи IP-MAC для узлов с IP-адресом установленным вручную.

При активизации функции IMPB на порте администратор должен указать режим его работы:

· **StrictMode**– в этом режиме порт по умолчанию заблокирован. Прежде чем передавать пакеты он будет отправлять их на ЦПУ для проверки совпадения их параметров IP-MAC с записями в «белом листе». Таким образом, порт не будет передавать пакеты до тех пор, пока не убедится в их достоверности. Порт проверяет все IP и ARP-пакеты.

· **LooseMode**– в этом режиме порт по умолчанию открыт. Порт будет заблокирован, как только через него пройдет первый недостоверный пакет. Порт проверяет только пакеты ARP и IP Broadcast.

Лекция № 17 Функции обеспечения безопасности и ограничения доступа к сети. (Интерактивная форма-2 ч)

1. Аутентификация пользователей 802.1X

2. Аутентификация пользователей 802.1X.

Стандарт *IEEE 802.1X* (*IEEE* Std 802.1X-2010) описывает использование протокола *EAP* (*Extensible AuthenticationProtocol*) для поддержки аутентификации с помощью сервера аутентификации и определяет процесс *инкапсуляции данных* EAP, передаваемых между клиентами (запрашивающими устройствами) и серверами аутентификации. Стандарт *IEEE 802.1X* осуществляет *контроль* доступа и не позволяет неавторизованным устройствам подключаться к локальной сети через порты коммутатора.

Роли устройств в стандарте 802.1X.

В стандарте *IEEE 802.1X* определены следующие три роли, которые могут выполнять устройства:

- клиент (Client/Supplicant);

- *аутентификатор* (Authenticator);
- сервер аутентификации (*AuthenticationServer*).

Сервер аутентификации RemoteAuthenticationinDial-InUserService (RADIUS) проверяет права доступа каждого клиента, подключаемого к порту коммутатора, прежде чем разрешить доступ к любому из сервисов, предоставляемых коммутатором или локальной сетью.

До тех пор, пока клиент не будет аутентифицирован, через порт коммутатора, к которому он подключен, будет передаваться только трафик протокола ExtensibleAuthenticationProtocolover LAN (EAPOL). Обычный трафик станет передаваться через порт коммутатора сразу после успешной аутентификации клиента.

Состояние порта коммутатора определяется тем, получил или не получил клиент право доступа к сети. Первоначально порт находится в*неавторизованном* состоянии. В этом состоянии он запрещает прохождение всего входящего и исходящего трафика за исключением пакетов EAPOL. Когда клиент аутентифицирован, порт переходит в *авторизованное* состояние, позволяя передачу через него любого трафика.

В стандарте *IEEE 802.1X* определены следующие три роли, которые могут выполнять устройства:

- клиент (Client/Supplicant);
- *аутентификатор* (Authenticator);
- сервер аутентификации (*AuthenticationServer*).

Клиент (Client/Supplicant) — это рабочая станция, которая запрашивает доступ к локальной сети и сервисам коммутатора и отвечает на запросы от коммутатора. На рабочей станции должно быть установлено клиентское ПО для 802.1X, например, то, которое встроено в ОС MicrosoftWindows XP.

Сервер аутентификации (AuthenticationServer) выполняет фактическую аутентификацию клиента. Он проверяет подлинность клиента и информирует коммутатор, предоставлять или нет клиенту доступ к локальной сети. RADIUS (RemoteAuthenticationDial-InUserService) работает в модели "клиент-сервер", в которой информация об аутентификации передается между сервером RADIUS и клиентами.

Аутентификатор (Authenticator) управляет *физическим доступом* к сети, основываясь на статусе аутентификации клиента. Эту роль выполняет коммутатор. Он работает как посредник (Proxy) между клиентом и сервером аутентификации: получает запрос на проверку подлинности от клиента, проверяет данную информацию при помощи сервера аутентификации и пересылает ответ клиенту. Коммутатор поддерживает клиент RADIUS, который отвечает за инкапсуляцию и деинкапсуляцию кадров *EAP* и взаимодействие с сервером аутентификации.

Инициировать процесс аутентификации могут или коммутатор, или клиент.

Клиент инициирует аутентификацию, посылая кадр EAPOL-start, который вынуждает коммутатор отправить ему запрос на идентификацию. Когда клиент отправляет EAP-ответ со своей идентификацией, коммутатор начинает играть роль посредника, предающего кадры EAP между клиентом и сервером аутентификации до успешной или неуспешной аутентификации. Если аутентификация завершилась успешно, порт коммутатора становится авторизованным.

. МЕТОДИЧЕСКИЕ УКАЗАНИЯПО ПРОВЕДЕНИЮ ЛАБОРАТОРНЫХ РАБОТ

2.1 Лабораторная работа № 1-5 (10 часов).

Тема: «Основные команды коммутаторов»

2.1.1 Задание для работы:

1.Ознакомиться с основными командами настройки коммутаторов D-Link.

2. Ознакомиться с основными командами контроля коммутаторов D-Link.

3. Ознакомиться с основными командами устранения неполадок коммутаторов D-Link.

2.1.2 Краткое описание лабораторной работы:

- Настройка DES-3200-28
- Вызов помощи по командам
- Управление учетными записями пользователей
- *Настройка параметров идентификации коммутатора*
- *Настройкапараметровбаннеровприветствия (Login banner (greeting message) and Command Prompt)*
- *Настройка времени на коммутаторе*
- *Настройка основных параметров портов Ethernet коммутатора*
- *Функция FactoryReset (сброс к заводским установкам)*

2.1.3 Результаты и выводы:

Коммутаторы D-Link классифицируются по возможностям управления. Существует три основных типа:

- неуправляемые коммутаторы
- настраиваемые коммутаторы
- управляемые коммутаторы

Обновление программного обеспечения может быть необходимо, когда доступна новая функциональность или требуется коррекция ошибок. Сохранять конфигурацию коммутатора необходимо при изменении его настроек, а также для упрощения восстановления функционирования коммутатора в результате сбоя его работы или поломки. Для загрузки программного конфигурации необходимо наличие в сети TFTP-сервера. Коммутаторы D-Link поддерживают возможность хранения на коммутаторе двух версий программного обеспечения и конфигурации, причем любая из них может быть настроена в качестве основной, что позволяет обеспечить отказоустойчивость оборудования при переходе на новое программное обеспечение или изменении конфигурации. Для анализа работы коммутатора имеется возможность выгрузки через протокол TFTP Log-файла.

2.2 Лабораторная работа № 6-9 (8 часов).

Тема: «Команды управления таблицами коммутации MAC- и IP-адресов, ARP-таблицы»

2.2.1 Задание для работы:

1. Изучить процесс управления таблицами MAC.

2. Изучить процесс управления таблицами IP.

3. Изучить процесс управления таблицами ARP.

2.2.2 Краткое описание лабораторной работы:

Настройка DES-3200-28

Изучение команд просмотра таблиц MAC-адресов

Изучение команд управления таблицей MAC-адресов

Настройка DGS-3612G (работа с таблицей коммутации уровня 3 (IP FDB))

Изучение команд просмотра таблиц коммутации IP-адресов

Настройка DES-3200-28 /DGS-3612G (управление ARP-таблицами)

Изучение команд просмотра ARP-таблиц

Изучение команд управления ARP-таблицей

2.2.3 Результаты и выводы:

При эксплуатации активного сетевого оборудования сетевые администраторы вынуждены тратить до 70% своего на изменение конфигурации активного оборудования вследствие изменения месторасположения рабочих мест пользователей, миграции пользователей между отделами и т.п. Для этого администратору необходимо максимально быстро определить порт подключения клиентского оборудования на основе MAC- и IP-адресов и перевести его в нужную VLAN и IP-подсеть.

Большинство современных коммутаторов, независимо от производителя, поддерживают множество дополнительных возможностей, отвечающих общепринятым стандартам. Среди них самые распространенные и наиболее используемые сегодня:

- виртуальные локальные сети (VLAN);
- семейство протоколов SpanningTree — IEEE 802.1D, 802.1w, 802.1s;
- статическое и динамическое по протоколу IEEE 802.3ad агрегирование каналов Ethernet;
- сегментация трафика;
- обеспечение качества обслуживания QoS;
- функции обеспечения безопасности, включая аутентификацию 802.1X, функции PortSecurity, IP-MAC-PortBinding и т.д.;
- протоколы поддержки Multicast-вещания;
- SNMP-управление и др.

2.3 Лабораторная работа № 10-13 (8 часа).

Тема: «Команды обновления программного обеспечения коммутатора и сохранения/восстановления конфигурационных файлов»

2.3.1 Задание для работы:

1. Изучить процесс обновления программного обеспечения

2. Изучить процесс сохранения/восстановления конфигурации.

3. Изучить процесс восстановления конфигурации.

2.3.2 Краткое описание лабораторной работы:

Настройка DES-3200-28

Подготовка к режиму обновления и сохранения программного обеспечения коммутатора

Загрузка файла программного обеспечения в память коммутатора

Настройка порядка загрузки программного обеспечения коммутатора

Управление изменением конфигурации

Выгрузка Log-файлов

Выгрузка информации для технической поддержки (Tech_support)

2.3.3 Результаты и выводы:

Обновление программного обеспечения (его иногда называют "прошивкой" коммутатора) может быть необходимо, когда доступна новая функциональность или требуется коррекция ошибок. Сохранять конфигурацию коммутатора необходимо при изменении его настроек, а также для упрощения восстановления функционирования коммутатора в результате сбоя его работы или поломки. Основным протоколом, применяемым для этих целей, служит протокол TFTP (TrivialFileTransferProtocol, простейший протокол передачи данных). Для передачи/загрузки программного обеспечения/конфигурации необходимо наличие в сети TFTP-сервера. Коммутаторы D-Link поддерживают возможность хранения на коммутаторе двух версий программного обеспечения и конфигурации, причем любая из них может быть настроена в качестве основной, т.е. используемой при загрузке коммутатора. Это позволяет обеспечить отказоустойчивость оборудования при переходе на новое программное обеспечение или изменении конфигурации. Для анализа работы коммутатора имеется возможность выгрузки через протокол TFTP Log-файла.

2.4 Лабораторная работа № 14-17 (8 часов).

Тема: «Настройка VLAN на основе портов и стандарта IEEE 802.1Q»

3.4.1 Задание для работы:

1. Понять технологию VLAN.
2. Понять технологию настройки VLAN на коммутаторах D-Link.

2.4.2 Краткое описание лабораторной работы:

Настройка VLAN на основе портов

Настройка DES-3200-28

Настройка VLAN на основе стандарта IEEE 802.1Q

2.4.3 Результаты и выводы:

Виртуальная локальная сеть (VirtualLocalAreaNetwrok, VLAN) представляет собой коммутируемый сегмент сети, который логически выделен по выполняемым функциям, рабочим группам или приложениям, вне зависимости от физического расположения пользователей. Виртуальные локальные сети имеют все свойства физических локальных сетей, но рабочие станции можно группировать, даже если они физически расположены не в одном сегменте, т.к. любой порт коммутатора можно настроить на принадлежность определенной VLAN. При этом одноадресный, многоадресный и широковещательный трафик будет передаваться только между рабочими станциями, принадлежащими одной VLAN. Каждая VLAN рассматривается как логическая сеть, т.е. кадры, предназначенные станциям, которые не принадлежат данной VLAN, должны передаваться через маршрутизирующее устройство (маршрутизатор или коммутатор 3-го уровня). Таким образом, с помощью виртуальных локальных сетей решается проблема ограничения области передачи широковещательных кадров и вызываемых ими следствий, которые существенно снижают производительность сети, вызывают широковещательные штормы.

Типы VLAN:

- VLAN на основе портов (Port-based VLAN

- VLAN на основе MAC-адресов (MAC-based VLAN
- VLAN на основе портов и протоколов IEEE 802.1v — тип протокола используется для определения членства в VLAN;
- VLAN на основе стандарта IEEE 802.

Существуют два метода назначения порта в определенную VLAN:

- статическое назначение
- динамическое назначение

2.5 Лабораторная работа № 18-21 (8 часов).

Тема: «Настройка протоколов связующего дерева STP, RSTP, MSTP»

2.5.1 Задание для работы:

1. Понять функционирование протоколов связующего на коммутаторах D-Link.
2. Изучить настройку протоколов связующего дерева на коммутаторах D-Link.

2.5.2 Краткое лабораторной работы:

Настройка DES-3200-28_A

Настройка DES-3200-28_E

Упражнения

Настройка протокола MSTP (IEEE 802.1s) для каждой VLAN

2.5.3 Результаты и выводы:

Протокол связующего дерева SpanningTreeProtocol (STP) является протоколом 2 уровня модели OSI, который позволяет строитьдревовидные, свободные от петель конфигурации связей между коммутаторами локальной сети.ля построения устойчивой активной топологии с помощью протокола STP необходимо с каждым коммутатором сети ассоциировать уникальный идентификатор моста (Bridge ID), с каждым портом коммутатора ассоциировать стоимость пути (PathCost) и идентификаторпорта (Port ID).Процесс вычисления связующего дерева начинается с выбора корневого моста (RootBridge), от которого будет строиться дерево. Второй этап работы STP — выбор корневых портов (RootPort). Третий шаг работы STP — определение назначенных портов (DesignatedPort).

Протокол RapidSpanningTreeProtocol (RSTP) является развитием протокола STP. Основные понятия и терминология протоколов STP и RSTPодинаковы. Существенным их отличием является способ перехода портов в состояние продвижения и то, каким образом этот переход влияет на роль порта в топологии. RSTP объединяет состояния Disabled, Blocking и Listening, используемые в STP, и создает единственное состояние Discarding ("Отбрасывание"), при котором порт не активен. Выбор активной топологии завершается присвоением протоколомRSTP определенной роли каждому порту: корневой порт (RootPort), назначенный порт (DesignatedPort), альтернативный порт (AlternatePort), резервный порт (BackupPort).

Протокол MultipleSpanningTreeProtocol (MSTP) является расширением протокола RSTP, который позволяет настраивать отдельное связующее дерево для любой VLAN или группы VLAN, создавая множество маршрутов передачи трафика и позволяя осуществлять балансировку нагрузки.Протокол MSTP делит коммутируемую сеть на регионы MST (MultipleSpanningTree (MST) Region), каждый из которых может содержать

множество копий связующих деревьев (MultipleSpanningTreeInstance, MSTI) с независимой друг от друга топологией.

При вычислении активной топологии связующего дерева IST и MSTI не используют значения полей MaxAge и MessageAge конфигурационного BPDU для отбрасывания устаревших сообщений. Вместо этого используется механизм счетчика переходов (Hopcount).

При вычислении активной топологии связующего дерева IST и MSTI не используют значения полей MaxAge и MessageAge конфигурационного BPDU для отбрасывания устаревших сообщений. Вместо этого используется механизм счетчика переходов (Hopcount).

С помощью команды configstpmaxhops на коммутаторах D-Link можно настроить максимальное число переходов между устройствами внутри региона, прежде чем кадр BPDU будет отброшен. Значение счетчика переходов устанавливается региональным корневым мостом MSTI или CIST и уменьшается на 1 каждым портом коммутатора, получившим кадр BPDU. После того как значение счетчика станет равным 0, кадр BPDU будет отброшен и информация, хранимая портом, будет помечена как устаревшая.

В данной лабораторной работе рассматривается работа протоколов связующего дерева и их настройка на коммутаторах

2.6 Лабораторная работа № 22-25 (8 часов).

Тема: «Установка и настройка протокола IPv6 на рабочей станции и коммутаторе D-Link»

2.6.1 Задание для работы:

1. Изучить процесс настройки IPv6 на рабочей станции и коммутаторе D-Link.

2.6.2 Краткое описание лабораторной работы:

настройка IPv6 на рабочей станции

настройка IPv6 на коммутаторе D-Link

2.6.3 Результаты и выводы:

Интернет-протокол версии 6 (IPv6) представляет сетевой слой пакетной передачи данных между сетями. Он разрабатывается в качестве преемника IPv4, текущей версии интернет-протокола для общего использования в Интернете.

Основным отличием IPv6 является гораздо большее адресное пространство, что добавляет большую гибкость при распределении адресов. Увеличенная длина адреса позволяет отказаться от использования NAT (networkaddresstranslation), что позволяет избежать нехватки интернет-адресов, а также упрощает назначения адресов и нумерации при смене интернет-провайдера.

Адресное пространство IPv6 поддерживает примерно 3,4x1038 адресов.

Большое число адресов позволяет использовать иерархическое распределение адресов, упрощая маршрутизацию. В IPv4, сложные CIDR-методы были разработаны для максимально эффективного использования адресного пространства. Изменение нумерации, при смене провайдеров, может вызвать серьезные проблемы с IPv4 (это уже обсуждается в RFC 2071 и RFC 2072). В IPv6 изменение нумерации осуществляется практически автоматически, поскольку идентификатор узла (хоста) отделен от идентификатора сети провайдера. Разделение адресных пространств провайдеров и узлов

добавляет "неэффективные" биты в адресное пространство, однако чрезвычайно эффективно для решения оперативных вопросов, таких как изменение сервис-провайдера.

2.7 Лабораторная работа № 26-29 (8 часов).

Тема: «Настройка QoS. Приоритизация трафика. Управление полосой пропускания»

2.7.1 Задание для работы:

1. Изучить настройку приоритизации трафика на коммутаторах D-Link.

2.Изучить возможности управления полосой пропускания

2.7.2 Краткое описание лабораторной работы:

Настройка DES-3200-28_A

Настройка DES-3200-28_B

Упражнения

2.7.3 Результаты и выводы:

Сети с коммутацией пакетов на основе протокола IP не обеспечивают гарантированной пропускной способности, поскольку не обеспечивают гарантированной доставки.

Для приложений, где не важен порядок и интервал прихода пакетов, время задержек между отдельными пакетами не имеет решающего значения. Для приложений, чувствительных к задержкам, в сети должны быть реализованы механизмы, обеспечивающие функции качества обслуживания (QualityofService, QoS).

Функции качества обслуживания в современных сетях заключаются в обеспечении гарантированного и дифференцированного уровня обслуживания сетевого трафика, запрашиваемого теми или иными приложениями на основе различных механизмов распределения ресурсов, ограничения интенсивности трафика, обработки очередей и приоритизации.

Для обеспечения QoS на канальном уровне модели OSI коммутаторы поддерживают стандарт IEEE 802.1p. Стандарт IEEE 802.1p позволяет задать до 8 уровней приоритетов (от 0 до 7, где 7 — наивысший), определяющих способ обработки кадра, используя 3 бита поля приоритета тега IEEE 802.1Q.

В лабораторной работе рассматривается следующий пример: на компьютерах B и D запущены приложения VoIP, и им необходимо обеспечивать высокий приоритет обработки по сравнению с приложениями других станций.

Современные коммутаторы позволяют регулировать интенсивность трафика на своих портах с целью обеспечения функций качества обслуживания.

Для управления полосой пропускания входящего и исходящего трафика на портах *Ethernet* коммутаторы D-Link поддерживают функцию*Bandwidth Control*, которая использует для ограничения скорости механизм *Traffic Policing*. *Администратор* может вручную устанавливать требуемую скорость соединения на порте в диапазоне от 64 Кбит/с до максимально поддерживаемой скорости интерфейса с шагом 64 Кбит/с.

2.8 Лабораторная работа № 30-33 (8 часов).

Тема: «Списки управления доступом (AccessControlList)»

2.8.1 Задание для работы:

1. На коммутаторе D-Linkнастроитьсписки управления доступом, в качестве критериев фильтрации используются MAC-и IP-адреса.

2.8.2 Краткое описание Лабораторной работы:

Настройка ограничения доступа пользователей в Интернет по MAC-адресу

Настройка ограничения доступа пользователей в Интернет по IP-адресам

Настройка ограничения доступа пользователей в Интернет по IP-адресам

2.8.3 Результаты и выводы:

Списки управления доступом (AccessControlList, ACL) являются средством фильтрации потоков данных. Фильтруя потоки данных, администратор может ограничить типы приложений, разрешенных для использования в сети, контролировать доступ пользователей к сети и определять устройства, к которым они могут подключаться. Также ACL могут использоваться для определения политики QoS путем классификации трафика и переопределения его приоритета.

ACL представляют собой последовательность условий проверки параметров пакетов данных. Когда сообщения поступают на входной интерфейс, коммутатор проверяет параметры пакетов данных на совпадение с критериями фильтрации, определенными в ACL, и выполняет над пакетами одно из действий: Permit ("Разрешить") или Deny ("Запретить").

Списки управления доступом состоят из профилей доступа (AccessProfile) и правил (Rule). Профили доступа определяют типы критериев фильтрации, которые должны проверяться в пакете данных (MAC-адрес, IP-адрес, номер порта, VLAN и т.д.), а в правилах указываются непосредственные значения их параметров. Каждый профиль может состоять из множества правил.

В коммутаторах D-Link существует три типа профилей доступа: Ethernet, IP и PacketContentFiltering (фильтрация по содержимому пакета).